Evolution of viral genomes: Interplay between selection, recombination and other forces

Stephanie J Spielman[1], Steven Weaver[1], Stephen D. Shank[1], Brittany Rife Magalis[1], Michael Li [1], and Sergei L Kosakovsky Pond[1,2]

[1]*Institute for Genomics and Evolutionary Medicine, Temple University, Philadelphia, USA*
[2]*spond@temple.edu*

Abstract

Natural selection is a fundamental force shaping organismal evolution, as it both maintains function and enables adaptation and innovation. Viruses, with their typically short and largely coding genomes, experience strong and diverse selective forces, sometimes acting on timescales that can be directly measured. These selection pressures emerge from an antagonistic interplay between rapidly changing fitness requirements (immune and antiviral responses from hosts, transmission between hosts, or colonization of new host species) and functional imperatives (the ability to infect hosts or host cells and replicate hosts). Many computational methods for quantifying such evolutionary forces using molecular sequences, dating back to the 1980s, were initially applied to the study viral pathogens, largely because strong selective forces are easier to detect, and because of clear biomedical relevance of such analyses. Recent commoditization of affordable high-throughput sequencing has made it possible to generate truly massive genomic samples, on which powerful and accurate methods can yield a very detailed depiction of when, where, and (sometimes) how, viral pathogens respond to various selective forces.

Here, we present recent statistical developments and state-of-the-art methods to identify and characterize these selection pressures from protein-coding sequence alignments and phylogenies. Methods described here can reveal critical information about various evolutionary regimes, including whole-gene selection, lineage-specific selection, and site-specific selection acting upon viral genomes, while accounting for confounding biological processes, such as recombination and variation in mutation rates.

Introduction

Natural selection is a powerful evolutionary force that shapes genomes of all living organisms. In typical applications, a single gene represented by isolates from different individuals (e.g. sequences from many HIV-1–infected hosts), or different hosts (e.g.

primate lentiviruses) is considered for selection analyses. Given an alignment of homologous gene sequences, the strength of natural selection acting on a given gene or genes can be measured in a phylogenetic context using *codon models* (Anisimova and Kosiol, 2009; Delport et al., 2009).

In this context, selection is typically measured using dN/dS (also referred to as ω, or Ka/Ks), which represents the ratio of the non-synonymous evolutionary rate (dN) to the synonymous evolutionary rate (dS). The synonymous evolutionary rate is used to provide a baseline rate of neutral evolution because the average selective effect of a synonymous substitution is assumed to be negligible compared to the effect of a non-synonymous substitution [1]. The selective regime can be deduced by establishing, with a degree of statistical confidence, that dN/dS differs from unity, i.e., the neutral expectation. Diversifying, balancing, or (sometimes) directional selection yields $dN/dS > 1$, whereas purifying selection effects $dN/dS < 1$. Comparative methods for selection detection **estimate** dN/dS, or dS and dN separately, and perform a statistical test to establish which side of the neutral expectation the inferences fall on. As with any statistical procedure applied to finite data, each inference can be a false positive or a false negative, although methods typically take care to control the rates of both.

While the question ‘‘Is this gene under selection?" is an obvious one, the nearly universally applicable answer to this question is "yes". That is because a functional gene is (or has been) subject to some form of selection, e.g., negative selection to maintain essential features. On the other extreme is the question that has an immediate biological significance: *"Is changing a leucine to an arginine at position 209 in gene X along a specific branch in the phylogeny adaptive?"*. Without additional information, such as a carefully experimentally measured fitness impact of introducing said substitution, current comparative sequence approaches cannot answer this question. Indeed, such a scenario presents a sample size of one, which cannot be statistically meaningful.

In this chapter, we present a collection of statistical methods, each of which is designed to carefully address a biological question somewhere on the spectrum between the two extremes: sufficiently specific to be interesting, yet general enough to be answerable based on only on the evolutionary history of homologous sequences. We will not discuss the technical details of codon substitution methods here (for details, please see one of the excellent available reviews Yang (2006); Anisimova and Kosiol (2009); Delport et al. (2009), or the primary methods papers including Goldman and Yang (1994); Muse and Gaut (1994); Nielsen and Yang (1998); Kosakovsky Pond and Muse (2005)). Instead, we present each method operationally (*"How and when does one use this method?"*), by answering the following questions:

1. What biological question is the method designed to answer?
2. What are the recommended applications?
3. How is this question posed in terms of dN/dS and which statistical test is used to establish significance?
4. How to interpret positive and negative test results?
5. Rules of thumb for when this method is likely to work well, and when it is not.

[1] We note that there are a variety of well-documented situations where synonymous substitutions can have strong effects on fitness (Hershberg and Petrov, 2008; Plotkin and Kudla, 2011)

We conclude by discussing two evolutionary processes (recombination and synonymous rate variation) which are important to model, both in their own right, and because ignoring their effects could mislead selection inference tools.

Recombination plays a key role in the evolution of many viral pathogens. For instance, major pandemic strains of the Influenza A virus (IAV) have arisen through segmental reassortment, which can be thought of as intergenic, or gene preserving, recombination (note that because recombination is intergenic in influenza, the prior analyses we perfomed here on IAV H3 are not confounded). For example, the swine origin HIN1 virus has undergone at least two reassortment events, and carries genes from three different ancestral IAV lineages (Smith et al., 2010).

Moreover, in HIV-1, each viral particle packages two RNA genomes. During reverse transcription (RT), the RT enzyme switches between two RNA templates at rates as high as 2×10^3 per nucleotide per replication cycle (Schlub et al., 2010), creating recombinant DNA templates, which in turn can give rise to recombinant progeny. If a single cell is infected with multiple divergent HIV-1 viruses (this can occur in up to 10% of infected hosts, e.g., see Smith et al. (2009), depending on a variety of factors), then it is possible that resulting recombinants will found distinct and novel viral lineages. Molecular epidemiology of HIV-1 is replete with examples of such lineages, termed Circulating Recombinant Forms (CRFs), with over 60 characterized to date (Taylor et al., 2008).

The viral type and species strongly influences how frequently recombination occurs. For example, Chare and Holmes (2006) found evidence of recombination in 40% of plant RNA genomes that they had examined, but in fewer than 10% of negative sense RNA viruses (Chare et al., 2003). Apart from its importance in generating novel or removing deleterious genetic diversity and accelerating evolution (Worobey and Holmes, 1999), recombination has a strong effect on many practical aspects of evolutionary analyses (Posada et al., 2002).

The rate of synonymous codon evolution (represented by dS in the context of codon models) has long been modeled as constant across sites. Yet, there is ample evidence that this rate varies across species, genes, and even intragenic positions. In particular, intragenic synonymous rate variation has been identified across domains of life (Hershberg and Petrov, 2008; Plotkin and Kudla, 2011; Lynch et al., 2016) and can arise from a variety of evolutionary processes, including selection on mRNA secondary structure (Chamary and Hurst, 2005), gene expression (Drummond and Wilke, 2008), GC-biased gene conversion (Harrison and Charlesworth, 2011), and other neutral mutation processes. For example, even the genomic context of a given nucleotide can influence its mutation rate; indeed, experimental work has shown that GC-neighboring sites can feature up to a 75-fold increase in mutation rate (Sung et al., 2015; Lynch et al., 2016). In addition, the synonymous rate at certain sites may be elevated due to the mutational vulnerability of the non-template DNA strand during transcription (Lynch et al., 2016). In viruses, specifically, strong signals of intragenic synonymous rate variation have been shown to result from the presence of overlapping genomic elements. When a genic region overlaps transcription factor binding sites or microRNAs, each genomic element is subject to unique selection pressures which can ultimately influence the mutation, and hence synonymous, rate (Sealfon et al., 2015). We have previously shown that, due to the many biological confounding factors discussed above, discounting dS variation as in modeling coding sequence evolution can have strongly negative consequences, notably that when dS variation is ignored, this can create false positve and false negative results

in selection detection (Kosakovsky Pond and Muse, 2005).

Materials

The data used throughout the following tutorials and exercises are available from `https://github.com/veg/evogenomics_hyphy` . A "README" file in the top directory of this repository provides a detailed description of all contents. Importantly, all datasets used here reside in the `datasets` directory. Please refer to `https://www.hyphy.org` for instructions on downloading and installing HyPhy to your system. All exercises have been validated using version 2.3.3. Throughout, we will use the HYPHYMP executable (MP = multiprocessor). For all analyses, you will need the following information:

a). the **full path** to all files being analyzed (alignment and tree), e.g. `/home/user/data/alignment.fna`,

b). the genetic code (in almost all cases, universal),

c). level of statistical significance; suggestions are given for each method below.

All methods will produce a final file of results in JSON (JavaScript Object Notation) format, a highly extensible format that is simple, relatively compact, and both machine- and human-readable. JSON output files can be visually and interactively examined within our new web application, HYPHY-VISION, accessible at `vision.hyphy.org`.

All methods employ the general time reversible nucleotide model for initial branch length optimization and correcting nucleotide substitution biases, followed by fitting a Muse-Gaut model (with general time reversible nucleotide biases) to obtain preliminary dN/dS estimates [see Kosakovsky Pond and Frost (2005) for more details] for detailed model description] for selection inference. Codon frequencies estimated using the CF3x4 procedure (Kosakovsky Pond et al., 2010). In our view, the historical rationale for using simpler evolutionary models (e.g. K80, F81, or HKY85), namely computational cost, to fit nucleotide data is no longer relevant.

Finally, we consider different P-value thresholds depending on the given analysis method. As site-level methods (FEL, SLAC, MEME) tend to be conservative on biological data, we consider significance as $P \leq 0.1$ (or posterior probability ≥ 0.9 for FUBAR). By contrast, we consider significance as $P \leq 0.05$ for alignment-wide methods BUSTED, RELAX, and aBSREL.

Methods

How to run a selection analysis

There is a uniform workflow to run any of the described methods, either locally (on one's own computer and/or a high-performance computing environment) in HyPhy, or using the Datamonkey web-service, available at `www.datamonkey.org`. The version of HyPhy that supports all of the analyses is a command-line program, i.e., it must be run from a terminal prompt (similar to most other bioinformatics packages) in Linux or Mac OS X. It is also possible to run the program in Windows, with an appropriate POSIX emulation environment (e.g., MinGW) installed.

To execute a selection analysis locally, the following steps will need to be taken.

1. Prepare your coding sequence alignment. In general, any duplicate sequences should be removed before analysis. Most importantly, it is imperative that the sequence alignment be in reading frame, meaning that alignment must be performed with codon structure in mind. A common approach to ensuring that this criterion is met is to perform alignment on the translated amino-acid data, and then back-translate to the original nucleotides.
2. Prepare a phylogenetic tree from the multiple sequence alignment. Note that certain analyses may require a labeled phylogenetic tree, as indicated within each subsequent tutorial. Keep in mind that for most selection analyses, a tree topology is a nuisance parameter. Hence, while it is advisable to use good practices when inferring trees, minor errors in tree inference tend to have minor effects on gene- and site-level inference. A notable exception occurs when lineage-specific selection is investigated; in this case ensuring high quality tree topologies is important.
3. An essential and *strongly* recommended step before analyzing data for selection is to screen sequences for recombination. If recombinant sequences are naively analyzed with an appropriate phylogenetic correction, inference results are likely to be biased (Posada et al. (2002), and Section).
4. Prepare your data (alignment and phylogeny) for input to HyPhy. There are three ways to provide a dataset for HyPhy analysis, each of which will trigger a different analysis prompt at runtime:
    - Two separate files containing the alignment and phylogeny, respectively. In this circumstance, HyPhy will issue two successive prompts: the first for the file containing the alignment, and the second for the file containing the tree.
    - A single file containing an alignment in one of the formats supported by HyPhy (FASTA, MEGA, PHYLIP), with a Newick-formatted phylogeny included at the bottom of this file. In this circumstance, HyPhy will issue two successive prompts: the first for the file containing the alignment, and the second asking whether to accept the tree found in the file (provide the affirmative response, e.g., "y", to accept it).
    - A NEXUS file containing both the alignment and phylogeny. In this circumstance, HyPhy will automatically accept the provided phylogeny and therefore will only issue a single prompt for the file containing the alignment. This is also the format that can be used to specify partitioned data, which is necessary to account for recombination.
5. Execute the appropriate method in HyPhy, selecting options suitable for the specific analysis.
6. Each method will provide live on-the-screen progress updates and, when finished, a text summary of the analysis. The output is generated in Markdown, which can either be read directly as text or formatted using one of many Markdown viewers.
7. When an analysis is finished, HyPhy will write a JSON file with numerous details about the analysis to disk. By convention, this file will be placed in the same directory as the input alignment file, with the added `<method>.json` extension, e.g., `flu_ha.nex.BUSTED.json` for an input alignment named `flu_ha.nex`

analyzed by the method BUSTED. All results contained in this JSON file can be explored visually within a web browser using a web application from the HYPHY-VISION suite of tools, accessible at `vision.hyphy.org`. Since JSON files can be easily accessed by scripting and data-analysis languages, these are also well-suited for incorporation into pipelines.

When run through `www.datamonkey.org`, this entire workflow is automated: one simply uploads an alignment, selects options for the analysis, and waits for the job to finish. Once the job has completed, the results will be displayed in an interactive application within the web browser. Note that, if there are duplicate sequences in the provided alignment, Datamonkey will automatically remove these sequences before executing any analysis.

BUSTED

What biological question is the method designed to answer? Is there evidence that some sites in the alignment have been subject to positive diversifying selection, either pervasive (throughout the evolutionary tree) or episodic (only on some lineages)? In other words, BUSTED(Murrell et al., 2015) asks whether a given gene has been subject to positive, diversifying selection at any site, at any time. If *a priori* information about lineages of interest is available (e.g., due to migration, change in the environment, etc.), then BUSTED can be restricted to test for selection only on a subset of tree lineages, potentially boosting power.

Recommended applications.

1. **Annotating** a collection of alignments with a binary attribute: has this alignment been subject to positive diversifying selection (yes/no)? (Price, 2015).
2. Testing **small or low-divergence alignments** (i.e. $\leq\sim$ 30 sequences) for evidence of positive diversifying selection, where neither branch nor site level methods have sufficient power.

Statistical test procedure. Each (branch, site) pair evolves with $\omega_1 \leq \omega_2 \leq 1$, or $\omega_3 \geq 1$, with the ratio chosen independently of other (branch, site) pairs with probability p_1, p_2, p_3 (normalized to sum to 1). The three-rate ω distribution is estimated jointly from the entire alignment, i.e., they are shared by all (branch,site) combinations. Therefore, BUSTED is technically a "branch-site" model (Kosakovsky Pond et al., 2011), although it is not intended to detect individual sites which drive signal of selection.

The test for episodic diversifying selection is performed by comparing the full model versus the nested null model, where ω_3 is constrained to 1. Statistical significance is obtained by the likelihood ratio test, assuming the χ^2_2 asymptotic distribution of the likelihood ratio statistic under the null model.

When only some of the branches are chosen for testing, and the remainder are designated as the background, two independent three-rate ω distributions are fitted: one for the test branches, and one for the background branches. Testing for selection is carried out by constraining the distribution on the test branches as described above.

Example Analysis To begin, we will perform a BUSTED analysis using a dataset of primate-specific KSR2, kinase suppressor of RAS2, genes from (Enard et al., 2016). This gene has been implicated as a so-called "virus-interacting protein," and previous work has suggested it has experienced adaptation in mammalian lineages due to selective pressures exerted by viruses (Enard et al., 2016). We will test all lineages for positive selection (rather than a subset of "test" branches), thereby asking the question: "Has KSR2 been subject to diversifying selection at some time during evolution in primates?"

To run BUSTED, open a terminal session and enter HYPHYMP from the command line to launch the HyPhy analysis menu. Enter `1` (Selection Analyses) and then `5` to reach the BUSTED analysis menu, and supply values for the following prompts:

1. **Choose genetic code.** This option tells HyPhy which translation table to use for codon-level analyses. Enter `1` to use the Universal genetic code.
2. **Select a coding sequence alignment file.** Provide the full path to the dataset of interest: `/path/to/data/ksr2.fna`.
3. **A tree was found in the data file...Would you like to use it (y/n)?** Enter "y" to use the tree.
4. **Choose the set of branches to test for selection.** Enter `1` to test all branches for selection.

BUSTED will now run to completion, printing status indicators to screen while it runs. For an example of how this output will look when rendered into HTML (or similarly, PDF), see this link: `http://bit.ly/2vsRZrh`.

Listing 1: Partial BUSTED screen output

```
### Branches to test for selection in the BUSTED analysis
* Selected 15 branches to test in the BUSTED analysis: `HUM, PAN, Node6, GOR, Node5, PON, Node4, GIB, Node3, MAC,
     BAB, Node12, Node2, MAR, BUS`


### Obtaining branch lengths and nucleotide substitution biases under the nucleotide GTR model
* Log(L) = -5768.01, AIC-c = 11582.06 (23 estimated parameters)

### Obtaining the global omega estimate based on relative GTR branch lengths and nucleotide substitution biases
* Log(L) = -5342.48, AIC-c = 10745.17 (30 estimated parameters)
* non-synonymous/synonymous rate ratio for *test* =   0.0342

### Improving branch lengths, nucleotide substitution biases, and global dN/dS ratios under a full codon model
* Log(L) = -5333.46, AIC-c = 10727.13 (30 estimated parameters)
* non-synonymous/synonymous rate ratio for *test* =   0.0307

### Performing the full (dN/dS > 1 allowed) branch-site model fit
* Log(L) = -5319.67, AIC-c = 10707.62 (34 estimated parameters)
* For *test* branches, the following rate distribution for branch-site combinations was inferred

|          Selection mode           |     dN/dS     |Proportion, %|               Notes                |
|-----------------------------------|---------------|-------------|------------------------------------|
|        Negative selection         |     0.024     |   99.151    |                                    |
|        Negative selection         |     0.085     |    0.812    |                                    |
|      Diversifying selection       |    118.143    |    0.037    |                                    |


### Performing the constrained (dN/dS > 1 not allowed) model fit
* Log(L) = -5326.18, AIC-c = 10718.63 (33 estimated parameters)
* For *test* branches under the null (no dN/dS > 1 model), the following rate distribution for branch-site
     combinations was inferred

|          Selection mode           |     dN/dS     |Proportion, %|               Notes                |
|-----------------------------------|---------------|-------------|------------------------------------|
|        Negative selection         |     0.000     |   10.598    |                                    |
|        Negative selection         |     0.000     |   86.086    |       Collapsed rate class         |
|         Neutral evolution         |     1.000     |    3.316    |                                    |

----
## Branch-site unrestricted statistical test of episodic diversification [BUSTED]
Likelihood ratio test for episodic diversifying positive selection, **p =   0.0015**.
```

Interpreting results. The results printed to the terminal indicate a highly significant result ($P = 0.0015$) in the test for whole-gene selection. Analysis with BUSTED therefore provides robust evidence that KSR2 experienced episodic positive selection in the primates. Because we performed the original BUSTED analysis on the entire tree (i.e. without a specified set of test branches), we do not know from this result along which lineages KSR2 was subject to positive selection. We can conclude only that a non-zero proportion of sites on some lineage(s) in the primate tree experienced diversifying selection pressure.

The output additionally provided information about the specific BUSTED model fits to the test data, including the inferred ω distributions and corresponding weights. The BUSTED alternative model (shown under the output header `Performing the full (dN/dS > 1 allowed) branch-site model fit`) found that a very small proportion (only ∼0.037%) of sites evolved under a very large ω of over 100 (118.143). Importantly, neither of these estimates will be precise because they were derived from a small subset of the data. As such, all the BUSTED test establishes is that the proportion of sites along test lineages (here, the entire phylogeny) with $\omega > 1$ is non-zero. For example, if BUSTED had inferred a rate category of $\omega = 10$ on a different gene, it would *not* be correct to claim that this gene evolves under weaker selection than does KSR2. A formal statistical test would have to be carried out to establish such a claim.

Conversely, had the result not been statistically significant, we would not be able to reject the null hypothesis that no positive selection had occurred in KSR2. Importantly, however, a negative *would not* unequivocally rule out the presence of positive selection. This outcome could be caused a lack of statistical power wherein the provided data did not contain a sufficiently strong signal for BUSTED to detect selection.

Because BUSTED assumes a fixed model complexity (3 values of ω) a priori, this sometimes leads to over-parameterized (or under-parameterized) models. For example, in the constrained model for KSR2, two of the three rate classes have the same value of $\omega(0.0)$, implying that one of them is unnecessary. HyPhy will report this to the screen as a diagnostic message, but there is no corrective action that needs to be taken. These messages simply point to *low-complexity* data.

We will additionally take this opportunity to showcase the visual power of our accompanying web browser, HyPhy-Vision. Figure 1 displays the rendering of the output `ksr2.fna.BUSTED.json` as it appears in HyPhy-Vision. On this site, users can interactively view and explore inference results, view figures and charts, and perform other tasks.

Rules of thumb for BUSTED use.

1. Best applied to small or medium-sized datasets (e.g., up to 100 sequences). Larger datasets will take longer to run, and may be well described by a fixed complexity model.
2. If one suspects that only a small subset of lineages is subject to selection, e.g., because the phenotype, environment, or fitness changed along those branches, designating those *a priori* as the test set will significantly boost power (see an Exercise).
3. In simulation studies, BUSTED performs best when a sufficient proportion (5-10%) of branch site combinations is subject to positive diversifying selection, and

the effect size (ω value) is reasonably large (e.g., ≥ 3).

RELAX

What biological question is the method designed to answer? Is there evidence the strength of selection has been relaxed (or conversely intensified) on a specified group of lineages (*Test*) relative to a set of reference lineages (*Reference*)? Importantly, RELAX *is not* designed to detect diversifying selection specifically. We note that the RELAX framework can perform both this specific hypothesis test as well as fit a suite of descriptive models which address, for example, overall rate differences between test and reference branches or lineage-specific inferences of selection relaxation. We focus our attention here on RELAX's hypothesis testing abilities. More information about descriptive analyses is available on `hyphy.org` as well as in RELAX's primary publication (Wertheim et al., 2015).

Recommended applications.

1. Testing for a systematic shift (relaxation / intensification) in the distribution of selection pressure associated with major biological transitions such as hosting switching in viruses Forni et al. (2017), lifestyle evolution in bacteria (i.e. transition from free-living to endosymbiotic lifestyle (Wertheim et al., 2015))
2. Comparing selective regimes between two subsets of branches in the tree, e.g. to investigate selective differences among transmission routes in HIV-1 (Tully et al., 2016).

Statistical test procedure. Given a tree with at least two sets of branches, one of which is designated as *Test*, and the other - as *Reference*, the core version of RELAX compares two nested models, which follow the same general framework as BUSTED. Each (branch, site) combination is drawn independently from a 3-rate ω distribution. The evolutionary rates for *Test* branches are functions of those for *Reference* branches. Specifically, $\omega_{\text{Test}} = \omega_{\text{Reference}}^{K}$, where K is the relaxation or intensification parameter. The alternative model infers K from the data, and the null model sets $K = 1$. Statistical significance is obtained by the likelihood ratio test, assuming the χ_1^2 asymptotic distribution. asymptotic distribution of the likelihood ratio statistic under the null model. A significant result of $K > 1$ indicates that selection strength has been *intensified* along the test branches, and a significant result of $K < 1$ indicates that selection strength has been *relaxed* along the test branches. This is because for $K < 1$ the values of ω for *Test* branches shrink towards neutrality ($\omega = 1$) relative to *Reference*, and for $K > 1$ they move away from neutrality.

If some branches in the tree belong to neither the *T*est or the *R*eference set, they are allocated to a group with its own (*Unclassified*) distribution of ω, which is uncoupled from the testing procedure.

Example Analysis. We will perform a RELAX analysis using a dataset of Influenza A PB2 subunit sequences[2] from Tamuri et al. (2012). The PB2 subunit, which is part of

[2]Note that the original dataset in Tamuri et al. (2012) contained 401 sequences. For the purposes of this tutorial, we analyze a subset of this alignment with only 60 sequences thereby achieving a tractable runtime

influenza's RNA polymerase complex, has emerged as a critical determinant of influenza infectivity and, as a consequence, host range (Labadie et al., 2007; Graef et al., 2010). The dataset we examine here contains both sequences from avian host and human host strains. Previous studies have shown that this host switch is correlated with significant shifts in selection pressures and preferred amino acids at key sites in PB2 (Tamuri et al., 2009, 2012; Rodrigue, 2013). We now re-analyze this dataset using RELAX to ask a different but related question: "Was the shift from avian to human hosts associated with a relaxation of selection pressures in Influenza A PB2?"

RELAX requires an *a priori* specification of test and reference lineages, although not all lineages in a tree need to be classified. As such, you must label your test (and reference, if desired) branches in the input phylogeny. We provide an online widget to assist with tree labeling at `http://phylotree.hyphy.org`. The dataset we have provided for this analysis already has a labeled phylogeny, with the Human host lineages labeled as "test".

To run RELAX, open a terminal session and enter HYPHYMP from the command line to launch the HyPhy analysis menu. Enter `1` (Selection Analyses) and then `7` to reach the RELAX analysis menu, and supply values for the following prompts:

1. **Choose genetic code.** Enter `1` to use the Universal genetic code.
2. **Select a coding sequence alignment file.** Provide the full path to the dataset of interest: `/path/to/data/pb2.fna`.
3. **A tree was found in the data file...Would you like to use it (y/n)?** Enter "y" to use the tree.
4. **Choose the set of branches to test for selection.** This option asks you to specify the *label* inside your tree used to specify the test lineages. You can either select all unlabeled branches, or HyPhy will show all labels it found in the tree you provided. Enter `1` to select the branches labeled as "test" as the test set in RELAX analysis.
5. **Analysis type.** This option asks you to specify the scope of RELAX analysis. Selecting "Minimal" will run the RELAX hypothesis test, and selecting "All" will run hypothesis testing and fit two additional descriptive models, described earlier. Here, we will perform only hypothesis testing to determine whether the data shows evidence for a relaxation or intensification of selection intensity between the test and reference lineages. Enter the option `2` to run the "Minimal" analysis.

RELAX will now run to completion, printing status indicators to screen while it runs:

Listing 2: Partial RELAX screen output

```
### Obtaining branch lengths and nucleotide substitution biases under the nucleotide GTR model
* Log(L) = -16755.26, AIC-c = 33660.66 (75 estimated parameters)

### Obtaining the global omega estimate based on relative GTR branch lengths and nucleotide substitution biases
* Log(L) = -14410.97, AIC-c = 28988.46 (83 estimated parameters)
* non-synonymous/synonymous rate ratio for *Reference* =   0.0401
* non-synonymous/synonymous rate ratio for *Test* =   0.0604

### Improving branch lengths, nucleotide substitution biases, and global dN/dS ratios under a full codon model
* Log(L) = -14354.67, AIC-c = 28875.86 (83 estimated parameters)
* non-synonymous/synonymous rate ratio for *Reference* =   0.0358
* non-synonymous/synonymous rate ratio for *Test* =   0.0609
```

on a personal machine.

```
### Fitting the alternative model to test K != 1
* Log(L) = -14337.22, AIC-c = 28849.02 (87 estimated parameters)
* Relaxation/intensification parameter (K) =     0.73
* The following rate distribution was inferred for **test** branches

|          Selection mode           |     dN/dS     |Proportion, %|               Notes                |
|-----------------------------------|---------------|-------------|------------------------------------|
|        Negative selection         |     0.031     |   94.752    |                                    |
|        Negative selection         |     0.086     |    2.951    |                                    |
|      Diversifying selection       |     1.406     |    2.297    |                                    |

* The following rate distribution was inferred for **reference** branches

|          Selection mode           |     dN/dS     |Proportion, %|               Notes                |
|-----------------------------------|---------------|-------------|------------------------------------|
|        Negative selection         |     0.009     |   94.752    |                                    |
|        Negative selection         |     0.035     |    2.951    |                                    |
|      Diversifying selection       |     1.591     |    2.297    |                                    |


### Fitting the null (K := 1) model
* Log(L) = -14342.33, AIC-c = 28857.22 (86 estimated parameters)
* The following rate distribution for test/reference branches was inferred

|          Selection mode           |     dN/dS     |Proportion, %|               Notes                |
|-----------------------------------|---------------|-------------|------------------------------------|
|        Negative selection         |     0.010     |   94.149    |                                    |
|        Negative selection         |     0.021     |    3.391    |                                    |
|      Diversifying selection       |     1.735     |    2.460    |                                    |

----
## Test for relaxation (or intensification) of selection [RELAX]
Likelihood ratio test **p =   0.0014**.
>Evidence for intensification of selection among **test** branches _relative_ to the **reference** branches at P
      <=0.05
----
```

Interpreting results. On this data, RELAX has inferred a relaxation parameter $K = 0.73$ with a highly significant $P = 0.0014$. Therefore, there is evidence to reject the null hypothesis that selection pressure has not been shifted in the test (here, human host) lineages. We instead have strong evidence that selection has been *relaxed* (because the inferred $K < 1$) in the human host lineages. In other words, selection in the test branches has generally moved towards neutrality ($\omega = 1$) compared to the reference branches. This finding is consistent with the evolutionary changes that typically occur during a virus host-switching event, wherein selection stringency will be reduced to facilitate viral adaptation.

Keep in mind that RELAX defines relaxation (or intensification) in a fairly restrictive fashion, i.e., all selective regimes (negative and positive) must weaken (or strengthen). Therefore certain relaxation scenarios, e.g., when only positive selection is relaxed, but negative selection is maintained, may result in a negative RELAX test.

Rules of thumb for RELAX use.

1. Always provide a labeled phylogeny indicating which branches to include in the "test" lineages. You can additionally label "reference" lineages if you wish to keep some branches as unclassified. It is convenient to use the PHYLOTREE.JS online widget at `http://phylotree.hyphy.org/` to label branches before analysis.

aBSREL

It is often of interest to determine whether a specific lineage or lineage(s) have been subject to selection. Such analyses have historically been performed using the so-called "branch" or "branch-site" class of models, which allow evolutionary rates to vary across branches, or across sites **and** branches (Yang and Nielsen, 2002; Zhang et al., 2005; Kosakovsky Pond et al., 2011). Early versions of branch-site models allowed users to

compare selection pressure on a pre-selected branch sets of "foreground" branches to a pre-selected set of "background" branches, on which positive selection was disallowed (Yang and Nielsen, 2002; Zhang et al., 2005). [Note that this approach is similar to how BUSTED performs gene-wide selection inference (Murrell et al., 2015)]. Later efforts demonstrated that disallowing positive selection on background branches could lead to highly elevated false positive rates and advocated a strategy wherein any branch, regardless of data partition, could evolve at any rate (Kosakovsky Pond et al., 2011). This strategy has been described as the BS-REL model in HyPhy (Kosakovsky Pond et al., 2011). However, in BS-REL, each branch was constrained to have three rate categories, an assumption with little justification. Since then, we have developed a greatly improved branch-site model called aBSREL ("adaptive branch-site random effects likelihood"). Rather than assuming that each branch should be fit with three rate classes, aBSREL infers, using small-sample Akaike Information Criterion correction (AICc), the optimal number of rate categories per branch. In this manner, computational complexity and the number of parameters are greatly reduced, leading to a tractable runtime for larger datasets that could not otherwise be studied with other more intensive branch-site models.

What biological question is the method designed to answer? Like classical branch-site models, aBSREL asks whether some proportion of sites is subject to positive selection along specific branches or lineages of a phylogeny.

Recommended applications.

1. Exploratory testing for evidence of lineage-specific positive diversifying selection in small to medium sized alignments (up to 100 sequences).
2. Targeted testing of branches selected *a priori* for positive diversifying selection, including alignments with prohibitive runtimes under older branch-site models (up to 1,000 sequences) (Smith et al., 2015).

Statistical test procedure. aBSREL uses the theoretic information criterion AIC_c to automatically determine the complexity of the evolutionary process at every branch (Smith et al., 2015). As a heuristic optimization, aBSREL will always examine branches in order from longest to shortest, because longer branches tend to be the ones requiring more complex models. In this adaptive model, one rate class is allowed to assume any value of $\omega > 1$, whereas for any other inferred rate class is constrained as $\omega \leq 1$. In the null model, all ω categories are constrained as $\omega \leq 1$. For any branch inferred to have sufficient rate variation (i.e. more than one rate category) where one rate category is described by $\omega > 1$, aBSREL will proceed to fit a null model to this branch. If $\max \omega \leq 1$, the null model will have the same exact fit as the alternative model, and the resulting P-value is 1. The test for lineage-specific diversifying selection is performed by comparing the full model versus the nested null model, and statistical significance is obtained by the likelihood ratio test. Significance is evaluated using a mixture of $50\%\chi^2_0$, $20\%\chi^2_1$, and $30\%\chi^2_2$ distributions (proportions determined via simulations Smith et al. (2015)). Finally, aBSREL will correct all P-values obtained from individual tests for multiple comparisons using the Bonferroni-Holm procedures which controls family wise false positive rates (i.e., the probability of generating one or more false positives, when all null hypotheses are correct).

One can either select a specific set of branches in order to test a specific *a priori* hypothesis, or one can perform an exploratory analysis across the entire phylogeny by testing all branches for selection. The former approach may have substantially more power to detect selection, especially if only a few branches in a large tree are chosen, due to the decreased volume of multiple testing, at the risk of potentially missing branches subject to positive selection that have not been included in testing.

Example Analysis. Here, we will demonstrate aBSREL use and interpretation using a dataset of HIV-1 *env* sequences collected from an epidemiologically-linked donor-recipient transmission pair (Frost et al., 2005). This dataset can be found in the provided file `hiv1_transmission.fna`.

To run aBSREL, open a terminal session and enter HYPHYMP from the command line to launch the HyPhy analysis menu. Enter `1` (Selection Analyses) and then `6` to reach the aBSREL analysis menu, and supply values for the following prompts:

1. **Choose genetic code.** This option tells HyPhy which translation table to use for codon-level analyses. Enter `1` to use the Universal genetic code.
2. **Select a coding sequence alignment file.** Provide the full path to the dataset of interest: `/path/to/hiv1_transmission.fna`.
3. **A tree was found in the data file...Would you like to use it (y/n)?** Enter "y" to use the included tree.
4. **Choose the set of branches to test for selection.** You can now select on which branches aBSREL should conduct a formal hypothesis test for positive selection.Enter `1` to test all branches for selection.

aBSREL will now run to completion, printing status indicators to screen while it runs (some output abbreviated)

Listing 3: Partial aBSREL screen output

```
### Obtaining branch lengths and nucleotide substitution biases under the nucleotide GTR model
* Log(L) = -5524.50, AIC-c = 11153.08 (52 estimated parameters)

### Fitting the baseline model with a single dN/dS class per branch, and no site-to-site variation.
* Log(L) = -5402.40, AIC-c = 11009.72 (102 estimated parameters)
* Branch-level non-synonymous/synonymous rate ratio distribution has median  0.66, and 95% of the weight in  0.00
      -  5.41

### Determining the optimal number of rate classes per branch using a step up procedure

|   Branch    |  Length  |  Rates   |     Max. dN/dS     |    Log(L)     |     AIC-c     |Best AIC-c so far|
|-------------|----------|----------|--------------------|---------------|---------------|-----------------|
|   0564_22   |   0.01   |    2     |    1.96 (52.27%)   |   -5402.41    |   11013.78    |    11009.72     |
|   0564_7    |   0.01   |    2     |    0.74 ( 5.19%)   |   -5402.40    |   11013.76    |    11009.72     |
|  Separator  |   0.01   |    2     |  197.32 ( 3.95%)   |   -5397.53    |   11004.02    |    11004.02     |
|  Separator  |   0.01   |    3     |  180.22 ( 4.08%)   |   -5397.53    |   11008.06    |    11004.02     |
|   0564_4    |   0.01   |    2     |   29.79 ( 2.15%)   |   -5394.37    |   11001.74    |    11001.74     |
|   0564_4    |   0.01   |    3     |   29.78 ( 2.15%)   |   -5394.37    |   11005.78    |    11001.74     |
|   0564_3    |   0.01   |    2     |  126.86 ( 3.14%)   |   -5388.59    |   10994.22    |    10994.22     |
|   0564_3    |   0.01   |    3     |  135.96 ( 3.05%)   |   -5388.59    |   10998.25    |    10994.22     |
|   0564_9    |   0.01   |    2     |   10.01 ( 8.61%)   |   -5388.37    |   10997.82    |    10994.22     |
...
|   Node53    |   0.00   |    2     |   1.00 (100.00%)   |   -5371.63    |   10976.46    |    10971.76     |
|   0557_6    |   0.00   |    2     |  27.66 (100.00%)   |   -5371.32    |   10975.83    |    10971.76     |
|   0557_21   |   0.00   |    2     |    0.25 ( 1.96%)   |   -5371.30    |   10975.80    |    10971.76     |
|   0557_7    |   0.00   |    2     |    0.25 ( 1.96%)   |   -5371.30    |   10975.80    |    10971.76     |

### Rate class analyses summary
*  38 branches with **1** rate classes
*  6 branches with **2** rate classes

### Improving parameter estimates of the adaptive rate class model
* Log(L) = -5370.66, AIC-c = 10970.49 (114 estimated parameters)

### Testing selected branches for selection
```

Branch	Rates	Max. dN/dS	Test LRT	Uncorrected p-value
0564_22	1	1.22 (100.00%)	0.11	0.43015
0564_7	1	0.61 (100.00%)	0.00	1.00000
Separator	2	197.72 (3.95%)	14.13	0.00029
0564_4	2	28.89 (2.15%)	4.81	0.03281
0564_3	2	127.66 (3.14%)	14.06	0.00030
0564_9	1	0.72 (100.00%)	0.00	1.00000
0564_1	1	1.07 (100.00%)	0.01	0.48208
...				
0557_21	1	1.00 (100.00%)	0.00	1.00000
0557_7	1	1.00 (100.00%)	0.00	1.00000

Adaptive branch site random effects likelihood test

Likelihood ratio test for episodic diversifying positive selection at Holm-Bonferroni corrected _p = 0.0500_
found **3** branches under selection among **44** tested.

* Node35, p-value = 0.00018
* Separator, p-value = 0.01251
* 0564_3, p-value = 0.01266

Interpreting results. The first table summarizes the model selection process. For example, when two ω rates were assigned to branch *S*eparator, this improved the AIC_c score of the fit (compared to the single rate model) from 11009.72 to 11004.02. However, allocating three ω rates to the same branch worsens the score to 11008.06. Therefore the aBSREL model will use two ω rates at the branch.

The second table shows the results of tests for episodic selection on individual branches. At branch *0*564_4 , the tested model included two ω rates, with the positive selection class taking on value 28.89 (2.15% proportion of the mixture). Constraining this rate to range between 0 and 1 yields the likelihood ration test statistic of 4.81, which maps to a P-value (before multiple test correction) of 0.03281.

For this dataset, aBSREL identified three branches that were subject to diversifying selection pressure. Further examination of results using HyPhy Vision shows that these branches are found i) along the transmission event from donor to recipient, and ii) within a highly-diverged clade in the donor (Figure 2). The first finding is consistent with an expected increase in evolutionary rate when a virus infects a new host and encounters novel host immunity, and the second finding is consistent with intrahost adaptive dynamics of the donor's long-term HIV infection. Importantly, a close examination of the markdown-output table under the header `Testing selected branches for selection` reveals several nodes with uncorrected p-values whose significance was lost upon applying the Bonferroni-Holm correction, e.g. `0564_4`. This result illustrates the potential loss of power incurred by this aBSREL exploratory analysis.

Rules of thumb for aBSREL use

1. *A priori* identification of branches to test for selection will generally increase power to detect selection on those branches. That said, to maintain statistical robustness, we *strongly discourage* performing multiple separate tests for selection on different branch sets. Such an approach will necessarily introduce false positives. In such a case, we recommend performing an exploratory analysis wherein all branches are considered
2. Exploratory analyses of very large datasets are unlikely to yield many significant results, because correcting for multiple testing will reduce power as the number of branches grows, while the amount of statistical signal does not increase for larger datasets. One option is to thin out large phylogenies (before performing any testing), retaining major clades and lineages of interest.

Site-level selection: MEME, FEL, SLAC, and FUBAR

What biological question is the method designed to answer? The methods FEL, SLAC, and FUBAR address the question: Which site(s) in a gene are subject to *pervasive*, i.e. consistently across the entire phylogeny, diversifying selection? MEME addresses a more general question: Which site(s) in a gene are subject to pervasive or *episodic*, i.e. only on a single lineage or subset of lineages, diversifying selection?

Recommended applications.

1. **MEME** is the sole method in HyPhy for detecting selection at individual sites that considers both pervasive and episodic selection. MEME is therefore our recommended method if maximum power is desired.
2. The phenomenon of pervasive selection is generally most prevalent in pathogen evolution and any biological system influenced by evolutionary arms race dynamics (or balancing selection), including adaptive immune escape by viruses. As such, FEL, SLAC, and FUBAR are ideally suited to identify sites under positive selection which represent candidate sites subject to strong selective pressures across the entire phylogeny . Each of these methods has a particular use case as well:
    - **FEL** is our recommended method for analyzing small-to-medium size datasets when one wishes only to study pervasive selection at individual sites.
    - **FUBAR** is our recommended method for detecting pervasive selection at individual sites on large (> 500 sequences) datasets for which other methods have prohibitive runtimes, unless you have access to a computer cluster.
    - **SLAC** provides legacy functionality as a counting-based method adapted for phylogenetic applications. In general, this method will be the least statistically robust.

Statistical test procedure. Each method presented here employs a distinct algorithmic approach to inferring selection. FEL uses maximum likelihood to fit a codon model to each site, thereby estimating a value for dN and dS at each site. FEL tests for selection with the likelihood ratio test, asking whether the dN estimate is significantly greater than the inferred dS estimate.

SLAC represents the most basic inference method and is an extension of the Suzuki-Gojobori counting-based method (Suzuki and Gojobori, 1999) for phylogenetically-related sequences (as opposed to sequence pairs). SLAC uses maximum likelihood to infer ancestral characters for each site across the phylogeny and then directly counts the number of synonymous and non-synonymous changes which have occurred at each site over evolutionary time. SLAC then tests for selection by testing whether or not there are too many or too few non-synonymous changes compared to what is expected under neutrality. The neutral expectation is derived based on the phylogeny-wide estimated numbers of synonymous and non-synonymous nucleotide sites at a given codon. The statistical test employs the binomial distribution to compute significance, e.g. how likely is it to observe 13 non-synonymous and 1 synonymous substitution at a site, if the expected synonymous to non-synonymous substitution count ratio under neutrality is $1 : 4$.

MEME employs a mixed-effects maximum likelihood approach. For each site, MEME infers two ω rate classes and corresponding weights representing the probability that

the site evolves under each rate class at a given branch. To this end, MEME infers a single α (dS) parameter and two separate β (dN) parameters, β_- and β_+. The ω rates per site, therefore, consist of β_+/α and β_-/α. MEME uses this framework to fit a null and alternative model each, both models enforcing the constraint $\beta_- \leq \alpha$. The null model disallows positive selection by enforcing the constraint $\beta_+ \leq \alpha$, whereas the alternative model places no constraint on β_+. MEME uses the likelihood ratio test to compare between null and alternative model fits, with significance assessed using the mixture of $33\%\chi^2_0$, $30\%\chi^2_1$, and $37\%\chi^2_2$.

FUBAR takes a Bayesian approach to selection inference and is a particular case of statistical models developed in the context of document classification (latent Dirichlet allocation). The key innovation to FUBAR's approach is its use of an *a priori* specified grid of dN and dS values (typically 20×20), spanning the range of negative, neutral, and positive selection regimes, whose likelihoods can be precomputed and used throughout analysis (rather than having to re-compute likelihoods during optimization as traditional random-effects approaches do (Nielsen and Yang, 1998; Kosakovsky Pond and Frost, 2005)). This approach, combined with other algorithmic advances, speeds computation time by an order of magnitude compared to FEL, whilst yielding comparable statistical performance. FUBAR estimates every model parameter except the proportion of sites allocated to each grid point using simple (and fast) nucleotide models. The proportions are estimated using an MCMC procedure, and non-neutral evolution at each site is inferred using a straightforward naive empirical Bayes approach (Nielsen and Yang, 1998). Sites are called positively or negatively selected if the corresponding posterior probabilities are sufficiently high.

All methods with the exception of MEME report both positively and negatively selected sites.

Example Analysis We will demonstrate the use and interpretation of site-level methods using data from influenza strain H3N2 (the "Hong Kong flu"), the primary circulating strain of seasonal influenza the late 1960s. We specifically will assess selection on the H3 hemagglutinin, the influenza surface protein which is responsible for host cell binding. Hemagglutinin experiences rapid evolution triggered by host immune escape, and previous studies have identified numerous signatures of positive diversifying selection in H3 sequences with a particular concentration around the host-binding domain (Nelson and Holmes, 2007)

We base analyses here on an alignment from Meyer and Wilke (2015) of 2555 full H3 sequences sampled over time since the 1991–1992 influenza season. We removed all partial and strongly outlying sequences (i.e. those with excessive divergence) from the original dataset before proceeding. We further subsetted this alignment to two smaller alignments with comparable numbers of taxa but spanning different evolutionary time frames: The first smaller alignment ("trunk") contains 163 sequences sampled along the influenza H3 trunk, whereas the second smaller alignment ("shallow") contains 121 sequences sampled from a single clade (Figure 3). Therefore, while these two smaller datasets contain a comparable number of sequences, the "trunk" dataset spans a much longer time frame and contains substantially more sequence divergence relative to the "shallow" dataset. Indeed, the trunk dataset has a total tree length (sum of branch lengths, in units substitutions/site/unit time) of 0.43, whereas the shallow dataset had a total tree length of 0.12, meaning that the trunk dataset contains nearly four times the

amount of sequence divergence seen in the shallow dataset. We have compiled results for all three datasets analyzed with all four methods (Table 1). We now describe, using the trunk dataset as an example, how to run each of these analyses in HyPhy.

FEL Launch HyPhy from the command line, and enter options `1` (Selection Analyses) and then `2` to reach the FEL analysis menu, and supply values for the following prompts:

1. **Choose genetic code**. Enter `1` to use the Universal genetic code.
2. **Select a coding sequence alignment file**. Provide the full path to the dataset of interest: `/path/to/data/trunk.fna`.
3. **A tree was found in the data file...Would you like to use it (y/n)?**. Enter "y" to use the tree.
4. **Choose the set of branches to test for selection**. This option allows you to specify which branches along which site-level inference should be performed. Enter `1` to test All branches for selection.
5. **Use synonymous rate variation?**. This option asks you to specify whether the dS parameter in the codon model should be allowed to vary across sites ("Yes") or be fixed to 1 at all sites ("No"). Enter `1` to use a model with synonymous rate variation.
6. **Select the p-value used to perform the test at (permissible range = [0,1], default value = 0.1)**. Provide the default threshold of `0.1`.

FEL will now run to completion and print status indicators to the screen, including results for any site found to be under selection (either positive or negative). Abbreviated results are shown below.

Listing 4: Partial FEL screen output

```
### Obtaining branch lengths and nucleotide rates under the  GTR model
* Log(L) = -7506.06

### Obtaining the global omega estimate based on relative GTR branch lengths and nucleotide substitution biases
* Log(L) = -7302.10
* non-synonymous/synonymous rate ratio for *test* =   0.2923

### Improving branch lengths, nucleotide substitution biases, and global dN/dS ratios under a full codon model
* Log(L) = -7289.65
* non-synonymous/synonymous rate ratio =   0.2598

### For partition 1 these sites are significant at p <=0.1

|     Codon      |   Partition    |     alpha      |      beta      |      LRT       |Selection detected?|
|:--------------:|:--------------:|:--------------:|:--------------:|:--------------:|:-----------------:|
...
|      146       |       1        |        3.818   |        0.000   |        7.336   |  Neg. p = 0.0068  |
|      152       |       1        |        1.968   |        0.000   |        3.634   |  Neg. p = 0.0566  |
|      154       |       1        |        0.000   |        3.912   |        4.652   |  Pos. p = 0.0310  |
|      159       |       1        |        4.413   |        0.716   |        2.972   |  Neg. p = 0.0847  |
|      164       |       1        |        2.082   |        0.000   |        2.713   |  Neg. p = 0.0995  |
|      176       |       1        |        1.659   |        0.000   |        2.986   |  Neg. p = 0.0840  |
|      177       |       1        |        6.393   |        0.000   |        8.421   |  Neg. p = 0.0037  |
|      181       |       1        |        1.928   |        0.000   |        3.286   |  Neg. p = 0.0699  |
|      190       |       1        |        2.085   |        0.000   |        2.715   |  Neg. p = 0.0994  |
|      201       |       1        |        1.645   |        0.000   |        3.370   |  Neg. p = 0.0664  |
|      208       |       1        |        0.000   |        3.625   |        4.668   |  Pos. p = 0.0307  |
...

### ** Found _3_ sites under pervasive positive diversifying and _115_ sites under negative selection at p <=
      0.1**
```

Inference details for codons with significant likelihood ratio tests for positive or negative selection are reported to the screen.

Codon
: The codon where non-neutral evolution has been detected

Partition
: Allows one to keep track which subset of the alignment a particular site belongs to. This is important for recombination-corrected partition analyses.

alpha
: site specific synonymous substitution rate

beta
: site specific non-synonymous substitution rate

LRT
: site specific likelihood ratio test statistic for non-neutral evolution (alpha $\neq$ beta)

Selection detected?
: selection classification (Positive or Negative) and the corresponding P-value

The "Codon" and "Partition" columns are common to all site-specific analyses.

MEME and SLAC SLAC and MEME follow identical menu prompts as FEL, with the exception that only FEL will prompt for synonymous rate variation. Instead, SLAC has a different prompt for Step 5: **Select the number of samples used to assess ancestral reconstruction uncertainty**. If this number is positive, then HyPhy will draw samples from the distribution of ancestral states and use them to measure whether or not inference is sensitive to ancestral inference uncertainty. When you encounter this option, provide the default value of `100` (or `0` to forego sampling). MEME does not emit any additional prompts.

Listing 5: Partial SLAC screen output

```
...
### For partition 1 these sites are significant at p <=0.1

| Codon |   Partition    |       S        |       N        |       dS       |       dN       |Selection detected?|
|:-----:|:--------------:|:--------------:|:--------------:|:--------------:|:--------------:|:-----------------:|
...
|  146  |       1        |     3.000      |     0.000      |     3.000      |     0.000      |  Neg. p = 0.037   |
|  154  |       1        |     0.000      |     8.000      |     0.000      |     4.000      |  Pos. p = 0.039   |
|  177  |       1        |     3.000      |     0.000      |     4.038      |     0.000      |  Neg. p = 0.020   |
|  208  |       1        |     0.000      |     6.000      |     0.000      |     2.994      |  Pos. p = 0.089   |

...
### Ancestor sampling analysis

>Generating 100 ancestral sequence samples to obtain confidence intervals

Resampling results for partition 1

| Codon | Part. | S [median, IQR] | N [median, IQR] | dS [median, IQR] | dN [median, IQR] | p-value [median, IQR]|
|:-----:|:-----:|:---------------:|:---------------:|:----------------:|:----------------:|:--------------------:|
...
|  146  |   1   | 3.00 [3.00-3.00]| 0.00 [0.00-0.00]| 3.00 [3.00-3.00] | 0.00 [0.00-0.00] |   0.04 [0.04-0.04].  |
|  154  |   1   | 0.00 [0.00-0.00]| 8.00 [8.00-8.00]| 0.00 [0.00-0.00] | 4.00 [4.00-4.00] |   0.04 [0.04-0.04]   |
|  177  |   1   | 3.00 [3.00-4.00]| 0.00 [0.00-0.00]| 4.04 [4.04-5.38] | 0.00 [0.00-0.00] |   0.02 [0.01-0.02]   |
|  208  |   1   | 0.00 [0.00-0.00]| 6.00 [6.00-6.00]| 0.00 [0.00-0.00] | 2.99 [2.99-2.99] |   0.09 [0.09-0.09]   |

...
```

SLAC reports several key quantities for codons with significant P-values for positive or negative selection to the screen.

S the number of synonymous substitutions inferred at this site

NS the number of non-synonymous substitutions inferred at this site

dS estimated site-specific synonymous rate

dN estimated site-specific non-synonymous rate

Selection detected?
selection classification (Positive or Negative) and the corresponding P-value for the binomial test

If the user elected to perform ancestral resampling, another table is reported, showing how much these quantities are affected by ancestral state reconstruction uncertainty. For example, at codon 177, some ancestral reconstructions yielded 3 synonymous substitutions, whereas others yielded 4; however this was not sufficient to move the P-value on different sides of the threshold.

Listing 6: Partial MEME screen output

```
...
| Codon  | Partition |  alpha  |  beta+  |   p+    |  LRT    |Episodic selection detected?| # branches |
|:------:|:---------:|:-------:|:-------:|:-------:|:-------:|:--------------------------:|:----------:|
|   64   |     1     |  0.000  |  14.717 |  0.204  |  3.512  |      Yes, p = 0.0816       |      0     |
|  154   |     1     |  0.000  |  35.302 |  0.145  |  5.334  |      Yes, p = 0.0317       |      0     |
|  171   |     1     |  0.000  |  45.005 |  0.017  |  5.753  |      Yes, p = 0.0256       |      0     |
|  208   |     1     |  0.000  |  59.749 |  0.089  |  5.554  |      Yes, p = 0.0283       |      0     |
|  242   |     1     |  1.839  |  34.114 |  0.216  |  4.273  |      Yes, p = 0.0549       |      0     |
|  402   |     1     |  0.000  |  10.476 |  0.091  |  3.493  |      Yes, p = 0.0824       |      0     |

### ** Found _6_ sites under episodic diversifying positive selection at p <= 0.1**
```

MEME prints information only about codons subject to positive selection, since MEME does not directly test for negative selection.

alpha
site specific synonymous substitution rate

beta+
site specific non-synonymous substitution rate for the positive selection category

p+ site specific weight ($\sim$ proportion of branches) assigned for the positive selection category

LRT
site specific likelihood ratio test statistic for episodic diversifying selection (beta+ $>$ 1 and p+ > 0)

Episodic selection detected?
selection classification (Yes) and the corresponding P-value

 # branches
an exploratory estimate of the number of individual branches which have sufficient empirical Bayes support for positive selection; since MEME pools signal from multiple branches, there may be overall evidence for selection, without any individual branches "lighting up".

FUBAR To run FUBAR, launch HyPhy from the command line, and enter options `1` (Selection Analyses) and then `4` to reach the FUBAR analysis menu, and supply values for the following prompts[3]:

1. **Choose genetic code**. Enter `1` to use the Universal genetic code.
2. **Select a coding sequence alignment file**. Provide the full path to the dataset of interest: `/path/to/data/h3_trunk.fna`.

[3]Note that for all prompts with default values, simply pressing `enter` will choose this default

3. **A tree was found in the data file...Would you like to use it (y/n)?**. Enter "y" to use the tree.
4. **Number of grid points per dimension.** This option controls how **fine** the FUBAR analysis is by setting the range of possible dN and dS values that can be inferred, along an $N \times N$ grid. We will use the default value of `20` (leading to a 20×20 grid of dN/dS ratios). FUBAR will now pre-compute likelihoods for each value in the grid.
5. **Number of MCMC chains to run**. This option determines the number of Markov Chain Monte Carlo chains to run during Bayesian inference of evolutionary rates. Enter the default value of `5` to run 5 chains.
6. **The length of each chain**. This option controls for how long each MCMC chain should be run. Enter the default value of `2000000` to run each chain for two million generations (thus obtaining two million samples).
7. **Use this many samples as burn-in**. This option determines how many initial samples drawn from the MCMC chain should be discarded as burn-in, as is standard in Bayesian analyses. Enter the default value of `1000000`, leading to a final value of one-million draws per chain.
8. **How many samples should be drawn from each chain**. This option determines the final number of samples to draw from the full set of one-million draws per chain. Enter the default value of `100`.
9. **The concentration parameter of the Dirichlet prior**. This option controls the shape of the Dirichlet prior distribution. Enter the default value of `0.5`.

Listing 7: Partial FUBAR screen output

```
...
### Tabulating site-level results
|  Codon    |Partition |     alpha      |      beta      |     N.eff      |Posterior prob for positive selection|
|:---------:|:--------:|:--------------:|:--------------:|:--------------:|:-----------------------------------:|
|    61     |    1     |        0.753   |        4.365   |       64.549   |       Pos. posterior = 0.9262       |
|    64     |    1     |        0.753   |        3.920   |       77.106   |       Pos. posterior = 0.9095       |
|    69     |    1     |        0.730   |        4.447   |       64.182   |       Pos. posterior = 0.9325       |
|   154     |    1     |        0.637   |        6.595   |       53.312   |       Pos. posterior = 0.9826       |
|   208     |    1     |        0.622   |        5.908   |       55.794   |       Pos. posterior = 0.9731       |
|   242     |    1     |        2.215   |       12.055   |     1489.879   |       Pos. posterior = 0.9131       |
----
## FUBAR inferred 6 sites subject to diversifying positive selection at posterior probability >= 0.9
Of these,  0.36 are expected to be false positives (95% confidence interval of 0-2 )
```

Like other site analyses, FUBAR will print a number of inferences about each individual site detected to be under pervasive positive selection

alpha
: the posterior estimate of the synonymous substitution rate at a site

beta
: the posterior estimate of the non-synonymous substitution rate at a site

N.eff
: an estimate of the effective sample size for inferring positive selection at this site; smaller values (e.g. < 20) imply that the MCMC procedure may have failed to sample the parameter space well, and longer chains (or more chains) might be indicated

Posterior prob for positive selection
: the estimated posterior probability for pervasive diversifying selection ($dN/dS > 1$).

Dataset	Method	Sites under selection at $P \leq 0.1^*$
Full H3	MEME	(15) ***19***, **47**, **61**, **69**, **110**, ***154***, **156**, **173**, ***208***, **236**, **241**, **277**, **278**, **292**, **538**
Full H3	FEL	(15) ***19***, **47**, **61**, **69**, **110**, ***154***, **156**, **173**, **236**, **237**, **241**, **277**, **278**, **292**, **538**
Full H3	SLAC	(20) ***19***, **47**, **61**, **69**, **110**, 137, ***154***, **156**, 158, **173**, 189, ***208***, **236**, **237**, **241**, **277**, **278**, **292**, 505, 546, 564
Full H3	FUBAR	(13) **47**, **61**, **69**, **110**, ***154***, **160**, **173**, ***208***, **236**, **237**, **241**, **278**, **538**
Shallow H3	MEME	(2) **49**, 320
Shallow H3	FEL	(2) **49**, ***241***
Shallow H3	SLAC	*None*
Shallow H3	FUBAR	(3) ***19***, **49**, ***241***
Trunk H3	MEME	(6) **64**, ***154***, 171, ***208***, **242**, 402
Trunk H3	FEL	(3) **64**, ***154***, ***208***
Trunk H3	SLAC	(2) ***154***, ***208***
Trunk H3	FUBAR	(6) **61**, **64**, **69**, ***154***, ***208***, **242**

Table 1: Sites identified as positively selected across the H3 datasets analyzed here. **Bold** sites are those identified by multiple methods for a given dataset. *Bold italicized* sites are those identified in more than one dataset, generally by more than one method. Numbers in parentheses give the total number of positively-selected sites identified with the given method and dataset.
* For FUBAR, significance is assessed as Posterior Probability ≥ 0.9.

Interpreting results Sites identified as positively selected by each method, across all three datasets, are given in Table 1. In general, we expect MEME to be the most comprehensive and robust of all site level methods because it uniquely considers both pervasive and episodic selection (Murrell et al., 2012). In addition, power studies have shown that FUBAR is expected to outperform FEL and SLAC under most circumstances (Murrell et al., 2013). Finally, we expect that SLAC will be the least robust method due to its reliance on a relatively naive counting-based approach (Kosakovsky Pond and Frost, 2005).

These expectations are generally borne out in the results obtained here in our brief study of H3 selection. For the full H3 dataset of 2555 sequences, MEME and FEL each identified 15 sites under positive selection, with all sites identical except for a single difference: MEME uniquely identified site 208 and FEL uniquely identified with 237. Interestingly, site 208 was additionally identified as positively selected by all methods on the trunk H3 dataset. Combined, these results demonstrate MEME's ability to identify sites subject to both pervasive and episodic selection, as site 208 appears to be under pervasive selection only along the H3 trunk. Because FEL uses a less stringent test statistic distribution (χ^2_1) to call significance, occasionally sites subject to pervasive selection near the significance thresholds may be detected by FEL (e.g., 237, with p = 0.08), but missed by MEME (e.g., 237, with p = 0.105).

FUBAR identified two fewer selected sites in the full H3 alignment compared to FEL (which is a directly comparable test), missing sites 19 (posterior 0.83), 277 (posterior 0.59), and 292 (posterior 0.89) relative to FEL, but adding site 160 (FEL p = 0.8).

In addition to differences across methods, we expect to see some important differ-

ences for sites inferred across the the full, shallow, and trunk H3 datasets. Because the trunk and full H3 datasets span similar time frames, we expect sites returned for these two datasets to have the most overlap. In addition, sites found to be under selection in the shallow lineage may not be detected across the full H3 phylogeny, as selection may have been fleeting, weak, or constrained to the specific shallow clade examined here. For example site 49 was specifically selected in the shallow H3 lineage alone, as indicated by three of the four methods. In contrast, sites 19 and 241 were found to be selected in both the shallow and the full H3 datasets, but this signal was not apparent when the trunk lineage was examined independently, perhaps because these sites experience only transient changes that do not propagate along the trunk.

What are some potential reasons for seeing discrepancies in inferences across H3 datasets? The site 154, for example, is positively selected in both the full H3 phylogeny and the trunk H3 lineage, but not the shallow H3 lineage. This result suggests that site 154 may have experienced pervasive selection throughout H3 evolution, but its signal in the shallow clade alone was either too weak to detect or selection was attenuated in the shallow clade. In addition, sites which appeared only in the shallow clade analyses may have experienced lineage-specific selection where the signal was too weak to detect when the entire phylogeny was considered.

Furthermore, while MEME, FEL, and FUBAR were able to detect selected sites in the shallow H3 lineage, SLAC did not identify any such sites, because SLAC requires a large number of substitutions to achieve significance, and those are unlikely to take place at any particular site in the shallow sample.

We emphasize that there in many cases different site-level methods will **not** identify exactly the same set of sites under selection, although, as the H3 example shows, the agreement between is typically good.

Rules of thumb for site-level detection of selection

1. Small datasets, i.e., fewer than 10 sequences especially when coupled with low divergence, are unlikely to yield any sites under selection. Consider using gene-wide methods like BUSTED or aBSREL to look for selection in these cases.

2. On large datasets (e.g. $>$ 500 sequences), all methods tend to give similar results (but see the MEME exception below), hence the default method of choice is FUBAR, since its run time is much shorter than FEL or MEME, and its statistical performance is better than SLAC.

3. MEME tends to be the most sensitive method, because it is the only one designed to detect episodic selection. Indeed, sometimes SLAC, FEL, or FUBAR may all call a site subject to episodic positive selection site negatively selected, if a burst of selection is followed by strong conservation. MEME is often able to tease the two processes apart and correctly call such sites positively selected. Hence, MEME should be the preferred method, unless computationally prohibitive.

4. Should one always run all the available methods on a given dataset and then aggregate the results, as done in Table 1? We don't recommend this approach in all cases. Firstly, while it may be tempting to use agreement between all methods as a hedge against false positives, calling a site selected only if all the methods agreed on it, reduces the power of the analysis to that of the least sensitive method. Secondly, while there is potentially a lot of information to be gleaned by comparing the sites

on which methods disagree (e.g., a site detected by MEME but not FUBAR may be under strong episodic selection), considerable effort and diligence must be put into disentangling biologically meaningful differences from statistical artifacts. Thirdly, statistical strategy must be informed before the analysis commences by deciding what it is that is important to optimize: does one care more about specificity (reducing false positives) or sensitivity (reducing false negatives)? For example if little is known about a gene, it may be advisable to generate the most inclusive list of sites that could be subject to selection for subsequent testing using other approaches; in this case the most sensitive method or the union of all methods may be appropriate.

5. Should one perform multiple testing or false discovery rate correction on individual site results? We don't recommend this. Firstly, methods are calibrated to not generate excessive false positives on strictly neutral data. In most genes, most sites will be under relatively strong negative selection, making the statistical testing procedure conservative. Secondly, multiple testing corrections will nearly always yield no significant results on small to moderate size datasets. Thirdly, some key of assumptions of methods for correcting false discovery rates are not applicable for site level testing: for example, a typical collection of results from site-level testing will contain very few true "null" (neutral) p-values.

Screening sequences for recombination

A critical aspect of sequence analysis we have not yet covered is the detection of and correction for *recombination* in an alignment of homologous sequences. In general, we strongly advocate screening an alignment for recombination before proceeding with additional analyses, unless there is good *a priori* reason to discount intragenic recombination due to the biology of the system (e.g., such recombination is thought to be negligibly rare in Influenza A viruses). Indeed, because recombination causes different regions of an alignment to be related by different phylogenies, its presence can heavily influence selection detection and other downstream applications.

There are many computational approaches to finding evidence of recombination in a sequence alignment (Posada and Crandall, 2001), however at their core, many such methods look for evidence of phylogenetic incongruence. Here, we demonstrate one such method, GARD (Genetic Algorithms for Recombination Detection) that we have found to perform very well among a wide range of approaches on simulated data (Kosakovsky Pond et al., 2006).

GARD

What biological question is the method designed to answer? Have sequences in the given alignment undergone recombination, and if so what are the recombination breakpoints and segment-specific phylogenies?

Recommended applications. GARD is geared towards mapping the breakpoints and detecting segments of the alignment which can be adequately described by a single tree topology. Therefore, alignments, particularly alignments of viral sequences, should be screened for the presence of recombination before performing any selection inference.

The NEXUS output from GARD can be directly used as input for most downstream selection detection analyses.

Statistical test procedure. GARD employs a genetic algorithm to find a solution to a complex optimization problem by mimicking processes of biological evolution (mutation, recombination and selection) in a population of competing solutions. In this application of genetic algorithms, we are evolving a population of "chromosomes" that specify different numbers and locations of recombination breakpoints in the alignment with the objective of detecting topological incongruence, i.e., support for different phylogenies by separate regions of the alignment. The "fitness" of each chromosome is determined by using maximum likelihood methods to evaluate a separate phylogeny for each non-recombinant fragment defined by the breakpoints (e.g. to the left and to the right of a breakpoint in Figure 4), and computing a goodness of fit (AIC_c) for each such model. The genetic algorithm searches for the number and placement of breakpoints yielding the best AIC_c and also reports confidence values for inferred breakpoint locations based on the contribution of each considered model weighted by how well the model fit the data. For computational expedience, the current implementation of GARD infers topologies for each segment using Neighbor Joining (Saitou and Nei, 1987) based on the TN93 pairwise distance estimator (Tamura and Nei, 1993) and then fits a user-specified nucleotide evolutionary model using maximum likelihood to obtain AIC_c scores.

Example Analysis 1 We will demonstrate the use of GARD, as well as its benefits for downstream analysis, using a dataset consisting of 13 glycoprotein sequences from Cache Valley Fever virus (`cvf.fna`). We will first use GARD to detect recombination in this dataset, and then we will process both the GARD-informed data and the original alignment (with no recombination assumed) with FEL to see how the presence of recombination may confound selection inference.

Importantly, GARD specifically requires the use of HyPhy's MPI-enabled executable, HYPHYMPI. To run GARD from the command line, you will need an operating system with a MPI headers and libraries installed so that this executable can be compiled. Here, we will describe how to use GARD from the command line, but we emphasize that GARD is fully implemented and available on `datamonkey.org` and takes the same input options described here.

To run GARD, open a terminal session and start HYPHYMPI in the appropriate MPI environment (e.g. MPIRUN in OpenMPI) from the command line to launch the HyPhy analysis menu. Enter `12` (Recombination) and then `1` to reach the GARD analysis menu, and supply values for the following prompts:

1. **Nucleotide file to screen:**. Provide the full path to the dataset of interest: `/path/to/data/cvf.fna`.
2. **Please enter a 6 character model designation (e.g:010010 defines HKY85)**. This option controls which nucleotide substitution model is to be used for analysis, using PAUP notational shorthand. The six character shorthand allows the user to specify the entire spectrum from F81 (000000) to GTR (012345), which we recommend as default option. Provide the value `012345` for this prompt.
3. **Rate variation options**. This option determines how site-to-site rate variation should be modeled. The option `None` will discount site-to-site rate variation,

allowing the analysis to run several times faster than other options but also creating the risk of mistaking rate heterogeneity for re- combination. As such, we can only recommend this option for extremely small alignments (i.e. 3-5 sequences). The option `General Discrete` (the default) models rate variation using an N bin general discrete distribution, and option `Beta-Gamma` models rate variation using an adaptively discretized distribution, a more flexible version of the standard Gamma+4 model. Enter option `2` to select the General Discrete model.

4. **How many distribution bins [2-32]?**. If rate variation was selected in the previous step, this option allows the user to decide how many different rate classes should be included in the model. We recommend using 3 rate classes by default, as both General Discrete and Beta-Gamma distributions are flexible enough to reliably capture rate variability in the majority of alignments with only a few rate classes. Therefore, enter the value `3`.

5. **Save results to**. For this option, provide a full path to the output file to which you would like GARD to write results. The supplied file name will ultimately contain an HTML-formatted summary of the analysis. HyPhy will generate several other files with names obtained by appending suffixes (as in `<file name>_suffix` to the main result file. In particular, the `_finalout` file stores the original alignment in NEXUS format with inferred non-recombinant sections of the alignment saved in the `ASSUMPTIONS` block and trees inferred for each partition in the `TREES` block. This NEXUS file can be input into many recombination-aware analyses in HyPhy and other programs that can read NEXUS. The `_ga_details` file contains two lines of information about each model examined by the genetic algorithm: its AICc score and the location of breakpoints in the model. Finally, the `_ga_splits` file stores information about the location of breakpoints and trees inferred for each alignment region under the best model found by the GA.

GARD will now run to completion, printing status indicators to screen while it runs:

Listing 8: Partial GARD output

```
Fitting a baseline nucleotide model...
Done with single partition analysis. Log(L) = -5921.9511901113, c-AIC  = 11914.85153276497
Starting the GA...

GENERATION 2 with 1 breakpoints (~0% converged)
Breakpoints    c-AIC  Delta c-AIC [BP      1]
          0 11914.85
          1 11804.56      110.291       1393
GA has considered        92/         328 (92 over all runs) unique models
Total run time           0 hrs 0 mins 2 seconds
Throughput               46.00 models/second
Allocated time remaining 999 hrs 59 mins 58 seconds (approx. 165599908 more models.)
...
GENERATION 52 with 4 breakpoints (~100% converged)
Breakpoints    c-AIC  Delta c-AIC [BP      1] [BP      2] [BP      3] [BP      4]
          0 11914.85
          1 11804.56      110.291       1445
          2 11783.92       20.638        617        1490
          3 11778.94        4.978        587         962        1475
          4 11778.94        0.000        587         962        1475
GA has considered        268/   473490550 (1356 over all runs) unique models
Total run time           0 hrs 4 mins 2 seconds
Throughput                5.60 models/second
Allocated time remaining 999 hrs 55 mins 58 seconds (approx. 20170544.82644628 more models.)
Performing the final optimization...
```

Interpreting results GARD found evidence of recombination in this data set with three breakpoints, yielding a 135.9 point AIC_c improvement over the model without recombination. Among all models with three breakpoints in the Cache Valley Virus

glycoprotein alignment, the best model places them at nucleotides 587, 962, and 1475. The score Importantly, if GARD had reported that the best model had 0 breakpoints, we could conclude that no evidence of recombination had been found. Note that because genetic algorithms are stochastic, there is no guarantee that replicate runs will converge to exactly the same quantitative results. When there is a strong signal of recombination breakpoints in the data, however, the qualitative results (number and general location of breakpoints) should be fairly robust.

Example Analysis 2 The NEXUS file that GARD produced is a *partitioned dataset*, wherein different groups of sites are be described by different trees. Most HyPhy selection analyses discussed here[4], including MEME, FUBAR, FEL, SLAC, and BUSTED, are able to analyze partitioned data. To demonstrate the importance of screening for recombination, we will now compare results for a FEL analysis performed on the original alignment of 13 Cache Valley Virus glycoproteins, as well as on the GARD-inferred partitioned alignment. All steps here were carried out as described earlier in this chapter.

Interpreting results FEL inference on the GARD-processed partitioned Cache Valley Virus data does not detect sites under selection at $P \leq 0.1$. By contrast, FEL inference on the unpartitioned Cache Valley Virus data (i.e. not pre-screened for recombination) detects three positively selected sites at $P \leq 0.1$ (212, 516, and 558 at $P = 0.08$, $P = 0.03$, and $P = 0.09$, respectively). From these results, we can clearly tell that not screening for recombination has the potential for adverse consequences including an increased false positive rate as seen here. As such, we strongly encourage users to screen alignments for recombination if such processes are suspected before proceeding to selection detection.

Synonymous rate variation

We demonstrate the importance of considering synonymous rate variation for selection inference using a dataset of 10 mammalian CD2 genes, which code for a specific T-cell surface adhesion molecule (Lynn et al., 2005). We use FEL to detect selection in this dataset under two specifications: with synonymous rate variation ("Yes" in prompt 4 in the FEL analysis menu), and without synonymous rate variation ("No" in prompt 4 in the FEL analysis menu).

Interpreting Results At $P \leq 0.1$, analysis of CD2 *with* synonymous rate variation revealed a total of 14 sites under positive selection. By contrast, CD2 analysis with FEL without dS variation only detected 4 sites under positive selection. Similarly, analysis with dS variation revealed 27 sites under purifying selection, but analysis without dS variation revealed only 15 sites under purifying selection. Most importantly, all sites detected when dS was fixed to 1 were a subset of the sites identified by the model with dS variation (Figure 5). Together, these results demonstrate that ignoring dS variation can induce both an increased false negative rate for identifying sites under positive selection as well as an overall decrease in power to detect any selective regime.

[4]Note that neither aBSREL nor RELAX accept partitioned data because they require a consistent phylogeny to define branch sets

We acknowledge that it is possible that the opposite conclusion might be true, namely that that additional sites identified by FEL with dS variation might instead be false positives. However, in our experience, this is much less frequently the case (Kosakovsky Pond and Frost, 2005).

Tips and tricks

Here we provide some helpful notes on HyPhy usage.

- An actively-maintained board for usage questions and filing bug reports is available at `https://github.com/veg/hyphy/issues`.
- Each HyPhy analysis described here will export a JSON file. This file can either be uploaded to **HyPhy Vision** for visual examination, or it can be easily parsed using a standard scripting language using standard packages, for example the `json` package in Python or the `jsonLite` package in R. All fields used in these output files are defined [AT THIS LINK].
- Mac OS(X) users may need to install a new set of compilers (i.e. gcc-6) that are compatible with openMP in order to have full functionality from the HYPHYMP executable, as is described on the HyPhy website.

Exercises

1. Earlier, we performed a BUSTED analysis without designating a specific subset of test lineages. For this exercise, we will analyze the HIV-1 transmission dataset with BUSTED two different ways: testing all branches, and testing only recipient-derived HIV-1 sequences. The input data for this exercise, with an appropriately labeled phylogeny, is available in `exercises/hiv1_transmission_exercise1.fna`. For select branches labeled `All` or `test` as the test lineages.
    - Is there evidence (compare model fits using the small sample AIC) that `test` branches have a different selective regime than the rest of the tree?
    - The entire data set should provide evidence for episodic diversification, but the recipient only analysis should return a negative result. What does this mean biologically, i.e. where does the selection signal come from?
2. Investigate the effect of recombination of site-specific inference of episodic selection using MEME. Run MEME on `exercises/cvf.fna` (single partition data, i.e. assuming no recombination), and then on the same dataset screened for recombination using GARD `exercises/cvf-gard.nex`, testing for selection on `all` branches, with `P=0.1`. Compare the list of sites detected to be under selection by the two analyses.
    - Which analysis generated more positive results?
    - Do you think these results are true of false positives? How does this compare to the FEL analysis we described in the text?
    - Compare site-wise estimates of substitution rates (e.g. α), between the two analyses. Is there a discernible bias introduced by not accounting for recombination.

3. When analyzing intra-species or intra-host data, dN/dS estimates may be inflated due to the fact that not all observed sequence variation is due to substitutions, but some are simply mutations that have not yet been filtered by selection (e.g. Pond et al. (2006)). In other words, dN/dS may be elevated by intra-species / intra-host polymorphism that need not be attributable by positive selection. One simple approach to mitigating this undesirable effect is to restrict site-specific analyses to `Internal` branches only. This is because internal branches encompass at least one step that is visible to selection (transmission and/or multiple rounds of replication), and are less likely to contain spurious polymorphic variants. Apply MEME and FEL to an intra-host sample of HIV-1 sequences from an infected individual analyzed in Lorenzo-Redondo et al. (2016), first choosing to test `All` branches, and next choosing `Internal` branches.
4. Compare the lists of selected sites between All/Internal analyses. How different are they?
5. Use RELAX to formally test whether or not selective regimes (dN/dS distributions) are different between terminal and internal branches.

References

M. Anisimova and C. Kosiol. Investigating protein-coding sequence evolution with probabilistic codon substitution models. *Mol. Biol. Evol.*, 26:255–271, 2009.

J. V. Chamary and L. D. Hurst. Evidence for selection on synonymous mutations affecting stability of mRNA secondary structure in mammals. *Genome Biol.*, 6:R75, 2005.

E. R. Chare and E. C. Holmes. A phylogenetic survey of recombination frequency in plant RNA viruses viruses. *Arch. Virol.*, 151(5):933–946, 2006.

E. R. Chare, E. A. Gould, and E. C. Holmes. Phylogenetic analysis reveals a low rate of homologous recombination in negative-sense RNA viruses. *J. Gen. Virol.*, 84, 2003.

W. Delport, K. Scheffler, and C. Seoighe. Models of coding sequence evolution. *Briefings in Bioinformatics*, 10(1):97–109, 2009. doi: 10.1093/bib/bbn049. URL `+http://dx.doi.org/10.1093/bib/bbn049`.

D. A. Drummond and C. O. Wilke. Mistranslation-induced protein misfolding as a dominant constraint on coding-sequence evolution. *Cell*, 134:341 – 352, 2008.

D. Enard, L. Cai, C. Gwennap, and D. A. Petrov. Viruses are a dominant driver of protein adaptation in mammal. *eLife*, 5:e12469, 2016.

D. Forni, R. Cagliani, M. Clerici, and M. Sironi. Molecular evolution of human coronavirus genomes. *Trends in Microbiology*, 25(1):35 – 48, 2017. ISSN 0966-842X. doi: https://doi.org/10.1016/j.tim.2016.09.001. URL `http://www.sciencedirect.com/science/article/pii/S0966842X16301330`.

S. D. W. Frost, Y. Liu, S. L. Kosakovsky Pond, C. Chappey, T. Wrin, C. J. Petropoulos, S. J. Little, and D. D. Richman. Characterization of Human Immunodeficiency Virus Type 1 (HIV-1) envelope variation and neutralizing antibody responses during transmission of HIV-1 Subtype B. *J. Virol.*, 79:6523–6527, 2005.

N. Goldman and Z. Yang. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Mol Biol Evol*, 11:725 – 736, 1994.

K. M. Graef, F. T. Vreede, Y.-F. Lau, A. W. McCall, S. M. Carr, K. Subbarao, and E. Fodor. The PB2 subunit of the Influenza virus RNA Polymerase affects virulence by interacting with the mitochondrial antiviral signaling protein and inhibiting expression of beta interferon. *J. Virol.*, 84:8433–8445, 2010.

R. J. Harrison and B. Charlesworth. Biased gene conversion affects patterns of codon usage and amino acid usage in the Saccharomyces sensu stricto group of yeasts. *Mol. Biol. Evol.*, 28:117–129, 2011.

R. Hershberg and D. Petrov. Selection on codon bias. *Annu. Rev. Genet.*, 42, 2008.

S. L. Kosakovsky Pond and S. W. D. Frost. Not so different after all: A comparison of methods for detecting amino acid sites under selection. *Mol. Biol. Evol.*, 22:1208–1222, 2005.

S. L. Kosakovsky Pond and S. V. Muse. Site-to-site variation of synonymous substitution rates. *Mol. Biol. Evol.*, 22:2375–2385, 2005.

S. L. Kosakovsky Pond, D. Posada, M. B. Gravenor, C. H. Woelk, and S. D. W. Frost. Automated phylogenetic detection of recombination using a genetic algorithm. *Mol. Biol. Evol.*, 23(10):1891–901, 2006.

S. L. Kosakovsky Pond, W. Delport, S. V. Muse, and K. Scheffler. Correcting the bias of empirical frequency parameter estimators in codon models. *PLOS ONE*, 5:e11230, 2010.

S. L. Kosakovsky Pond, B. Murrell, M. Fourment, S. D. Frost, W. Delport, and K. Scheffler. A random effects branch-site model for detecting episodic diversifying selection. *Mol. Biol. Evol.*, 28:3033—3043, 2011.

K. Labadie, E. Dos Santos Afonso, M.-A. Rameix-Welti, S. van der Werf, and N. Naffakh. Host-range determinants on the PB2 protein of influenza A viruses control the interaction between the viral polymerase and nucleoprotein in human cells. *Virology*, 362:271–282, 2007.

R. Lorenzo-Redondo, H. R. Fryer, T. Bedford, E.-Y. Kim, J. Archer, S. L. K. Pond, Y.-S. Chung, S. Penugonda, J. G. Chipman, C. V. Fletcher, T. W. Schacker, M. H. Malim, A. Rambaut, A. T. Haase, A. R. McLean, and S. M. Wolinsky. Persistent hiv-1 replication maintains the tissue reservoir during therapy. *Nature*, 530(7588): 51–6, Feb 2016. doi: 10.1038/nature16933.

M. Lynch, M. S. Ackerman, J.-F. Gout, H. Long, W. Sung, W. K. Thomas, and P. L. Foster. Genetic drift, selection and the evolution of the mutation rate. *Nature*, 17 (11):704–714, 2016.

D. J. Lynn, A. R. Freeman, C. Murray, and D. G. Bradley. A genomics approach to the detection of positive selection in cattle: adaptive evolution of the t-cell and natural killer cell-surface protein cd2. *Genetics*, 170(3):1189–1196, 2005.

A. G. Meyer and C. O. Wilke. Geometric constraints dominate the antigenic evolution of Influenza H3N2 Hemagglutinin. *PLOS Pathogens*, 11:e1004940, 2015.

B. Murrell, J. O. Wertheim, S. Moola, T. Weighill, K. Scheffler, and S. L. Kosakovsky Pond. Detecting individual sites subject to episodic diversifying selection. *PLOS Genet.*, 8(7):e1002764, 2012.

B. Murrell, S. Moola, A. Mabona, T. Weighill, D. Scheward, S. L. Kosakovsky Pond, and K. Scheffler. FUBAR: A Fast, Unconstrained Bayesian AppRoximation for inferring selection. *Mol. Biol. Evol.*, 30:1196–1205, 2013.

B. Murrell, S. Weaver, M. D. Smith, J. O. Wertheim, S. Murrell, A. Aylward, K. Eren, T. Pollner, D. P. Martin, D. M. Smith, K. Scheffler, and S. L. Kosakovsky Pond. Gene-wide identification of episodic selection. *Mol. Biol. Evol.*, 32:1365–1371, 2015.

S. V. Muse and B. S. Gaut. A likelihood approach for comparing synonymous and non-synonymous nucleotide substitution rates, with application to the chloroplast genome. *Mol. Biol. Evol.*, 11:715–724, 1994.

M. I. Nelson and E. C. Holmes. The evolution of epidemic influenza. *Nat Rev Genet*, 8 (3):196–205, 03 2007. doi: 10.1038/nrg2053.

R. Nielsen and Z. Yang. Likelihood models for detecting positive selected amino acid sites and applications to the HIV-1 envelope gene. *Genetics*, 148:929–936, 1998.

J. B. Plotkin and G. Kudla. Synonymous but not the same: the causes and consequences of codon bias. *Nature Rev. Genet.*, 12:32–42, 2011.

S. L. K. Pond, S. D. W. Frost, Z. Grossman, M. B. Gravenor, D. D. Richman, and A. J. L. Brown. Adaptation to different human populations by hiv-1 revealed by codon-based analyses. *PLoS Comput Biol*, 2(6):e62, Jun 2006. doi: 10.1371/journal.pcbi.0020062.

D. Posada and K. A. Crandall. Evaluation of methods for detecting recombination from DNA sequences: computer simulations. *Proc. Natl. Acad. Sci. U. S. A.*, 98(24): 13757–13762, 2001.

D. Posada, K. A. Crandall, and E. C. Holmes. Recombination in evolutionary genomics. *Annu. Rev. Genet.*, 36:75–97, 2002.

S. A. Price. Comparative genomics of amphibian-like Ranaviruses, nucleocytoplasmic large DNA viruses of Poikilotherms. *Evol. Bioinform. Online.*, 11:71–82, 2015.

N. Rodrigue. On the statistical interpretation of site-specific variables in phylogeny-based substitution models. *Genetics*, 193:557–564, 2013.

N. Saitou and M. Nei. The neighbor-joining method: a new method for reconstructing phylogenetic trees. *Mol Biol Evol*, 4(4):406–25, Jul 1987.

T. E. Schlub, R. P. Smyth, A. J. Grimm, J. Mak, and M. P. Davenport. Accurately measuring recombination between closely related HIV-1 genomes. *PLoS Comput. Biol.*, 6(4):e1000766, 2010.

R. S. Sealfon, M. F. Lin, I. Jungreis, M. Y. Wolf, M. Kellis, and P. C. Sabeti. FRESCo: finding regions of excess synonymous constraint in diverse viruses. *Genome Biology*, 16(1):38, 2015.

D. M. Smith, S. J. May, S. Tweeten, L. Drumright, M. E. Pacold, S. L. Kosakovsky Pond, R. L. Pesano, Y. S. Lie, D. D. Richman, S. D. W. Frost, C. H. Woelk, and S. J. Little. A public health model for the molecular surveillance of HIV transmission in San Diego, California. *AIDS*, 23(2):225–232, 2009.

G. J. D. Smith, D. Vijaykrishna, J. Bahl, S. J. Lycett, M. Worobey, O. G. Pybus, S. K. Ma, C. L. Cheung, J. Raghwani, S. Bhatt, J. S. M. Peiris, Y. Guan, and A. Rambaut. Origins and evolutionary genomics of the 2009 swine-origin H1N1 influenza A epidemic. *Nature*, 459(7250):1122–1125, 2010.

M. D. Smith, J. O. Wertheim, S. Weaver, B. Murrell, K. Scheffler, and S. L. Kosakovsky Pond. Less is more: an adaptive branch-site random effects model for efficient detection of episodic diversifying selection. *Mol. Biol. Evol.*, 32:1342–1353, 2015.

W. Sung, M. S. Ackerman, J.-F. Gout, S. F. Miller, E. Williams, P. L. Foster, and M. Lynch. Asymmetric context-dependent mutation patterns revealed through mutation–accumulation experiments. *Mol. Biol. Evol.*, 32(7):1672–1683, 2015.

Y. Suzuki and T. Gojobori. A method for detecting positive selection at single amino acid sites. *Mol. Biol. Evol.*, 16:1315–1328, 1999.

K. Tamura and M. Nei. Estimation of the number of nucleotide substitutions in the control region of mitochondrial dna in humans and chimpanzees. *Mol. Biol. Evol.*, 10, 1993.

A. U. Tamuri, M. dos Reis, A. J. Hay, and R. A. Goldstein. Identifying changes in selective constraints: Host shifts in influenza. *PLOS Comput. Biol.*, 5(11):e1000564, 2009.

A. U. Tamuri, M. dos Reis, and R. A. Goldstein. Estimating the distribution of selection coefficients from phylogenetic data using sitewise mutation–selection models. *Genetics*, 190:1101–1115, 2012.

B. S. Taylor, M. E. Sobieszczyk, F. E. McCutchan, and S. M. Hammer. The challenge of HIV-1 subtype diversity. *N. Engl. J. Med.*, 358(15):1590–1602, 2008.

D. C. Tully, C. B. Ogilvie, R. E. Batorsky, D. J. Bean, K. A. Power, M. Ghebremichael, H. E. Bedard, A. D. Gladden, A. M. Seese, M. A. Amero, K. Lane, G. McGrath, S. B. Bazner, J. Tinsley, N. J. Lennon, M. R. Henn, Z. L. Brumme, P. J. Norris, E. S. Rosenberg, K. H. Mayer, H. Jessen, S. L. Kosakovsky Pond, B. D. Walker, M. Altfeld, J. M. Carlson, and T. M. Allen. Differences in the selection bottleneck between modes of sexual transmission influence the genetic composition of the hiv-1 founder virus. *PLoS Pathog*, 12(5):e1005619, May 2016. doi: 10.1371/journal.ppat.1005619.

J. O. Wertheim, B. Murrell, M. D. Smith, S. L. Kosakovsky Pond, and K. Scheffler. RELAX: detecting relaxed selection in a phylogenetic framework. *Mol. Biol. Evol.*, 32(3):820–832, 2015.

M. Worobey and E. C. Holmes. Evolutionary aspects of recombination in RNA viruses. *J Gen. Virol.*, 80(10):2535–2543, 1999.

Z. Yang. *Computational Molecular Evolution*. Oxford University Press, 2006.

Z. Yang and R. Nielsen. Codon-substitution models for detecting molecular adaptation at individual sites along specific lineages. *Mol. Biol. Evol.*, 19:908–917, 2002.

J. Zhang, R. Nielsen, and Z. Yang. Evaluation of an improved branch-site likelihood method for detecting positive selection at the molecular level. *Mol. Biol. Evol.*, 22: 2472–2479, 2005.

Figures

Figure 1: Example analysis visualization in HyPhy-Vision of BUSTED results. (A) The **summary** section provides a brief overview of the analysis performed, including information about the inputted data (which can be downloaded via the linked file name) and primary results from the hypothesis test performed. (B) The **model statistics** section provides information about models fitted to the data. In BUSTED, this section additionally includes an interactive display of site evidence ratios, which can be interpreted as a *descriptive* measure for which sites may have contributed to the selection signal. (C) The **tree** section displays the phylogeny as fitted under all inferred models and data partitions, if specified. Tree views can be toggled under the *Options* dropdown menu. (D) Graphical views of each model's inferred ω distribution can be viewed when clicking on a given row in the **Model fits** table seen in (B).

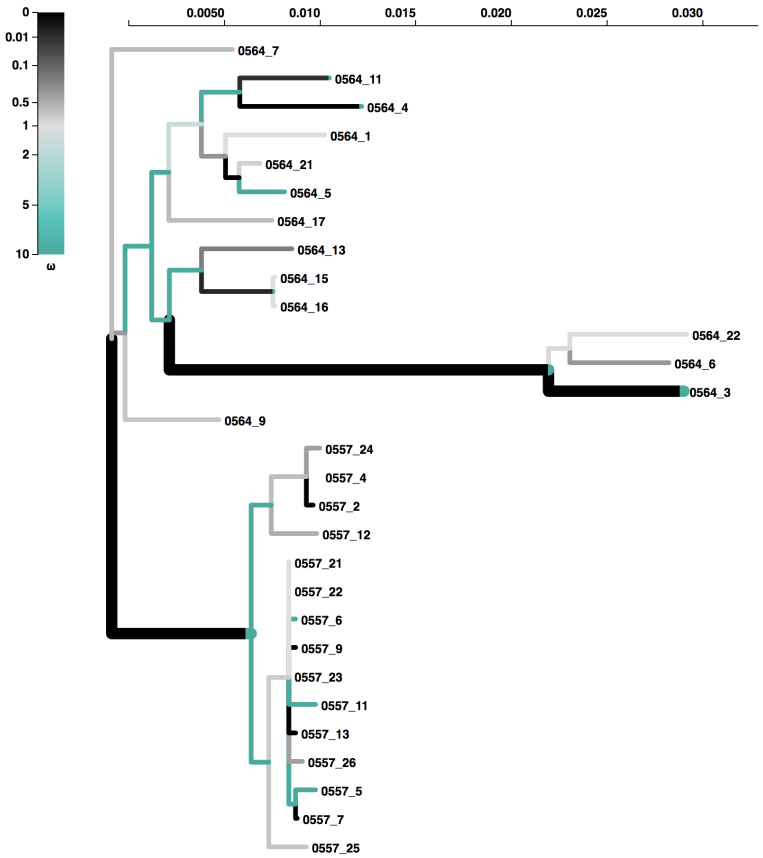

Figure 2: HyPhy-Vision tree viewer depicting the fitted aBSREL Adaptive model to HIV-1 data. Branches are colored by their inferred ω distribution, as indicated in the legend. Lineages identified as positive selection at $P < 0.05$ after correction for multiple testing are shown with thick branches, with color representing the relative proportions of inferred ω categories. Note that taxon labels beginning with '0554' represent HIV-1 sequences derived from the donor patient, and labels beginning with '0557' represent HIV-1 sequences derived from the recipient patient.

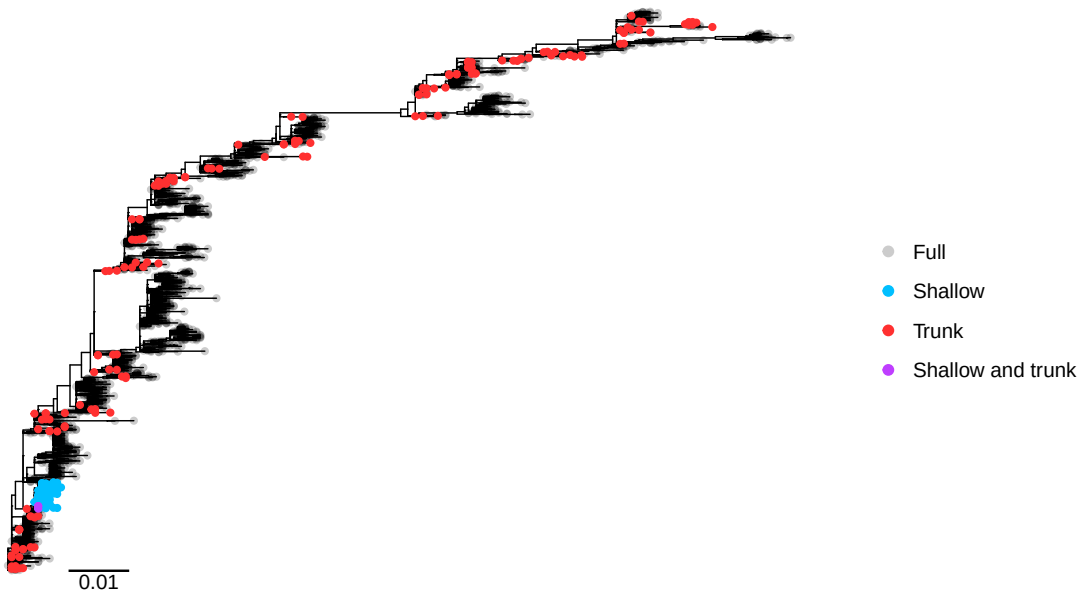

Figure 3: Phylogeny of H3 hemagglutinin sequences analyzed here. Tip colors indicate those selected for each dataset.

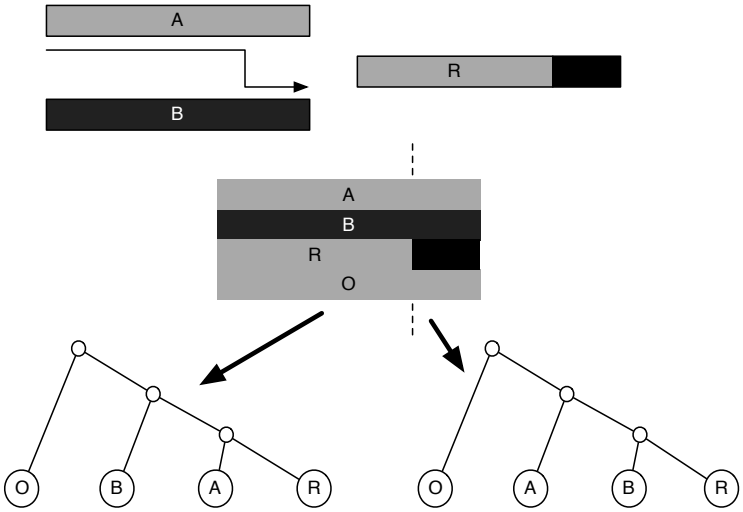

Figure 4: Phylogenetic incongruence caused by the presence of a recombinant sequence in an alignment. Sequence R is a product of homologous recombination between sequences A and B. Phylogenies reconstructed from sequences A,B,R and an outgroup sequence (O) will differ based on which part of the alignment is being consideredl to the left of the breakpoint, R clusters with A, whereas to the right of the breakpoint R clusters with B.

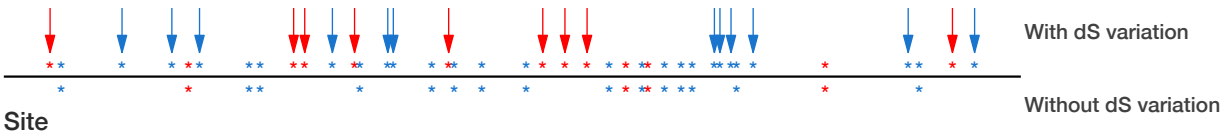

Figure 5: Sites identified as positively (red) and negatively (blue) selected in CD2 at $P \leq 0.1$ by FEL run with (above the line) and without dS variation (below the line). Sites with arrows represent those identified as selected by FEL with dS variation but that were *not* identified by FEL when dS variation was ignored.