

Limited utility of residue masking in positive-selection inference

Stephanie J. Spielman^{1*} and Eric T. Dawson¹ and Claus O. Wilke¹

Address:

¹Department of Integrative Biology, Center for Computational Biology and Bioinformatics,
and Institute of Cellular and Molecular Biology. The University of Texas at Austin, Austin,
TX 78731, USA.

*Corresponding author

Email: stephanie.spielman@utexas.edu

Manuscript type: research article

Keywords: multiple sequence alignment, alignment filters, positive-selection inference, sequence simulation

Abstract

Recent studies have noted that errors in multiple sequence alignments have the potential to reduce accuracy in positive-selection inference. It has therefore been suggested that users filter their alignments before further analyses. One widely-used filter, known as Guidance, generates position-based confidence scores for a given alignment using a bootstrap approach, thereby allowing users to remove positions of low confidence from the alignment. Here, we have reimplemented the Guidance software to include two novel algorithms for phylogenetically correcting confidence scores. We additionally introduced a new score normalization scheme which assigned lower scores to highly-gapped alignment columns. Using a sequence simulation approach, we compared our reimplemented Guidance algorithms to the original. We found, surprisingly, that no approach examined could substantially improve positive-selection inference across multiple inference methods, including PAML and FUBAR. Instead, it seemed that the analysis methods used influenced positive-selection inference substantially more than did filtering alignments with a Guidance-based strategy.

Introduction

Constructing a multiple sequence alignment represents the first step of analysis in most studies of molecular evolution, namely phylogenetic reconstruction and evolutionary rate inference. Recently, several studies have shown that poor alignment quality can significantly hinder accuracy in such downstream analyses (Jordan and Goldman 2012; Markova-Raina and Petrov 2011; Dwivedi and Gadagkar 2009; Talavera and Castresana 2007; Ogden and Rosenberg 2006). In particular, using low-quality MSAs during positive-selection inference elevated false positive rates (Schneider et al. 2009; Fletcher and Yang 2010; Markova-Raina and Petrov 2011). As a consequence, some studies have advocated applying an alignment filter to MSAs (Jordan and Goldman 2012; Privman et al. 2012). Such filters, which include Guidance (Penn et al. 2010; Privman et al. 2012), GBLOCKS (Castresana 2000), and T-Coffee (Notredame et al. 2000), locate putatively poorly aligned regions in alignments, thereby allowing users to curate their alignments to maximize signal. The hope is that culling unreliable positions and/or columns from MSAs will yield increased accuracy in positive-selection inference without excessively sacrificing power.

Of these filters, Guidance (Penn et al. 2010) is currently the most widely used in positive-

selection inference. Guidance derives a confidence score for each MSA position using a bootstrap approach that samples variants in the guide tree used to construct progressive alignments. Using the resulting scores, users can mask positions that score below a given threshold, thereby removing residues that cannot be confidently placed in the alignment. This method is fairly conservative, as particular positions of low confidence can be removed rather than entire columns. Recent simulation studies have demonstrated that filtering poorly aligned residues with Guidance may increase accuracy when inferring positive selection (Jordan and Goldman 2012; Privman et al. 2012).

However, while Privman et al. (2012) found dramatic improvements in positive-selection inference when applying the Guidance filter, a comprehensive study by Jordan and Goldman (2012) on alignment methods and filtering found that Guidance affected this inference modestly, if at all. In particular, they reported that, at lower protein divergence levels or insertion/deletion (indel) rates, Guidance neither substantially improved nor worsened positive-selection detection. Alternatively, at high divergence levels (1.8 mean path length, defined as mean root-to-tip branch length) or indel rates (0.2 indel/substitution events), Jordan and Goldman (2012) found that Guidance strongly boosted true positive rates. While these results were compelling, protein sequences used in positive selection studies rarely, if ever, contain sequences separated by such high divergences; for example, a typical mammalian gene tree’s MPL (defined as the mean distance from root-to-tip across all taxa in a phylogeny) is only about 0.17 (Spielman and Wilke 2013), 10% the divergence level at which Jordan and Goldman detected improvements when using the Guidance filter.

Motivated by these apparent discrepancies, we sought to determine whether altering the Guidance scoring algorithm could improve positive-selection inference at more realistic divergences. To this end, we reimplemented the Guidance software (see Methods for details) and examined the effects of new scoring algorithms, which take the sequences’ phylogenetic relationships into account. Using this reimplementation, we filtered alignments produced from realistic protein-coding sequence simulations and subsequently inferred positive selection with two methods: the recently-described method FUBAR (Murrell et al. 2013) and the current standard of use, PAML’s M8 model (Yang 2007). Overall, we found that neither the original Guidance filter nor our newly implemented filters conferred any substantial benefits to positive-selection inference, and any improvements recovered were of extremely small magnitude. Instead, the inference method used had a much greater effect

on positive-selection detection.

Results

Guidance reimplementations and analysis pipeline

To systematically evaluate Guidance’s influence on positive-selection inference, we reimplemented the Guidance software in Python and C++. Before conducting any analyses, we verified that, given a set of perturbed alignments, our program produced the same confidence scores as the original Guidance. In addition to our basic Guidance reimplementations, we derived two new algorithms which employ phylogenetic weighting when assigning confidence scores (see Methods for details). Briefly, the first method incorporated a weight for each sequence in the alignment, as calculated by BranchManager (Stone and Sidow 2007), and the second method incorporated patristic distances (the sum of branch lengths between two taxa). We called these methods, respectively, BMweights and PDweights. We additionally defined a “gap-penalization” score-normalization scheme, which naturally assigned lower confidence scores to residues in highly gapped regions, given that such regions were more likely to be poorly aligned. We referred to filters using the gap-penalization scheme as GuidanceP, BMweightsP, and PDweightsP.

We began by simulating 100 realistic protein-coding sequences using Indelible (Fletcher and Yang 2009) along each of four different gene trees of sizes 11, 26, 60, and 158 taxa. To ensure that these simulations produced real sequence data to the extent possible, we simulated according to evolutionary parameters inferred from H1N1 influenza hemagglutinin (HA), a protein well known to contain positively selected regions (Bush et al. 1999; ?; Meyer and Wilke 2012). We then processed the unaligned sequences with our Guidance reimplementations using the aligner MAFFT L-INS-I (linsi) (Katoh et al. 2005). We chose to use only linsi, which strongly outperforms other similar alignment softwares for protein alignments (Thompson et al. 2011; Nuin et al. 2006) without sacrificing speed. We scored each alignment using each of our three scoring algorithms (original Guidance, BMweights, and PDweights) with each of our two normalization schemes (original Guidance and gap-penalization). In all cases, we masked positions with scores below 0.5, the same threshold as used by Jordan and Goldman (2012), with “?”.

Finally, we assessed positive selection with both the recently-described FUBAR (Murrell

et al. 2013) as implemented in HyPhy (Kosakovsky Pond et al. 2005) and the widely-used M8 model in PAML (Yang 2007). Note that we use PAML throughout this paper to refer specifically to the M8 model. All phylogenies used during these inferences were constructed from unfiltered amino acid alignments in RAxML (Stamatakis 2006) using the CAT model under the WAG substitution matrix. All filtered alignments stemming from the same unfiltered alignment were analyzed with the same phylogeny to remove any potential bias that distinct phylogenies could introduce. We considered sites as positively selected if the given method returned a posterior probability ≥ 0.90 . Note that, while we processed all alignments with FUBAR, we did not infer positive selection for the largest simulation set with PAML due to prohibitive runtimes.

Filtering with Guidance-based methods has a minimal effect on positive-selection inference

We assessed how filtering alignments with a Guidance-based method influences positive-selection detection by comparing the resulting true positive rates (TPRs) of positive-selection inference between all filtered alignments and their corresponding unfiltered alignments. We chose to primarily compare the TPRs among alignments as opposed to the false positive rates (FPRs), as FPRs were exceedingly small, never exceeding an average of 0.017 across simulation sets and methods. These FPRs were similar to those recovered by Jordan and Goldman (2012).

We first compared the two normalization schemes, original and gap-penalization, to one another for each of the three algorithms (Guidance, BMweights, and PDweights) examined. To this end, we built linear models with the difference in TPR between normalization schemes as the response and algorithm as a fixed effect (Table 1) for each simulation set and inference method. Note that only filtered alignments were considered in these models. In all instances in which we detected a significant difference between normalization schemes, the gap-penalization scheme outperformed the original Guidance scheme, albeit by very small magnitudes. Therefore, we proceeded to compare the TPRs from gap-penalization algorithms (GuidanceP, BMweightsP, PDweightsP) to those of unfiltered alignments with a series of mixed-effects models. These models included TPR as the response variable, simulation count as a random effect, thus accounting for the paired structure of our analysis, and algorithm as a fixed effect. Table 2 gives the results from these models for each simulation

set and selection inference method.

Overall, we did not recover any difference in TPR among gap-penalization filtering algorithms; Guidance and the two phylogenetically-weighted methods performed statistically indistinguishably for a given simulation set. Figure 1 displays ROC curves for positive-selection inference with FUBAR and PAML. Figure 1B and 1C highlight the effect of alignment filtering with FUBAR and PAML, respectively; for each respective method, areas under the unfiltered alignment curves were essentially the same as for the filtered alignment curves.

When processed with FUBAR, alignment filtering significantly increased TPRs for the 60 and 158-sequence simulation sets. The magnitude of its improvements were very small, boosting unfiltered TPRs by an average of 0.03 for the 60-sequence set and by 0.01 for the 158-sequence set. In other words, TPR increased by 10% for the 60-sequence simulation set and by 3% for the 158-sequence simulation set. Filtering neither improved nor worsened positive-selection inference for the 11 or 26-sequence simulation sets as analyzed with FUBAR.

Analyzing the data with PAML similarly did not result in a significant TPR change for the 11-sequence simulation set. Unlike when analyzing with FUBAR, we did not recover significant TPR changes for the 60-sequence simulation set. Alternatively, PAML results showed that filtering did improve TPRs in the 26-sequence simulation set, albeit by an average of 4.68%. While statistically significant, an increase of this low a magnitude may not have any noticeable effect on positive-selection inference in studies of real sequences.

Influence of posterior probability on selection inference methods

Our analysis recovered several differences in how PAML and FUBAR behaved when assessing positive selection. In particular, whether FUBAR or PAML performed better depended on the number of sequences analyzed; FUBAR outperformed PAML for the two smaller simulation sets of 11 and 26-sequences, but PAML outperformed FUBAR for the 60-sequence simulation set (Table 2). We found that this trend resulted from the posterior probability threshold chosen to call sites as positively selected. Figure 2 shows how TPRs and FPRs for unfiltered alignments behaved for two different simulation sets (26 and 60-sequence) across posterior probability cutoffs. With regards to TPRs, FUBAR generally performed better than PAML at lower posterior probabilities, whereas PAML outperformed FUBAR

at high probabilities. As the number of sequences increased, the intersection between methods' TPRs shifted towards lower posterior probabilities. PAML's improved ability to call true positives with the inclusion of more sequences largely dictated this shift, as FUBAR's relationship between posterior probability and TPR remained similar between the two simulation sets. Filtering alignments affected these broad trends that FUBAR and PAML portrayed, only marginally, if at all (Figure 3). Thus, the analysis method used affected positive-selection inference substantially more than did filtering alignments with a Guidance-based method.

The relationship between posterior probability and FPR, unlike that of TPR, was mostly consistent across simulation sets for both FUBAR and PAML. While PAML achieved FPRs of nearly zero essentially immediately for both the 26 and 60-sequence simulation sets, FUBAR approached a zero FPR more slowly, reaching zero around a posterior probability of 0.8 for each simulation set. As a typical study of positive selection would almost definitely use a posterior probability above at least 0.8, alignment filtering would likely not be able to substantially reduce FPR.

How TPR and FPR behaved for FUBAR and PAML evidenced the different approaches these methods employ to detect positive selection; while PAML's M8 model aims to identify precise dN/dS value at each site in an alignment, FUBAR merely assesses whether a site is positively selected but does not attempt to assign a point estimate of dN/dS to each position. Having fewer sequences in an alignment, therefore, hindered PAML's ability to determine site-based dN/dS values, rendering its positive-selection inference worse than that of FUBAR. Increasing the number of sequences allowed for PAML to make more robust dN/dS estimates while having a relatively smaller effect on FUBAR's approximations.

Altering masking thresholds may (hinder?)...???

When filtering alignments with Guidance-based methods, one must select a specific score cutoff below which to mask residues. We chose to filter all residues with scores less than 0.5. However, it was possible that selecting a different threshold would have yielded different results. To assess whether our selected cutoff biased results, we masked all alignments at cutoffs of 0.3, 0.7 and 0.9 in addition of 0.5. As our phylogenetically-corrected scoring algorithms were unable to significantly improve upon the original Guidance algorithm, we considered only the original algorithm, although with both normalization schemes, for this

analysis. Further, we only inferred positive selection with FUBAR, whose rapid runtime (under 15 minutes per alignment) allowed for such a large-scale investigation.

Similar to previous analyses, we built two mixed-effects linear models for each simulation set, with TPR as the response, masking cutoff as a fixed effect, and simulation count as the random effect, for Guidance and GuidanceP results. Results showed that, for Guidance, all masking cutoffs yielded statistically similar TPRs (multiple comparisons gave all $P > 0.06$) for a given simulation set. For GuidanceP, on the other hand, masking cutoffs 0.3, 0.5, and 0.7 gave statistically similar TPRs (all $P > 0.45$), whereas using a masking cutoff of 0.9 always yielded significantly lower TPRs than the other cutoffs. Table ?? highlights this result, specifically comparing the cutoff of 0.5 to 0.9 for GuidanceP. Importantly, for the 60-sequence simulation set (for which we noted the maximum benefit of alignment filtering), masking at a stringent cutoff of 0.9 as opposed the more lenient 0.5 reduced the TPR by roughly 26%. This extreme TPR decrease yielded an average TPR of 0.281, well below its corresponding average unfiltered TPR of 0.345 (Table 2). The gap-penalization normalization, therefore, had the potential to remove excessive amounts of information from alignments and worsen positive-selection detection.

Discussion

We have conducted a simulation-based study to evaluate the how introducing phylogenetically-aware scoring algorithms into the Guidance alignment filter influences the detection of positive selection in protein-coding sequences. Additionally, we tested a novel score normalization method, which assigned inherently lower scores to highly-gapped columns in an alignment. We ensured that our simulations resembled real sequence data by simulating along real gene trees according to evolutionary parameters derived from the H1N1 influenza hemagglutinin protein. We inferred positive selection for all alignments using two methods: the recently introduced and very fast FUBAR (Murrell et al. 2013) within the HyPhy package (Kosakovsky Pond et al. 2005) and the current standard for evolutionary rate inference, PAML (Yang 2007) (specifically the M8 model).

We found that, while the gap-penalization scheme marginally improved upon the original normalization method, incorporating phylogenetic information into the scoring scheme did not significantly change accuracy in such analyses. Even so, filtering alignments with

any Guidance-based method, including the original, did not substantially improve positive-selection inference under any circumstance. That the phylogenetically-weighted algorithms did not improve upon the original Guidance algorithm indicated the minimal benefits that filtering in this manner produced at all. Were the original Guidance to offer robust improvements in positive-selection detection, one might expect that our more statistically controlled approach would boost the method’s performance. However, as we have found that masking individual positions in an alignment only marginally affected positive-selection inference in the first place, one might not expect the algorithmic changes we implemented might to have a dramatic effect.

While others (Schneider et al. 2009; Fletcher and Yang 2010; Markova-Raina and Petrov 2011; Privman et al. 2012) have noted a prevalence of false positives in positive-selection inference of real sequence data, we recovered very low false positive rates. The FPRs we detected were very similar to those of Jordan and Goldman (2012), yet substantially lower than those detected by Privman et al. (2012). We attributed our small FPRs to the fact that we employed simulated sequences, which represents the only strategy to assess methodological accuracy with complete confidence. We ensured realism in our sequence data by applying evolutionary parameters inferred from a large alignment (1028 sequences) of H1N1 influenza hemagglutinin (HA), which previous studies have demonstrated to contain pervasive positive selection (Bush et al. 1999; ?; Meyer and Wilke 2012). We chose to use HA as a reference for evolutionary parameters because of its wealth of sequence data; having such a large data set allowed for robust evolutionary inference and the assurance that our simulations mimicked realistic sequence data. However, rather than conduct our simulations with this HA phylogenies, we decided to evolve sequences along eukaryotic gene trees for tractability. Simulating along a phylogeny of over 1000 sequences would not represent an excessive computational burden, but a typical positive-selection study would likely not use this many sequences. Additionally, the most important aspect to consider when selecting a phylogeny along which to simulate sequences is the phylogeny’s topology, and the gene trees we used represented the realistic topologies a typical study of evolutionary rates would encounter.

In sum, our simulations indicated that, while alignment filtering offered some benefits, those improvements were marginal at best. With such minimal changes to positive-selection inference, alignment filtering could easily decrease accuracy in a given positive selection

study. Indeed, we noted that using a stringent masking cutoff of 0.9 for algorithms normalized with our gap-penalization strategy resulted in extreme decreases in TPR relative to an unfiltered alignment. Choosing a low filtering threshold was necessary to achieve any improvement in positive-selection inference.

Overall, we cannot unequivocally recommend the use of a Guidance-based alignment filter when inferring evolutionary rates. Once an alignment has been constructed, it does not seem that much can be done to eliminate any misleading information. Instead, researchers should select inference methods in which the error can be minimized as much as possible without necessitating post-hoc correction. Therefore, we recommend that users select high-quality alignment and inference methods to minimize any obscuring signal, instead of relying on filters. If one must filter an alignment, we recommend using a lenient cutoff ≤ 0.5 to avoid sacrificing power, which might worsen inferences.

Methods

Guidance Reimplementation

Our reimplemented Guidance is written in Python and C++. Following the algorithm set forth in Penn et al. (Penn et al. 2010), we first create a reference amino-acid alignment using a user-specified progressive alignment software, with choices of Clustalw (Thompson et al. 1994), MUSCLE (Edgar 2004), or MAFFT (Katoh et al. 2002, 2005). We then generate N (where $N = 100$, by default) bootstrapped alignment replicates, each of which is used to create a bootstrapped tree in FastTree2 (Price et al. 2010). We then use these N trees as guide trees to create N new perturbed alignments, which we subsequently compare to the reference alignment to generate a confidence score for each residue. Users can specify options for aligner and phylogeny as desired.

Scoring Algorithms

We additionally implemented two new scoring algorithms which incorporate phylogenetic information. Before calculating scores, a phylogeny is built from the reference alignment. Our program includes functionality to build this phylogeny using either either FastTree2 (Price et al. 2010) or RAxML (Stamatakis 2006). Again, users can specify options to the phylogenetic construction software selected. Using this phylogeny, two types of phylogenetic

weights can be calculated. The first uses the software package BranchManager (Stone and Sidow 2007) to calculate a weight for each taxon in the phylogeny representing that taxon’s contribution to the phylogeny as a whole. We call this method “BMweights.” The second method calculates patristic distances (sum of branch lengths) between each taxon in the phylogeny using the python package DendroPy (Sukumaran and Holder 2010). We call this method “PDweights.”

We calculate positional confidence scores for each of the N bootstrap alignments as follows. A raw score, S , for a given residue in row i , column j of the reference alignment is calculated as

$$S_{ij} = \sum_{k \in R_{ng}^j} I_{ik}^j s_{ik}^j, \quad (1)$$

where R_{ng}^j represents all rows in column j which are not gaps. In this formula, the indicator function

$$I_{ik}^j = \begin{cases} 1 & \text{if reference alignment residue pair } (i, k) \text{ is present in bootstrap alignment} \\ 0 & \text{if reference alignment residue pair } (i, k) \text{ is absent in bootstrap alignment} \end{cases} \quad (2)$$

serves to compare the bootstrap- and reference-alignment residue pairings. We calculate s_{ik}^j according to the given scoring algorithm:

$$s_{ik}^j = \begin{cases} 1 & \text{if Guidance} \\ w_i w_k & \text{if BMweights} \\ d_p(i, k) & \text{if PDweights} \end{cases}, \quad (3)$$

where w_i is the phylogenetic weight of the taxon at row i , as calculated by BranchManager, and $d_p(i, k)$ is the patristic distance between the taxa at rows i and k .

We then sum positional scores S_{ij} determined from each bootstrap alignment and normalize them to yield a final score \tilde{S}_{ij} for each reference alignment residue. We use two different normalization schemes. The first scheme, given by

$$\tilde{S}_{ij} = \frac{\sum_n S_{ij}(n)}{N \sum_{k \in R_{ng}^j} S_{kj}}, \quad (4)$$

normalized by only considering the residue-residue comparisons made, as done by (Penn et al. 2010). The second scheme was given by

$$\tilde{S}_{ij} = \left(\sum_N^n S_{ij} \right) / \begin{cases} N \sum_{k \in R_{all}}^j 1 & \text{if Guidance} \\ N \sum_{k \in R_{all}}^j d_p(i, k) & \text{if BMweights} \\ Nw_i & \text{if PDweights} \end{cases} , \quad (5)$$

where R_{all}^j represents all rows in column j , including gap. This second normalization scheme naturally gives lower scores to highly gapped columns, and we therefore called it the “gap-penalization” normalization. We refer to the algorithms normalized by the first scheme as Guidance, BMweights, and PDweights. When normalized with the gap-penalization scheme, we refer to them, respectively, as GuidanceP, BMweightsP, and PDweightsP. Note that scores calculated using the Guidance algorithm with the first normalization scheme are equivalent to those originally derived by Penn et al. (2010).

Sequence Simulation

Coding sequences were simulated using Indelible (Fletcher and Yang 2009). To ensure that our simulations reflected realistic protein sequences, we simulated according to evolutionary parameters of the H1N1 hemagglutinin (HA) influenza protein. To derive these parameters, we aligned 1028 HA protein sequences with linsi (Kato et al. 2005) and then back-translated to a codon alignment using the original nucleotide sequence data. We generated a phylogeny from this codon alignment in RAxML (Stamatakis 2006) using the GTRGAMMA model. Using the codon alignment and phylogeny, we inferred evolutionary parameters with the REL (random effects likelihood) method (Nielsen and Yang 1998) using the software HyPhy (Kosakovsky Pond et al. 2005), with five evolutionary rate categories as free parameters under the GY94 evolutionary model (Goldman and Yang 1994). We employed a Bayes Empirical Bayes approach (Yang et al. 2000) to obtain infer dN/dS values at each site, which we used to assess a complete distribution of site rates. The resulting distribution was log-normal with a mean $dN/dS = 0.37$ with 8.3% of sites under positive selection ($dN/dS > 1$). We binned these rates into 50 equally spaced categories for specification in Indelible, which required a discrete distribution of dN/dS values. Again according to parameters derived from the HA analysis, we fixed κ , the transition-to-transversion ratio, at 5.3 for all simulations. We additionally set the state codon frequencies for our simulations according to those directly calculated from the HA alignment.

We simulated 100 alignments across four different real gene trees each, yielding a total of 400 simulated alignments. Phylogenies used included an 11-taxon tree of the mammalian olfactory receptor OR5AP2 (Spielman and Wilke 2013), a 26-taxon tree of mammalian rhodopsin sequences (Spielman and Wilke 2013), a 60-sequence tree of phosphoribulokinase (PRK) genes from photosynthetic eukaryotes (Yang et al. 2011), and a 158-taxon multilocus tree of flatfish sequences (Betancur-R et al. 2013). The latter two phylogenies were obtained from TreeBase. For each simulation set, we directly calculated an indel (insertion-deletion) rate directly from these trees original alignments, to use as simulation parameters, by dividing the total number of gaps present by the total number of positions in each alignment. Respectively, indel rates were 0.053, 0.019, 0.0041, and 0.0066.

Alignment and Positive-Selection Inference

We constructed all alignments using *linsi* (Katoh et al. 2002, 2005) within the context of our Guidance reimplementaion. In addition to an unfiltered alignment, we generated six filtered alignments (one for each filtering algorithm and each normalization scheme), masking residues with scores below cutoffs of 0.3, 0.5, 0.7, and 0.9. We inferred positive selection for every condition using both FUBAR (Murrell et al. 2013) with default parameters. We processed only the unfiltered alignments and alignments masked at a scoring cutoff of 0.5 with the PAML M8 model, specifying $F3 \times 4$ codon frequency and “cleandata = 0” in the control file (Yang 2007). Phylogenies specified for positive-selection inference were those constructed during the Guidance alignment procedure when deriving phylogenetic weights. All filtered alignments derived from a unfiltered alignment were processed with identical phylogenies to remove any confounding effects of differing phylogenies. Note that while we employed FUBAR to assess positive selection for all simulation sets, we did not use PAML to infer positive selection for the largest set (158 sequences).

We then compared resulting positive-selection inferences for each alignment to its respective true alignment’s dN/dS values, given by Indelible during simulation, to assess performance accuracy. As residues may have been differently aligned relative to the true simulated alignment, we adopted a consensus method to compare alignments we constructed to the true simulations. If at least 50% of the residues present in a true alignment column were present in an inferred alignment column, we considered the true alignment column’s dN/dS as the true value for that inferred alignment column. We considered sites positively

selected if the posterior probability of $(dN/dS > 1) \geq 0.9$.

Statistics were performed using Python and R. Linear modeling was conducted using the R packages lme4 (Bates et al. 2012) and multcomp (Hothorn et al. 2008). All code used is available at **the github or something**.

References

- Bates D, Maechler M, Bolker B. 2012. lme4: Linear mixed-effects models using S4 classes. R package version 0.999999-0.
- Betancur-R R, Li C, Munroe T A, Ballesteros J A, Orti G. 2013. Addressing gene tree discordance and non-stationarity to resolve a multi-locus phylogeny of the flatfishes (Teleostei: Pleuronectiformes). *Systematic Biology* 62(5):763–785.
- Bush R, Bender C, Subbarao K, Cox N, Fitch W. 1999. Predicting the evolution of human influenza a. *Science* 286:1921–1925.
- Castresana J. 2000. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. *Molecular Biology and Evolution* 17(4):540–552.
- Dwivedi B, Gadagkar S R. 2009. Phylogenetic inference under varying proportions of indel-induced alignment gaps. *BMC Evolutionary Biology* 9(1):211.
- Edgar R C. 2004. *Nucleic Acids Research* 32:1792–1797.
- Fletcher W, Yang Z. 2009. INDELible: A Flexible Simulator of Biological Sequence Evolution. *Molecular Biology and Evolution* 26(8):1879–1888.
- Fletcher W, Yang Z. 2010. The Effect of Insertions, Deletions, and Alignment Errors on the Branch-Site Test of Positive Selection. *Molecular Biology and Evolution* 27(10):2257–2267.
- Goldman N, Yang Z. 1994. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Molecular Biology and Evolution* 11:725–736.
- Hothorn T, Bretz F, Westfall P. 2008. Simultaneous inference in general parametric models. *Biometrical Journal* 50(3):346–363.

- Jordan G, Goldman N. 2012. The effects of alignment error and alignment filtering on the sitewise detection of positive selection. *Mol Biol Evol* 29:1125–1139.
- Katoh K, Kuma K I, Toh H, Miyata T. 2005. MAFFT version 5: improvement in accuracy of multiple sequence alignment. *Nucleic Acids Res* 33:511–518.
- Katoh K, Misawa K, Kuma K I, Miyata T. 2002. MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res* 30:3059–3066.
- Kosakovsky Pond S L, Frost S D W, Muse S V. 2005. HyPhy: hypothesis testing using phylogenies. *Bioinformatics* 12:676–679.
- Markova-Raina P, Petrov D. 2011. High sensitivity to aligner and high rate of false positives in the estimates of positive selection in the 12 *Drosophila* genomes. *Genome Research* 21(6):863–874.
- Meyer A G, Wilke C O. 2012. Integrating sequence variation and protein structure to identify sites under selection. *Mol Biol Evol* 30:36–44.
- Murrell B, Moola S, Mabona A, Weighill T, Scheward D, Kosakovsky Pond S L, Scheffler K. 2013. FUBAR: A Fast, Unconstrained Bayesian AppRoximation for inferring selection. *Molecular Biology and Evolution* 30:1196–1205.
- Nielsen R, Yang Z. 1998. Likelihood models for detecting positive selected amino acid sites and applications to the HIV-1 envelope gene. *Genetics* 148:929–936.
- Notredame C, Higgins D G, J H. 2000. T-Coffee: a novel method for fast and accurate multiple sequence alignment.
- Nuin P A S, Wang Z, Tillier E R M. 2006. The accuracy of several multiple sequence alignment programs for proteins. *BMC Bioinformatics* 7:471.
- Ogden T H, Rosenberg M S. 2006. Multiple Sequence Alignment Accuracy and Phylogenetic Inference. *Systematic Biology* 55(2):314–328.
- Penn O, Privman E, Landan G, Graur D, Pupko T. 2010. An alignment confidence score capturing robustness to guide tree uncertainty. *Mol Biol Evol* 27:1759–1767.
- Price M N, Dehal P S, Arkin A P. 2010. FastTree2: Approximately maximum-likelihood trees for large alignments. *PLoS One* 5:e9490.

- Privman E, Penn O, Pupko T. 2012. Improving the performance of positive selection inference by filtering unreliable alignment regions. *Mol Biol Evol* 29:1–5.
- Schneider A, Souvorov A, Sabath N, Landan G, Gonnet G H, Graur D. 2009. Estimates of positive darwinian selection are inflated by errors in sequencing, annotation, and alignment. *Genome Biology and Evolution* 1(0):114–118.
- Spielman S J, Wilke C O. 2013. Membrane environment imposes unique selection pressures on transmembrane domains of G protein-coupled receptors. *Journal of Molecular Evolution* 76:172–182.
- Stamatakis A. 2006. RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics* 22:21:2688–2690.
- Stone E, Sidow A. 2007. Constructing a meaningful evolutionary average at the phylogenetic center of mass. *BMC Bioinformatics* 8:222.
- Sukumaran J, Holder M T. 2010. DendroPy: A python library for phylogenetic computing. *Bioinformatics* 26:1569–1571.
- Talavera G, Castresana J. 2007. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. *Systematic Biology* 56(4):564–577.
- Thompson J D, Higgins D G, Gibson T J. 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Research* 22:4673–4680.
- Thompson J D, Linard B, Lecompte O, Poch O. 2011. A comprehensive benchmark study of multiple sequence alignment methods: Current challenges and future perspectives. *PLoS ONE* 6(3):e18093.
- Yang Y, Maruyama S, Sekimoto H, Sakayama H, Nozaki H. 2011. An extended phylogenetic analysis reveals ancient origin of “non-green” phosphoribulokinase genes from two lineages of “green” secondary photosynthetic eukaryotes: Euglenophyta and Chlorarachniophyta. *BMC Research Notes* 4:330.

- Yang Z. 2007. PAML 4: Phylogenetic analysis by maximum likelihood. *Molecular Biology and Evolution* 24:1586–1591.
- Yang Z, Nielsen R, Goldman N, Pedersen A M K. 2000. Codon-substitution models for heterogeneous selection pressure at amino acid sites. *Genetics* 155:431–449.

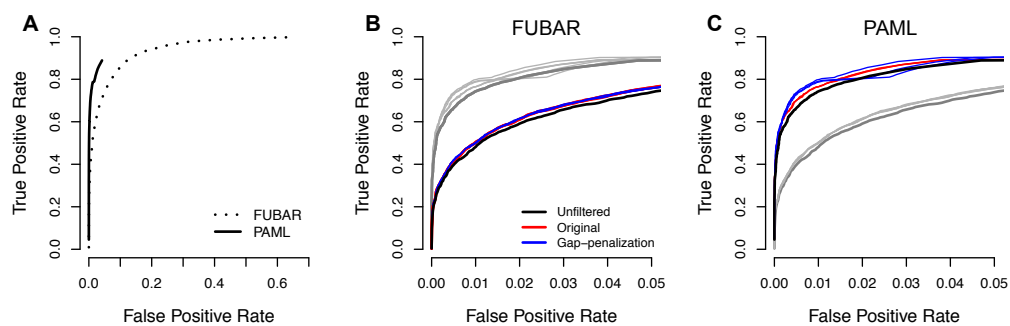


Figure 1: ROC curve as averaged across 60-sequence simulation set. A) Unfiltered alignments for FUBAR (solid) and PAML (dashed). Note that neither method achieved FPRs greater than shown. B) FUBAR in color with PAML results shown in grey. C) PAML in color with FUBAR results shown in grey.

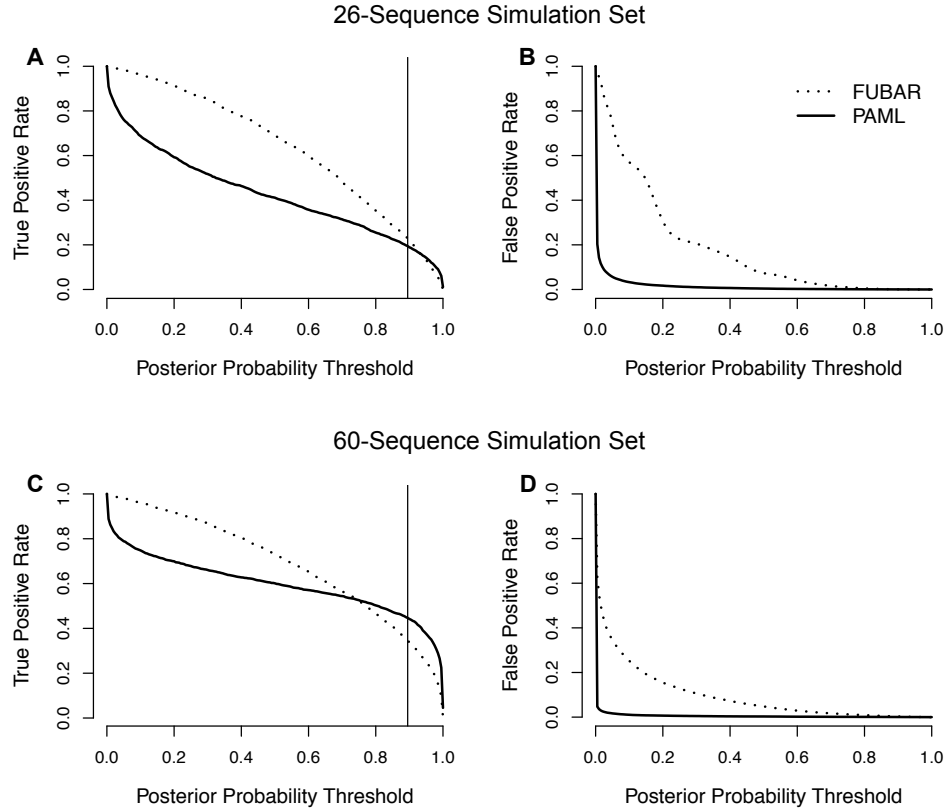


Figure 2: True and false positive rates recovered from FUBAR and PAML analysis with unfiltered alignments, averaged across the 26-sequence and the 60-sequence simulation sets, against the posterior probability threshold used to call sites as positively selected. A) TPR, 26-sequence set. B) FPR, 26-sequence set. C) TPR, 60-sequence set. D) FPR, 60-sequence set. As the number of sequences increased, PAML's TPR improved at lower posterior probabilities, while its FPR remained remarkably low across all posterior probabilities for both simulation sets. While FUBAR's performance did improve with the inclusion of more sequences, its overall TPR behavior did not change as dramatically as did PAML's.

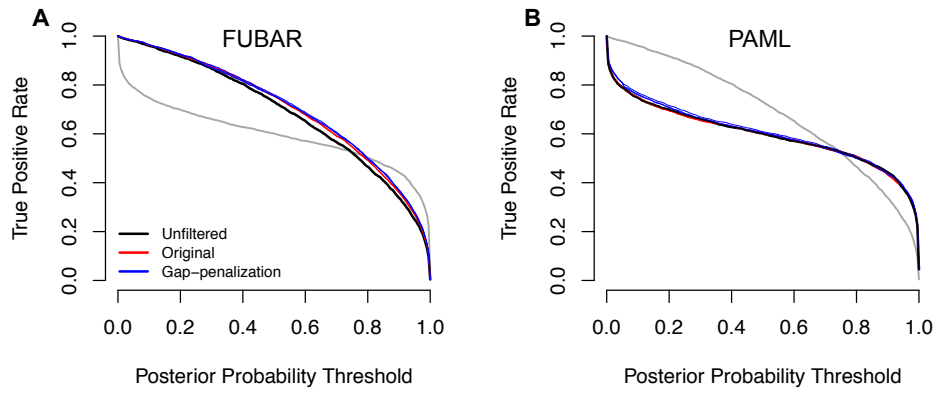


Figure 3: True positive rate against posterior probability threshold for calling positively selected sites, averaged across the 60-sequence simulation set. A) FUBAR in color with PAML results shown in grey. B) PAML in color with FUBAR results shown in grey. Filtered alignments behave similarly to unfiltered alignments across all posterior probabilities.

Table 1: Comparison of gap-penalization vs original normalization schemes on TPR of positive-selection inference.

Simulation Set	Method	Guidance	BMweights	PDweights
11 taxa	FUBAR	0.001 (0.98%)	-1.76×10^{-5} (-0.002%)	0.001 (0.95%)
	PAML	-2.68×10^{-4} (-0.31%)	2.57×10^{-5} (0.03%)	-0.001 (-1.50%)
26 taxa	FUBAR	0.0032 (1.38%)	0.00040 (1.75%)	0.0018 (0.77%)
	PAML	0.006* (3.02%)	0.007** (3.32%)	0.006* (2.89%)
60 taxa	FUBAR	0.010* (2.67%)	0.012** (3.30%)	0.005 (1.29%)
	PAML	-0.002 (-0.46%)	0.010 (2.23%)	0.008 (1.84%)
158 taxa	FUBAR	0.003 (0.79%)	0.002 (0.48%)	0.001 (0.28%)

NOTE.— Significance levels: $**P < 0.01$; $*P < 0.05$. Each column displays the magnitude of TPR difference between normalization schemes, represented as gap-penalization minus original, for each algorithm, respectively. Percentage changes from the original normalization scheme are shown in parentheses. All significance levels were corrected for multiple comparisons with the single-step method.

Table 2: Effect of alignment filtering on TPR of positive-selection inference.

Simulation Set	Method	Unfiltered TPR	Filtered TPR		
			GuidanceP	BMweightsP	PDweightsP
11 taxa	FUBAR	0.108	0.109 (1.04%)	0.110 (1.86%)	0.110 (1.37%)
	PAML	0.087	0.088 (0.49%)	0.088 (0.83%)	0.088 (0.79%)
26 taxa	FUBAR	0.229	0.232 (1.54%)	0.233 (1.83%)	0.233 (1.91%)
	PAML	0.194	0.204 (4.87%)*	0.203 (4.56%)*	0.203 (4.58%)*
60 taxa	FUBAR	0.345	0.379 (9.92%)**	0.377 (9.31%)**	0.375 (8.75%)**
	PAML	0.447	0.447 (0.19%)	0.440 (-1.43%)	0.445 (-0.30%)
158 taxa	FUBAR	0.374	0.388 (3.89%)**	0.387 (3.68%)**	0.387 (3.47%)**

NOTE.— Significance levels: $***P < 1 \times 10^{-5}$; $*P < 1 \times 10^{-4}$. *Unfiltered TPR*: average true positive rate (TPR) for unfiltered alignments; *Filtered TPR*: average true positive rate (TPR) for alignments filtered with each respective algorithm, with percent change from unfiltered alignment shown in parentheses. All significance levels are relative to the given simulation set’s unfiltered alignment. We detected no significant TPR differences among filters tested within sequence simulation sets. All significance levels were corrected for multiple comparisons with the single-step method.

Table 3: Effect of masking cutoff on TPR of positive-selection inference.

Simulation Set	0.5 TPR	0.9 TPR	Percent TPR Decrease
11 taxa	0.104	0.104	4.99%*
26 taxa	0.232	0.220	5.28%**
60 taxa	0.281	0.379	25.8%***
158 taxa	0.374	0.388	3.57%***

NOTE.— Significance levels: $***P < 1 \times 10^{-6}$; $**P < 1 \times 10^{-5}$; $*P < 1 \times 10^{-3}$.

0.5 TPR: average TPR for alignments masked at a cutoff of 0.5 with GuidanceP; *0.9 TPR*: average TPR for alignments masked at a cutoff of 0.9 with GuidanceP; *Percent TPR Decrease* average percent decrease in TPR recovered between alignments masked at cutoffs of 0.5 and 0.9. All significance levels were corrected for multiple comparisons with the single-step method.