Supplementary Material for "Limited utility of residue masking for positive-selection inference"

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

[1]Department of Integrative Biology, Center for Computational Biology and Bioinformatics, and Institute of Cellular and Molecular Biology.
The University of Texas at Austin, Austin, TX 78712, USA.

[*]Corresponding author
Email: stephanie.spielman@utexas.edu

Contents

1 Materials and Methods

Guidance Software Reimplementation and Scoring Algorithms

Our reimplemented Guidance is written in Python and C++ and is described in detail in SI. Following the algorithm set forth in Penn et al. (Penn et al. 2010), we first create a reference multiple sequence alignment (MSA) using a user-specified progressive MSA 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 MSA 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 MSAs, which we subsequently compare to the reference MSA to generate a confidence score for each residue. Users can specify options for their aligner and phylogeny reconstruction method as desired.

Before calculating confidence scores, a phylogeny is built from the reference MSA. Our program includes functionality to build this phylogeny using either FastTree2 (Price et al. 2010) or RAxML (Stamatakis 2006). Two types of phylogenetic weights can be calculated from this tree. 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 MSAs as follows. A raw score, S_{ij}, for a given residue in row i, column j of the reference MSA is calculated as

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

where $R_{\text{ng}}^{(j)}$ represents the set of rows in column j which are not gaps. We calculate s_{ik} according to the given scoring algorithm:

$$s_{ik} = \begin{cases} 1 & \text{if Guidance} \\ w_i w_k & \text{if BMweights} \\ d_p(i,k) & \text{if PDweights} \end{cases}, \tag{2}$$

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. The indicator function

$$I_{ik}^{(j)} = \begin{cases} 1 & \text{if reference MSA residue pair } (i,k)^{(j)} \text{ is present in bootstrap MSA} \\ 0 & \text{if reference MSA residue pair } (i,k)^{(j)} \text{ is absent in bootstrap MSA} \end{cases} \tag{3}$$

serves to compare the bootstrap- and reference-MSA residue pairings.

We then sum positional scores S_{ij} determined from each bootstrap replicate n. We normalize these scores across bootstrap replicates to yield a final score $\widetilde{S}_{ij}$ for each residue in the reference MSA. We use two different normalization schemes: original Guidance (defined in Penn et al. (2010)) and a novel gap-penalization scheme. These normalization schemes are given by

$$\widetilde{S}_{ij} = \sum_n S_{ij}(n) \Bigg/ \begin{cases} \sum\limits_n \sum\limits_{k \in R_{\text{ng}}^{(j)}} s_{ik}(n) & \text{if original Guidance} \\ \sum\limits_n \sum\limits_{k \in R_{\text{all}}^{(j)}} s_{ik}(n) & \text{if gap-penalization} \end{cases}, \tag{4}$$

where $R_{\text{all}}^{(j)}$ represents all rows in column j, including gaps, the sums over n run over all N replicates, and $S_{ij}(n)$ and $s_{ik}(n)$ each represent those respective quantities for bootstrap replicate n. By considering all rows instead of just rows that are not gaps, the gap-penalization scheme will naturally assign lower scores to highly gapped columns. We refer to the algorithms normalized by the original Guidance 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 original 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 sequences according to two distinct sets of evolutionary parameters. The first selective profile was derived from H1N1 hemagluttinin (HA) influenza protein, and the second selective profile was derived from HIV-1 envelope protein GP41, which contains only the cytosolic and tail regions.

To derive parameters for the HA selective profile, we aligned 1038 HA protein sequences collected from the Influenza Research Database (http://www.fludb.org) with MAFFT, specifying the "–auto" flag, (Katoh et al. 2002, 2005) and then back-translated to a codon MSA using the original nucleotide sequence data. We generated a phylogeny from this codon MSA in RAxML (Stamatakis 2006) using the GTRGAMMA model. Using the codon MSA and phylogeny, we inferred evolutionary parameters as described (Spielman and Wilke 2013). We used a REL (random effects likelihood) method (Nielsen and Yang 1998) using the HyPhy software (Kosakovsky Pond et al. 2005), with five dN/dS 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 dN/dS had a mean of 0.37. 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 and set state codon frequencies equal to those directly calculated from the HA MSA.

To derive parameters for the GP41 selective profile, we aligned 2192 sequences from the LANL database (http://HIV.lanl.gov), again with MAFFT. We generated an amino-acid phylogeny from this MSA using FastTree2 under the WAG model. We subsequently inferred a dN/dS distribution from this data in FUBAR. Note that we only used those dN/dS values which were below 6, as FUBAR's approximate approach yielded estimates well over 1000 when dS values were small. Again, we binned the resulting dN/dS distribution into equally spaced categories for specification in Indelible. The resulting dN/dS distribution had a mean of 0.89. We inferred a value for κ by selecting from this large dataset a random set of 150 sequences, which we aligned with linsi (Katoh et al. 2002, 2005) and inferred a phylogeny using FastTree2 (Price et al. 2010), specifying the options "-wag -fastest." We then inferred κ, which was calculated as 3.36, using the GY94 model in HyPhy. For all simulations under the GP41 selective profile, we fixed κ at 3.36 and set state codon frequencies equal to those directly calculated from the GP41 MSA.

We simulated 100 alignments across four different real gene trees each, for both selective profiles, yielding a total of 800 simulated MSAs. 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 (http://treebase.org). We set all insertion-deletion (indel) rates at 0.05, motivated by studies demonstrating that indel events occur at a rate 5% of the substitution rate in mammalian genomes (Cooper et al. 2004), and we set all average sequence lengths to 400 codons.

MSA Construction and Positive-Selection Inference

We build all MSAs using MAFFT L-INS-I (linsi) (Katoh et al. 2002, 2005) within the context of our Guidance reimplementation. Phylogenies used to calculate phylogenetic weights for the BMweights and PDweights algorithms were constructed in RAxML, specifying "PROTGAMMAWAG" as the model of sequence evolution (Stamatakis 2006). In addition to an unfiltered MSA, we generated six filtered MSAs (one for each filtering algorithm and each normalization scheme), masking residues with scores ≤ 0.5 with "?". To investigate potential biases introduced by the scoring threshold, we also masked residues below the scoring cutoffs of 0.3, 0.7, and 0.9 for MSAs constructed with the Guidance and GuidanceP filters. Only MSAs simulated according to HA parameters were included for this analysis.

We inferred positive selection using both the PAML M8 model (Yang 2007) and FUBAR (Murrell et al. 2013), which is implemented in the HyPhy (Kosakovsky Pond et al. 2005) software package. For inference with PAML, we specified the $F3 \times 4$ codon frequency model and "cleandata = 0" in the control file. For FUBAR inference, we used mostly default parameters, except when specifying grid dimensionality. As neither Indelible nor PAML simulates sequences without dS variation, we specified that FUBAR only consider dN variation, in order to make results from FUBAR and PAML fully comparable. We additionally specified 100 grid points to account for the reduced grid dimensionality caused by ignoring dS variation. Phylogenies specified for positive-selection inference were those constructed during the Guidance MSA procedure when deriving phylogenetic weights. All filtered MSAs derived from a unfiltered MSA were processed with identical phylogenies to avoid potentially confounding effects of different tree topologies. 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 MSA to its respective true MSA'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 MSA, we constructed a map from each unfiltered MSA to its respective true MSA. For this map, we selected the sequence in the unfiltered MSA with the fewest number of gaps and mapped its non-gap sites to the corresponding residue position in the true MSA. This strategy ensured that the most sites possible were included when calculating true positive rates. Importantly, all filtered variants of a given unfiltered MSA used the same map to the true MSA. We considered sites positively selected if the posterior probability of $(dN/dS > 1)$ was ≥ 0.9. For all alignments, we calculated a true positive rate (TPR) according to,

$$TPR = \frac{n_{TP}}{n_{TP} + n_{FN}} \tag{5}$$

, where n_{TP} is the number of true positives and n_{FN} is the number of false negatives, and a false positive rate (FPR) according to,

$$FPR = \frac{n_{FP}}{n_{FP} + n_{TN}} \tag{6}$$

, where n_{FP} is the number of false positives and n_{TN} is the number of true negatives.

All statistics were conducted using Python and R. Linear modeling was conducted using the R package lme4 (Bates et al. 2012). We inferred effect magnitudes and significance, which we corrected for multiple testing,g using the R multcomp package's glht() function with default settings (Hothorn et al. 2008). All code used in this study is available at https://github.com/clauswilke/alignment_filtering.

Analysis of MSA Filtering Threshold

When filtering MSAs 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, as previously done by Jordan and Goldman (2012). However, Privman et al. (2012) filtered all sites which scored below a threshold of 0.9. Therefore, it was possible that selecting a different threshold would have yielded different results, so we analyzed how changing this threshold might impact our findings. For this analysis, we considered only the Guidance and GuidanceP scoring schemes for the HA selective profile. Using the same position confidence scores previously generated, we masked all alignments at the additional cutoffs of 0.3, 0.7, and 0.9 and inferred positive selection with FUBAR and PAML as described.

We fit 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 each. In general, the threshold of 0.3, 0.5, and 0.7 performed statistically indistinguishably. The single exception to this finding occurred for the 60-sequence simulation set, as inferred with FUBAR, for which the cutoffs of 0.3 and 0.5 each performed marginally better than did the 0.7 cutoff ($P < 0.01$ and $P < 0.05$, respectively). However, as shown in Table S1, the cutoff of 0.9 tended to yield worse performances than did the 0.5 threshold. It is, however, important to note that PAML was generally more robust to changes in masking threshold than was FUBAR, and that GuidanceP was influenced more by changing the masking threshold than was Guidance. Likely the latter observation resulted from the fact that the gap-penalization algorithms mask far more sites than do algorithms which use the original normalization scheme, ultimately resulting in an excessive amount of information removed at a 0.9 masking threshold.

Therefore, we do not expect that our results were biased by our selection of a 0.5 masking threshold. Alternatively, using a stringent 0.9 cutoff likely would have excessively sacrificed power for positive-selection inference in several cases.

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.

Cooper G, Brudno M, Stone E, Dubchak I, Batzoglou S, Sidow A. 2004. Characterization of evolutionary rates and constraints in three mammalian genomes. Genome Research 14:539 – 548.

Edgar R C. 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. 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.

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.

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.

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.

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.

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.

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.

2 Supplementary Tables

Table S1 is given below. Tables S2 and S3 are given in separate excel files.

Table S1. Effect of masking cutoff on mean TPR of positive-selection inference for filtered MSAs.

		Guidance			GuidanceP		
Taxa	Method	0.5 TPR	0.9 TPR	Percent TPR Change	0.5 TPR	0.9 TPR	Percent TPR Change
11	FUBAR	0.085	0.082	**-4.22%***	0.085	0.079	**-6.60%****
	PAML	0.082	0.079	-2.59%	0.081	0.074	**-8.38%****
26	FUBAR	0.229	0.227	-0.87%	0.226	0.215	**-5.21%*****
	PAML	0.178	0.179	0.67%	0.183	0.175	**-4.33%****
60	FUBAR	0.479	0.458	**-4.43*****	0.464	0.332	**-28.5%****
	PAML	0.342	0.327	-4.19%	0.337	0.256	**-24.1%****
158	FUBAR	0.467	0.463	-0.73%	0.468	0.452	**-3.44%*****

NOTE.— Significance levels: $^{***}P < 10^{-4}$; $^{**}P < 10^{-3}$; $^{*}P < 10^{-2}$. 0.5 TPR: average TPR for MSAs masked at a cutoff of 0.5; 0.9 TPR: average TPR for MSAs masked at a cutoff of 0.9; Percent TPR Decrease: average percent decrease in TPR recovered between MSAs masked at cutoffs of 0.5 and 0.9. Linear models were conducted with the lme4 package in R (Bates et al. 2012). All significance levels were corrected for multiple comparisons using the multcomp package in R (Hothorn et al. 2008).

3 Supplementary Figures

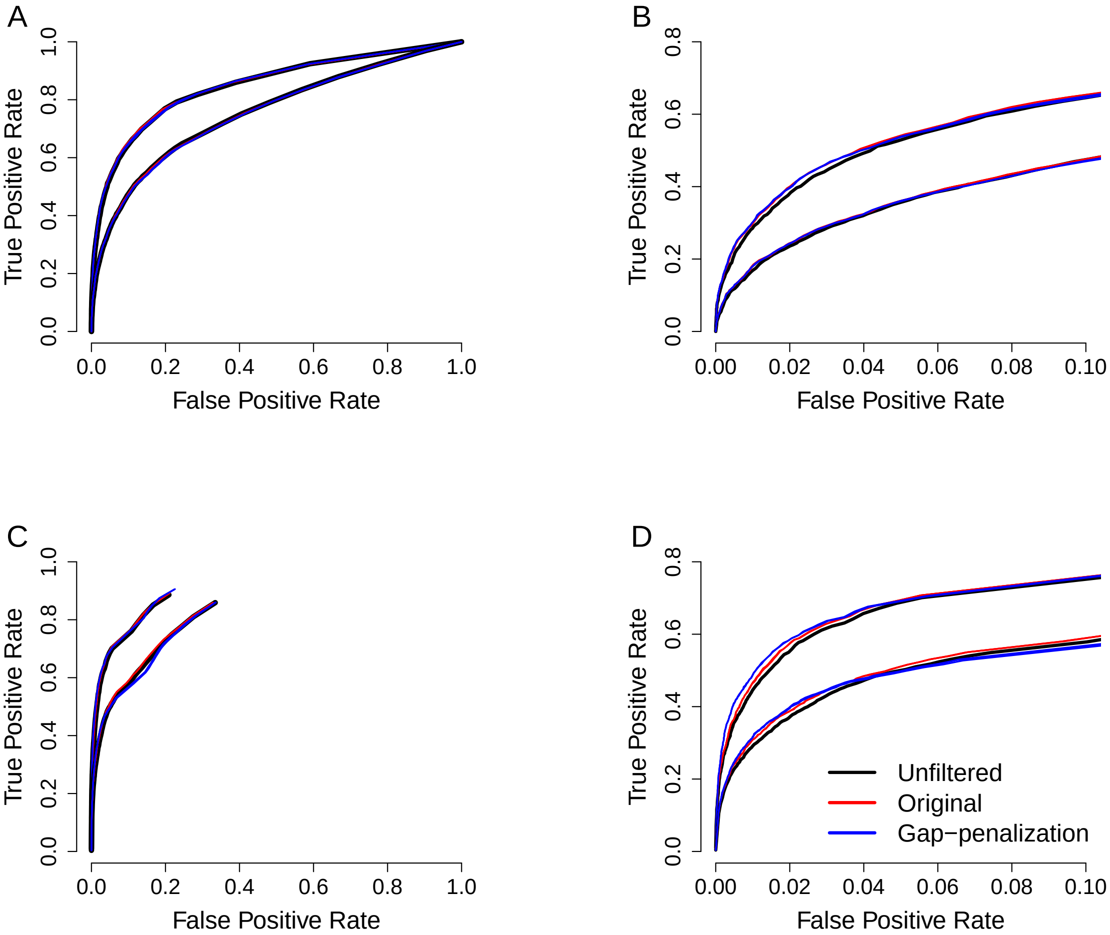

Figure S1. ROC curves for positive-selection inference with PAML M8, averaged within each simulation set. For all panels, the top curve represents results from the HA selective profile, and the bottom curve represents results from the GP41 selective profile. The left-hand panels display the entire ROC curves, while the right-hand panels display only the region of the curves with relatively low FPRs. Note that, for the full ROC curves, methods only achieved FPR levels shown. (A-B) ROC curves for the 11-sequence simulation sets. (C-D) ROC curves for the 26-sequence simulation sets.

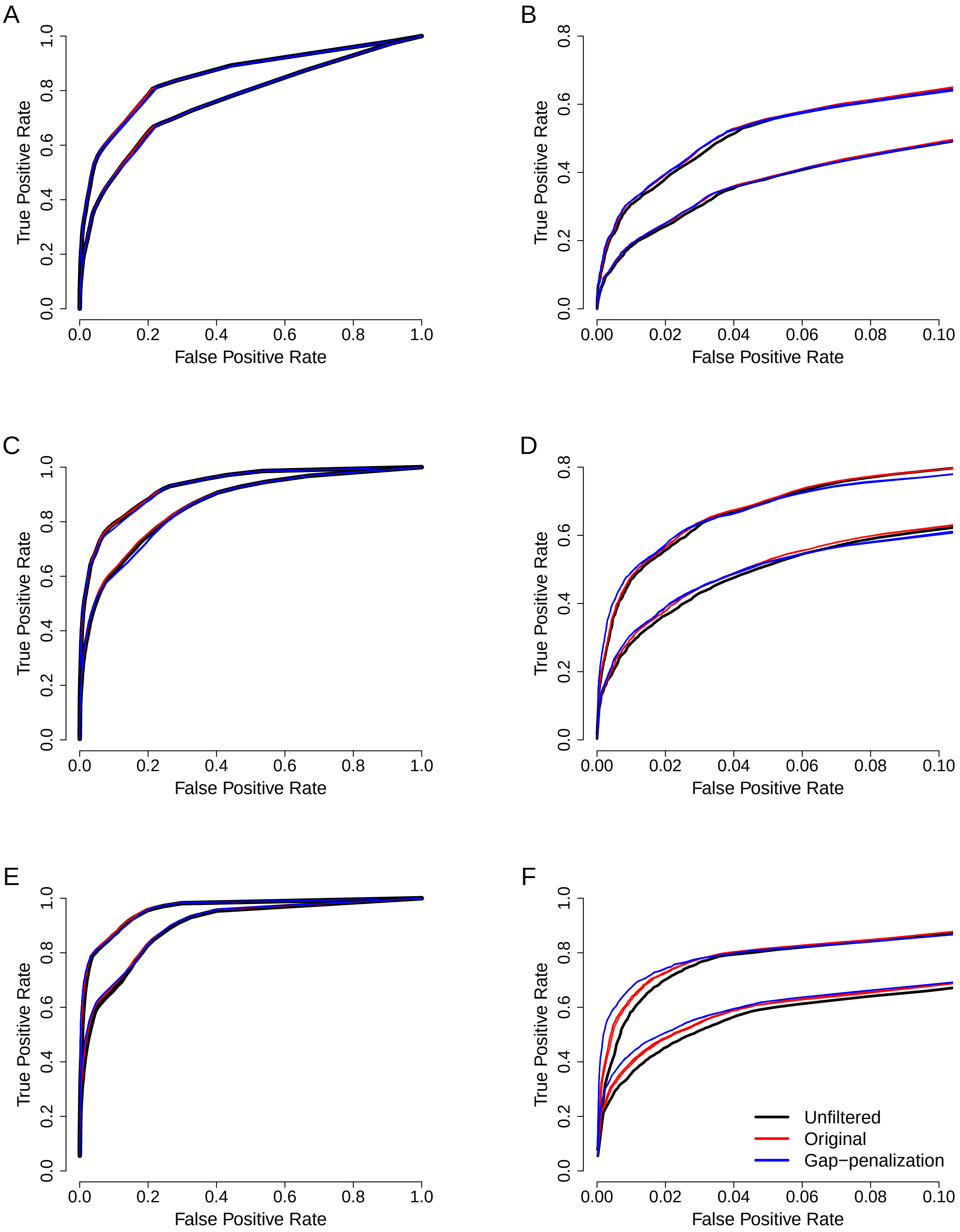

Figure S2. ROC curves for positive-selection inference with FUBAR, averaged within each simulation set. For all panels, the top curve represents results from the HA selective profile, and the bottom curve represents results from the GP41 selective profile. The left-hand panels display the entire ROC curves, while the right-hand panels display only the region of the curves with relatively low FPRs. (A-B) ROC curves for the 11-sequence simulation sets. (C-D) ROC curves for the 26-sequence simulation sets. (E-F) ROC curves for the 158-sequence simulation sets.