Masking individual residues for positive selection inference has a minimal effect

Stephanie J. Spielman[1*] and Eric T. Dawson[1] Claus O. Wilke[1]

Address:
[1]Section of Integrative Biology and Center for Computational Biology and Bioinformatics. 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

Most nucleotide or amino acid sequence studies begin with building a multiple sequence alignment (MSA) with the goal of assigning homology across positions of these diverged sequences. Although a critical step in nearly sequence evolution studies, MSAs may represent a substantial source of error in subsequent applications, ranging from phylogenetic reconstruction to evolutionary rate inference. Motivated by this issue, many have suggested methods which can identify putatively poorly aligned regions in MSAs so that researchers can remove these regions from analyses in an effort to reduce error. One recent method, known as Guidance, generates position-based confidence scores in a given MSA using a bootstrap approach. Some studies have suggested that applying a Guidance filter to protein-coding sequence alignments can improve accuracy in positive selection inference. However, the specific circumstances under which Guidance might offer improvements remain unclear. To elucidate how the Guidance filter behaves in realistic protein-coding sequences, we have reimplemented the Guidance software with several statistical improvements, including methods to phylogenetically correct confidence scores. We find that, overall, neither the original Guidance nor our phylogenetically-corrected version substantially improves positive selection inference. Thus, we cannot equivocally advocate the use of a Guidance-based filter for positive selection inference.

Introduction

Constructing a multiple sequence alignment (MSA) 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 MSA quality can significantly hinder accuracy in such downstream analyses (Jordan and Goldman 2011; 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 (Privman et al. 2012; Schneider et al. 2009; Fletcher and Yang 2010). As a consequence, many have advocated applying an alignment filter to MSAs. 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 a progressive alignment guide tree. Using the resulting scores, users can mask (i.e. replace with an ambiguous character such as "?" or "N") positions which 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 2011; Privman et al. 2012).

However, while Privman et al. Privman et al. (2012) found dramatic improvements in positive selection inference when applying the Guidance filter, Jordan and Goldman's Jordan and Goldman (2011) comprehensive study on alignment methods and filtering found that Guidance modestly, if at all, affected this inference. 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 found that Guidance significantly 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 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 changes to positive selection inference, and any benefits recovered were of extremely small magnitude. Thus, we cannot unequivocally advocate the use of such filters when inferring positive selection in protein-coding sequences.

Results

Guidance reimplementation 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 reimplementation, 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 proposed 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 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 hemagluttinin (HA), a protein well known to contain positively selected regions (Meyer and Wilke 2012). We then processed the unaligned sequences with our Guidance reimplementation 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), and masked positions with scores below 0.5 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). 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 has limited utility

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. To this end, we build mixed-effects linear models for each sequence simulation set, with simulation count as the random effect to account for the paired structure of our analysis (as in, each alignment was filtered with multiple strategies, rendering them dependent), for Fubar and Paml separately. 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 rates were similar to those recovered by (Jordan and Goldman 2011).

We first compared the two normalization schemes, original and gap-penalization, to one another using mixed effects models with TPR as the response variable, simulation count as a random effect, and normalization scheme as a fixed effect. Only filtered alignments were considered in these models, results from which are seen in Table 1. In all instances in which we detected 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 results from gap-penalization algorithms (GuidanceP, BMweightsP, PDweightsP) to those of unfiltered alignments with a second series of mixed-effects models. These models included TPR as the response variable, simulation count as a random effect, 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 2 displays ROC curves for positive

selection inference with Fubar and Paml. Figure 2B highlights the effect of alignment filtering with Fubar and Figure 2C with Paml; for each method, respectively, 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 roughly 0.009. 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.

Posterior Probability Threshold Calling Positive Selection is, in fact, a thing. OR Paml and Fubar are different methods

As our models' results were highly sensitive to the posterior probability threshold chosen to assign sites as positively selected, we visually assessed how selecting a different threshold would influence filtering. Figure shows, for unfiltered alignments, how this threshold influences TPR and FPR for both Fubar and Paml, averaged across results from the 60-sequence simulation set. While the TPR for Fubar decreased smoothly across increasing posterior probabilities, TPRs recovered from Paml analyses were more robust to changes in posterior probability; sharp changes in TPR occurred at very low and very high posterior probabilities, but TPR was less sensitive to this threshold between $0.2 - 0.8$. Similar to its TPR behavior, Fubar's FPR curve slowly trends downwards, reaching a zero FPR when ≥ 0.80. Paml, alternatively, reached an FPR of zero essentially immediately. These trends evidenced the different approaches that Fubar and Paml employ to detect positive selection; while Paml's M8 aims to identify precise dN/dS value at each site in an alignment, Fubar merely approximates whether a site is positively selected but does not attempt to assign a point estimate of dN/dS to each position. These distinct methodologies have some implications for alignment filters, as well. As seen in Figure 3, which shows TPR for both Fubar and Paml at posterior probability cutoffs that a typical study would employ. The filtered alignments, as processed with Fubar, were clearly above the unfiltered alignment line, whereas, when processed with Paml, they were indistinguishable from the unfiltered alignment. **but this is the same as the model, so I'm not sure that placing the text here is reasonable.**.

We conducted an ROC analysis to broadly compare accuracy across filters and between Fubar and Paml for the 60-sequence simulation set (Figure 2). The ROC curves suggested that filtering alignments and analyzing with Paml might substantially increase TPR in positive selection inference, but that analysis with Fubar would yield minimal, if any, benefit. Interestingly, our models revealed the opposite trend (filtering did improve positive selec-

tion inference with Paml but not with Fubar). This discrepancy reflected the fact that ROC curves did not control for the posterior probability threshold employed to call sites as positively selected. In other words, a given point on an ROC curve could easily correspond to different posterior probability thresholds for filtered and unfiltered alignments. Thus, while useful in assessing theoretical performance, ROC curves did not capture the reality of data analysis and might actually have been misleading.

paragraph of ambiguous placement, and also of ambiguous writing quality: Interestingly, for unfiltered alignments, Fubar slightly outperformed Paml for the 11 and 26-sequence simulation sets, but Paml strongly outperformed Fubar for the 60-sequence simulation set. While we did not process alignments with Paml for the 158-sequence set, it was likely that Paml would demonstrate much higher accuracy than Fubar. However, Paml took anywhere from one day to one week to analyze each 60-sequence alignment, whereas Fubar took under 15 minutes. While the Paml M8 model may have been more accurate, the runtimes were orders of magnitude higher than for Fubar. One shudders to think of how long the 158-sequences would have taken.

Discussion

all of this is old, probably written early December. Our comprehensive study of alignment filtering with Guidance-based methods indicated that, while masking individual sites rarely hinders positive selection inference, neither does it substantially improve inferences when analyzing protein sequences at realistic divergence levels. These results mirror those made by Jordan and Goldman's (Jordan and Goldman 2011) study on alignment methods and filtering. While the gap-penalization scheme we have proposed did perform significantly better than did the original normalization scheme, the magnitude of this improvement was minimal. Additionally, that the weighted algorithms did not improve upon the original Guidance algorithm may indicate the minimal benefits that filtering in this manner produces at all. Were Guidance to offer robust improvements when detecting positively selected sites, one might expect that the more statistically controlled approach would boost the method's performance. However, as we have found that the method itself does not dramatically, if at all, influence positive selection inferences, it is not entirely unexpected that improving the algorithm does not help, either. Based on our results, then, we cannot unequivocally recommend filtering alignments in the manner presented here.

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 100 bootstrapped alignment replicates, each of which is used to create a bootstrapped tree

in FastTree2 (Price et al. 2010). We then use these 100 trees as guide trees to create 100 new perturbed alignments, which we subsequently compared to the reference alignment to generate a confidence score for each residue.

Scoring Algorithms

As additionally implemented two new scoring algorithms which incorporating 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). 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 calculated positional confidence scores for each of the n bootstrap alignment as follows. A raw score, S, for a given residue in row i, column j of the reference alignment is calculated as

$$S_{ij} = \sum^{k} I^{j}_{ik} s^{j}_{ik} \tag{1}$$

, where k represents all rows in column j which are not gaps. In this formula, the indicator function

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

served to compare the bootstrap and reference alignment residue pairings. We calculated s^{j}_{ik} based on the given scoring algorithm:

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

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

We then summed positional scores S_{ij} determined from each bootstrap alignment and normalized them to yield a final score $\widetilde{S}_{ij}$ at each residue in the reference alignment. We used two different normalization schemes to this end. The first scheme, given by

$$\widetilde{S}_{ij} = \frac{\sum^{n} S_{ij}}{n \sum^{k} S_{kj}}, \tag{4}$$

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

$$\widetilde{S}_{ij} = (\sum^{n} S_{ij}) \Big/ \begin{cases} n \times (l-1) & \text{if Guidance} \\ n \sum d_p(i,l) & \text{if BMweights} \\ n w_i & \text{if PDweights} \end{cases}, \tag{5}$$

where l represented all rows (including gaps) in column j **l is used in two diff ways in the above formula – the "if Guidance" one is a number and the "if BMweights" one is a counter. Thoughts?**. This second normalization scheme inherently gave 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.

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 hemagluttinin (HA) influenza protein. To derive these parameters, we aligned 1028 HA protein sequences with mafft linsi (Katoh 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 kappa ($\kappa = TI/TV$) 5.3 for all simulations. We additionally set the state codon frequencies for our simulations according to those directly calculated from 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) rates 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 mafft linsi (Katoh et al. 2002, 2005) within the context of our Guidance reimplementation. 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 cutoff of 0.5. We inferred positive selection for every condition using both Fubar (Murrell et al. 2013) with default parameters and the Paml's M8 model, specifying $F3 \times 4$ codon frequency and "cleandata = 0" in the control file (Yang 2007). We used the phylogeny built during the Guidance alignment procedure as the input phylogeny for selection inference. Therefore, all alignments derived from the same unfiltered alignment were processed with identical phylogenies to remove any potential bias. 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 be differently aligned relative to the true simulated alignment, we adopted a consensus method to compare alignments we constructed to the true simulations. We required that at least 50% of the residues present in a true alignment column be present in an inferred alignment column. If this condition was met, we considered the true alignment columns dN/dS selection status (negative, neutral, or positive) to be the true value for the given inferred column. We considered sites positively selected if the posterior probability of $(dN/dS > 1) \geq 0.9$.

Statistics were performed using in-house python and R scripts. 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.

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.

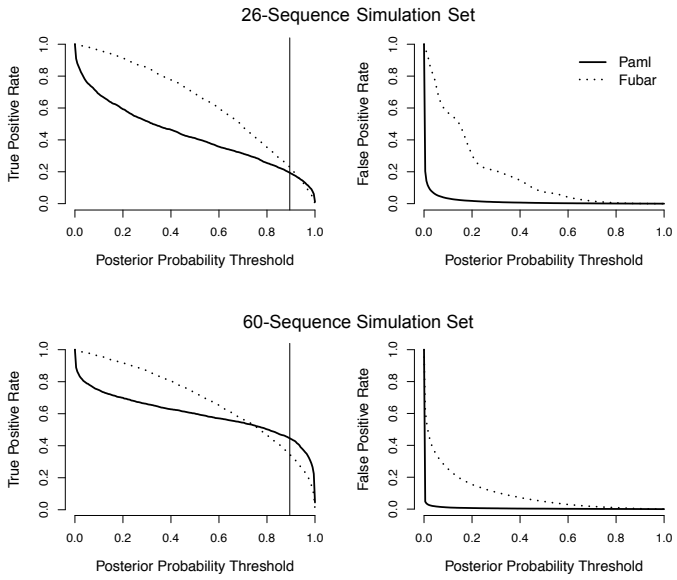

Figure 1: TPR (left) and FPR (right) for Paml and Fubar unfiltered alignments, averaged across the 60-sequence simulation set. Fubar is shown as a solid line and Paml as a dashed line.

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. 2011. 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.

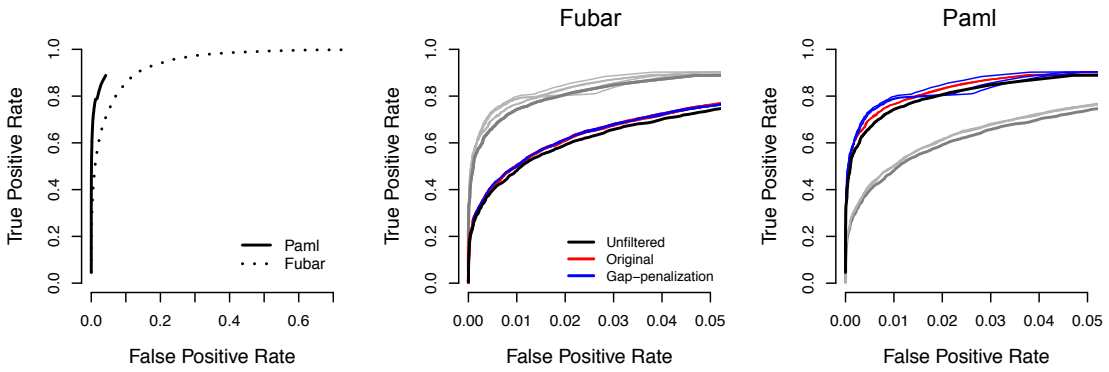

Figure 2: ROC curve as averaged across 60-sequence simulation set. Left: paml (dashed) and fubar (solid) unfiltered. Middle: zoomed in fubar with paml greyed out. Right: zoomed in paml with fubar greyed out. In middle and right, unfiltered alignment black. Gap-penalization in blue, original normalization in red.

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.

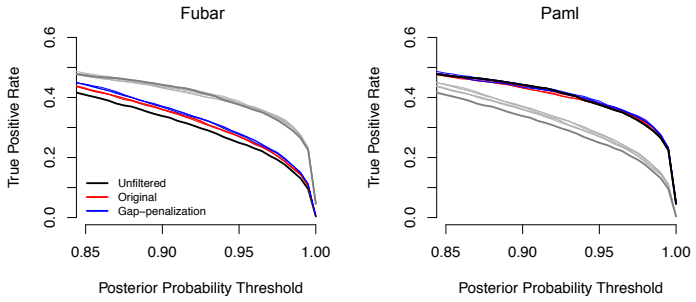

Figure 3: Zoomed in tpr on y-axis, pp cutoff on x axis. Left: fubar with paml greyed out. Right: paml with fubar greyed out. Unfiltered alignment black. Gap-penalization in blue, original normalization in red.

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.

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

Simulation Set	Method	d
11 taxa	Fubar	0.0007
	Paml	-0.0005
26 taxa	Fubar	**0.0018***
	Paml	**0.0061****
60 taxa	Fubar	**0.0089****
	Paml	0.0054
158 taxa	Fubar	**0.002****

NOTE.— Significance levels: $^{**}P < 1 \times 10^{-6}$; $^{*}P < 0.05$. d: Magnitude of TPR difference between normalization schemes, represented as gap-penalization minus original. All significance levels were corrected for multiple comparisons with the single-step method.

Table 2: Effect of alignment filtering on the true positive rate of positive selection inference.

Simulation Set	Method	Unfiltered	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	0.233	0.233
	Paml	0.194	**0.204***	**0.203***	**0.203***
60 taxa	Fubar	0.345	**0.379****	**0.377****	**0.375****
	Paml	0.447	0.447	0.440	0.445
158 taxa	Fubar	0.374	**0.388****	**0.387****	**0.387****

NOTE.— Significance levels: $^{**}P < 1 \times 10^{-5}$; $^{*}P < 1 \times 10^{-4}$. *Unfiltered*: average true positive rate (TPR) for unfiltered alignments; *GuidanceP*: average true positive rate (TPR) for alignments filtered with GuidanceP algorithm; *BMweightsP*: average TPR for alignments filtered with BMweightsP algorithm; *PDweightsP*: average TPR for alignments filtered with PDweightsP algorithm. 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.