JaCK & Jill: exact marginal likelihoods for small phylogenetic trees

Matthew D. Schwartz[1], Scott V. Edwards[2,3], and Paul O. Lewis[4]

[1]*Department of Physics, Harvard University*
[2]*Department of Organismic and Evolutionary Biology, Harvard University*
[3]*Museum of Comparative Zoology, Harvard University*
[4]*Department of Ecology and Evolutionary Biology, University of Connecticut*

October 1, 2026

This work is preliminary. Some content remains AI-generated and is still being validated.

PRELIMINARY

Abstract

In Bayesian phylogenetics, substitution models, and for a few taxa competing trees, are compared by their marginal likelihoods, the probability of the alignment under each model integrated over a prior on branch lengths and parameters. Current programs estimate this integral by sampling, with no bound on how far the result lies from the true value. Marginal likelihoods on four or five taxa arise throughout phylogenomics, as when species trees are assembled from quartets or models are chosen locus by locus. For four taxa no estimate is needed, because under the Jukes–Cantor and Kimura models with the branch-length priors used here the marginal likelihood is a ratio of two integers. Here, we present JaCK & Jill (the Python package phyloexact, version 0.2.3), which computes that ratio under the Jukes–Cantor model for alignments of transfer RNA length and under Kimura's two-parameter model for shorter ones. Beyond those limits it computes an estimate at any length, whose error we have measured against exact values on small alignments. Under HKY85 and GTR+Γ it returns an estimate or, at fitted parameters, an interval proven to contain the value. When a symmetry of the site-pattern counts forces two topologies to tie, it proves the tie, which no two estimates, however close, can establish. On transfer RNA quartets and quintets with exact answers, at equal single-thread run time its estimate is twenty to two hundred times more accurate than MrBayes, RevBayes and LoRaD, and it reaches a given accuracy in seconds where those samplers need minutes to hours. The package also compares models across loci without assuming a tree, scans genomes window by window and accepts user-written models.

1 Introduction

In Bayesian phylogenetics the choice between substitution models, and for a few taxa between tree topologies, rests on the marginal likelihood (Kass and Raftery, 1995; Oaks et al., 2019). The marginal likelihood of a model is the probability of the alignment under that model, averaged over its branch lengths and parameters under a stated prior. Two models, or two topologies, are compared by the ratio of their marginal likelihoods, the Bayes factor. The programs in common use estimate marginal likelihoods by sampling. MrBayes (Ronquist et al., 2012) provides stepping-stone sampling (Xie et al., 2011); RevBayes (Höhna et al., 2016) and BEAST 2 (Bouckaert et al.,

2019) add path sampling, also called thermodynamic integration (Lartillot and Philippe, 2006; Baele et al., 2012). The program physher (Fourment and Holmes, 2014) implements generalized stepping-stone sampling (Fan et al., 2011), nested sampling (introduced in BEAST 2 by Maturana Russel et al., 2019) and the rest of the nineteen methods compared by Fourment et al. (2020); LoRaD (Wang et al., 2023) has appeared since. None of these programs can say how far its estimate lies from the true value; the uncertainties they report, if any, are Monte Carlo standard errors. Accuracy has therefore been judged by comparing estimators with one another or with longer runs (Oaks et al., 2019; Fourment et al., 2020).

Problems on only four or five taxa, or on one locus at a time, arise throughout phylogenomic analyses. Species-tree methods such as ASTRAL-III (Zhang et al., 2018), wASTRAL (Zhang and Mirarab, 2022) and wQFM (Mahbub et al., 2021) build large trees by combining quartet topologies, and genome scans for introgression estimate which quartet topology each chromosome window favors (Fontaine et al., 2015; Martin and Van Belleghem, 2017). Before tree building, substitution models are commonly chosen locus by locus (Posada and Crandall, 1998; Kalyaanamoorthy et al., 2017; Lanfear et al., 2017). With four taxa the three possible topologies can be enumerated, and under a uniform prior on topologies their marginal likelihoods determine the posterior probability of each topology directly. Sampling estimators also have settings whose effect can be measured only against a known answer (Xie et al., 2011), which an exact value on a real alignment provides. For problems of this size the marginal likelihood need not be estimated. Under the Jukes–Cantor (JC69) model (Jukes and Cantor, 1969) and Kimura's two-parameter (K2P) and three-parameter (K3ST or K81) models (Kimura, 1980, 1981), each site-pattern probability on a fixed topology is a polynomial in exponentially transformed branch lengths (Hendy and Penny, 1989, 1993; Székely et al., 1993; Sturmfels and Sullivant, 2005). With branch-length priors uniform in those variables (for the Jukes–Cantor model, an exponential prior on each branch length), the marginal likelihood on a fixed topology is a ratio of two integers. A companion paper shows how to compute it for four- and five-taxon alignments roughly the length of a transfer RNA (tRNA) gene (Schwartz et al., 2026).

Here, we present JaCK & Jill, distributed as the Python package phyloexact (version 0.2.3), which computes these exact values for four taxa. In the name, JaCK stands for the exact Jukes–Cantor and Kimura computations, Jill for the interval and sampling routines. Given an alignment, a topology and a model, phyloexact returns one of four kinds of answer: an exact value, a proven interval, an estimate with a measured error, or a proof that two topologies tie. Because two estimates, however close, cannot establish such a tie, phyloexact proves it from a symmetry of the site-pattern counts. The package computes exact values under the Jukes–Cantor model and, on short alignments, under Kimura's two-parameter model with a separate transition–transversion ratio on each branch, and proven intervals under HKY85 (Hasegawa et al., 1985) and GTR+Γ (Tavaré, 1986; Yang, 1994) at fitted or user-supplied parameters. For all four models it also computes an importance-sampling estimate at any alignment length. Under the Jukes–Cantor and two-parameter models we have measured the estimate's error against exact values on small alignments. Each number comes with a certificate, a JSON document stating the model, the prior and the kind of answer, which an independent routine can test against the alignment. Further calls return Bayes factors and measures of model fit, compare substitution models across many loci without assuming a tree, scan genome alignments window by window, and accept user-written models.

The topology is always fixed; phyloexact performs no tree search and does not model the multispecies coalescent. All four kinds of answer are available for four taxa; for five or more taxa phyloexact provides only the Jukes–Cantor estimate and, for up to 32 taxa, tie proofs. As we

PRELIMINARY

show below on nine tRNA problems with exact answers, at its default budget the estimate runs in one to four seconds on one thread with an error below 10^{-3} log units, and at equal run time it is twenty to two hundred times more accurate than stepping-stone sampling in MrBayes and RevBayes and than LoRaD.

2 Four kinds of answer

The package phyloexact reports each marginal likelihood as one of four kinds of answer (Fig. 1). The marginal likelihood of alignment D on a fixed topology T under model M is

$$Z(D \mid T, M) \;=\; \int P(D \mid \mathbf{t}, \boldsymbol{\theta}, T, M)\, \pi(\mathbf{t})\, \pi(\boldsymbol{\theta})\, \mathrm{d}\mathbf{t}\, \mathrm{d}\boldsymbol{\theta}, \qquad \pi(\mathbf{t}) \;=\; \prod_{e=1}^{5} \tfrac{4}{3}\, e^{-4t_e/3}. \tag{1}$$

Here $\mathbf{t}$ holds the five branch lengths (unrooted tree, no clock) in expected substitutions per site, $\boldsymbol{\theta}$ the free parameters of M (none for JC69), and P the product of site-pattern probabilities over the N columns of D (Felsenstein, 1981). The default prior, exponential with rate 4/3 on each branch, makes $x_e = e^{-4t_e/3}$ uniform on $(0, 1)$.

Exact value. Under JC69 (Jukes and Cantor, 1969) and Kimura's two models (Kimura, 1980, 1981), here with branch-specific rate parameters (see Models), each site-pattern probability is a polynomial with rational coefficients in transformed branch lengths such as the x_e (Hendy and Penny, 1989, 1993; Székely et al., 1993; Sturmfels and Sullivant, 2005). Thus, with priors uniform in those variables, Z is a ratio of two integers, and phyloexact computes both by term-by-term integration in exact rational arithmetic (Schwartz et al., 2026). For the mitochondrial tRNA-Gln alignment of mouse, rat, chicken and *Xenopus*, the JC69 marginal likelihood of the mouse-rat topology is a 188-digit integer over a 302-digit one ($\ln Z = -261.40306$; six minutes on one thread of the benchmark server under full load). The cost grows steeply with m (roughly m^5 at short N), where $m \le N$ is the number of columns in which the two taxa of either sister pair of the topology differ; here $N = 68$ and $m = 25$.

Proven interval. Under HKY85 and GTR+Γ (Hasegawa et al., 1985; Tavaré, 1986; Yang, 1994) the site-pattern probabilities are not polynomials of bounded degree in any transformed branch length, and the integer computation does not apply. Instead phyloexact computes a proven interval, whose endpoints bracket Z and are evaluated in ball arithmetic (Arb; Johansson, 2017), where every intermediate quantity carries a rigorous bound on its rounding error. We fix the substitution parameters at fitted or user-supplied values and integrate over branch lengths only. The interval is computed under an adjustable time limit, and a run that reaches the limit may prove only the upper endpoint; at tRNA length a usable two-sided width takes hours on tens of processes (see Costs and limits).

Estimate with a measured error. The package also computes, for JC69, K2P, HKY85 and GTR+Γ at any alignment length, a randomized quasi-Monte Carlo importance-sampling estimate (Owen, 1997; Dick et al., 2013). Its budget of likelihood evaluations (10^5 by default) is shared by a pilot search for the posterior modes and four independent replicates. The error figure reported with the estimate is three times the largest deviation from exact values that we measured on alignments in the same size class as the input (a range of N and m). Where three times the replicate spread (the range of the four replicate values) exceeds this class tolerance, the error

PRELIMINARY

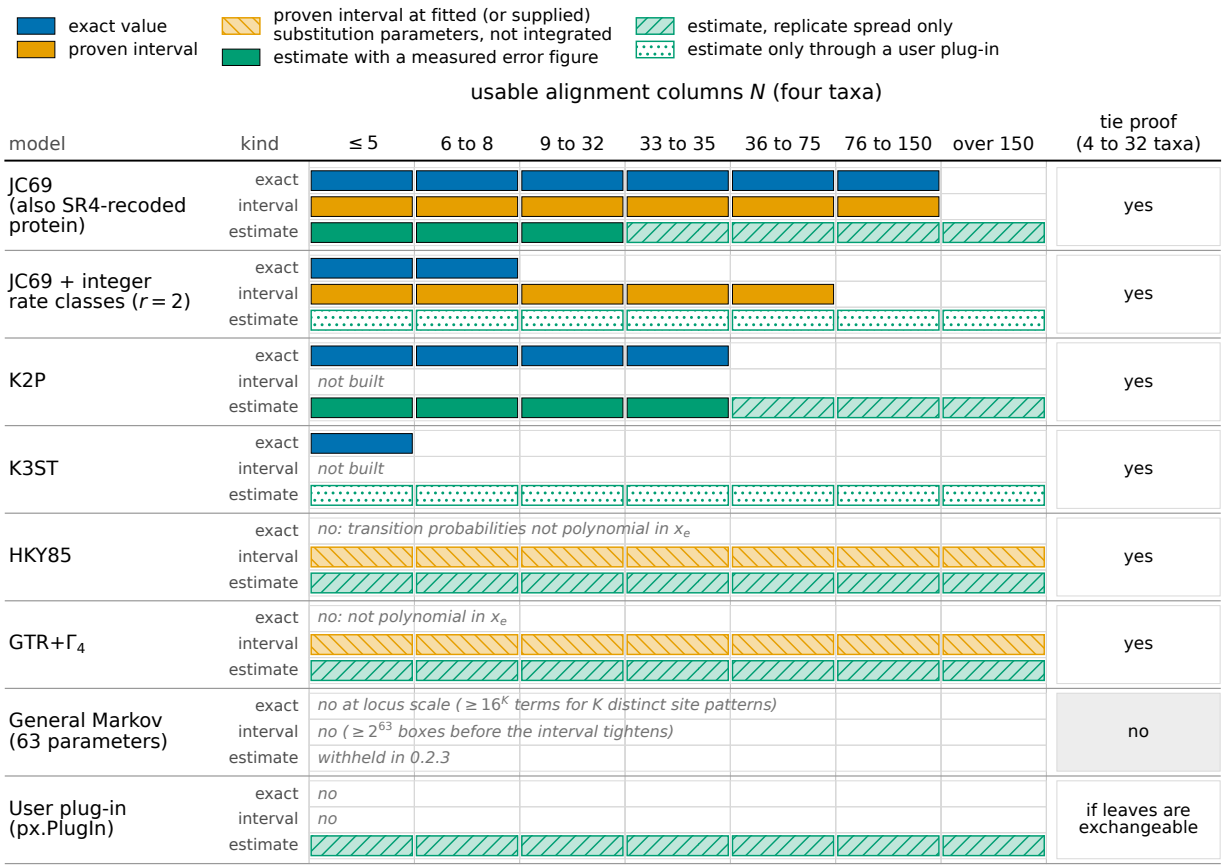

Figure 1: Kinds of answer in phyloexact 0.2.3 by model and usable columns N on four taxa (Table 2). In practice, exact values are confined to JC69 (by default inside the m limits of Fig. 5b) and K2P, and fully integrated intervals to JC69 and its rate-class extension; error figures exist only at the smallest sizes. Hatching as in the legend; blank cells mark kinds not offered.

figure is that product instead. It is an indicative accuracy measured on that class and not a bound (Supplement S3); when the spread itself exceeds the class tolerance, no error figure is given. For JC69 quartets at the default budget it is 0.014 in natural-log units (log units below; the program prints them as nats), and the measured size class extends to 32 columns and $m = 9$. A larger or more diverged alignment falls outside that class, and phyloexact then reports the estimate and its replicate spread but no error figure. The JC69 size class lies within the sizes at which the exact value is itself available, and the K2P class ($N \leq 35$, $m \leq 11$) nearly so (Table 2).

Proof of a tie. Suppose a permutation of the taxa leaves the site-pattern counts unchanged and maps T onto T'. Then $Z(D \mid T, M) = Z(D \mid T', M)$ for every model and prior that the permutation leaves invariant, in particular whenever every branch carries the same substitution model, with the branch lengths and any branch-specific parameters independent and identically distributed a priori (Schwartz et al., 2026). Such a permutation leaves every built-in model and its prior invariant except the general Markov model, whose prior is placed on a rooted tree and can change when the permutation moves the root. For that model phyloexact offers no tie proof (Table 2). Under JC69, which treats the four bases alike, it suffices that the permutation

PRELIM

preserve the number of columns showing each partition of the taxa by shared state. Because agreeing floating-point estimates cannot establish a tie, phyloexact searches for such a permutation before computing either marginal likelihood and, if it finds one, reports the permutation as the proof. The tRNA-Phe quartet of human, chimpanzee, gorilla and orangutan admits such a permutation: exchanging human and gorilla preserves the partition counts, and under JC69 the human-chimpanzee and gorilla-chimpanzee topologies tie.

Choice of kind. By default phyloexact returns a tie proof for a tied comparison, the exact value inside the size limits of Table 2, and otherwise the estimate. For JC69 we set those limits from measured timings such that, inside them, one thread of the benchmark server (see Accuracy and speed) computes the exact value within eight minutes, whereas the estimate at the default budget takes seconds. A user can instead request a specific kind, including an interval, and then receives either that kind or an error that states the obstacle (a theorem, a size limit or a measured cost) and the nearest request that would succeed.

3 Interface

An alignment is read with `px.Dataset.from_fasta` from a FASTA file (Pearson and Lipman, 1988), the one alignment format phyloexact reads (NEXUS and PHYLIP files are first converted to FASTA; Supplement S1.2), or is built from in-memory sequences or from site-pattern counts (Table 1; Fig. 2 shows a session). We drop every column that contains a gap or an ambiguity code when the alignment is read, and the number dropped is reported with each result. The order of the sequences in the file does not affect any value returned. Users specify a topology with a split key whose digits refer to the taxa by their position in the `taxa=` argument, which defaults to the order in the file. Thus `"12|34"` separates the first two taxa from the last two; a Newick string (Felsenstein, 2004) is accepted in place of a key.

The central call is `px.evidence(ds, topology, model, register=...)`. Its `register` keyword, the package's word for the kind of answer, takes `"R1"` (equivalently `target="exact"`) for the exact value, `"R2"` for the proven interval, `"R3"` for the estimate and `"R0"` for the tie proof. The default, `"auto"`, chooses among these kinds by the rule described above, and a request that cannot be met raises `RegisterUnavailable` at once. With the keyword `budget=` (10^5 by default), users cap the number of likelihood evaluations an estimate uses: a pilot search for the modes takes about a fifth and four replicates share the rest. The keyword `seed=` (0 by default) fixes the random stream, so an estimate is a deterministic function of the alignment, the seed and the budget. A separate call, `px.capabilities()`, lists which of the optional libraries (see Availability) are installed.

Every call that computes a marginal likelihood returns a report object, written `rep` here, whose value is a Python `Fraction` when the answer is exact and a float $\ln Z$ otherwise. For the exact and interval kinds, the field `rep.interval` is an outward-rounded floating-point interval containing the answer. The report's error field is 0 for an exact value, the width for a proven interval, the error figure for an estimate that has one, and `None` otherwise. The last field, the certificate, is a JSON object that states the kind, model, prior, topology, taxa and version, together with a hash of the alignment as read and its column counts. For an exact value it also gives the numerator and the denominator in full, an integer that the denominator must divide, and a source hash of the exact-arithmetic module (the `engine` field).

We provide an independent routine, `px.verify_certificate(cert, dataset=ds)`, that tests a certificate against the alignment and the installed package. The routine rereads the alignment,

PRELIMINARY

compares its hash and taxon order with the stated values, checks the source hash of the exact-arithmetic module against the installed package, and recomputes the result fully or in part according to the kind of answer. An estimate is rerun from its seed and budget and must agree to within a rounding bound derived from the alignment size. An exact value is recomputed in full up to the routine's limit of 40 sites, beyond which the value is instead tested against its denominator bound and the multinomial ceiling defined under Models below (Goldman, 1993). Because the 68-site tRNA-Gln alignment exceeds that limit, a run with the default options (0.06 s) reports the full recomputation as skipped. The routine performs it on request (`recompute=True`) and, under `strict=True`, treats any skipped test as a failure. We designed the routine to catch honest error, such as a wrong alignment file or a damaged installation, and it gives no protection against a forger able to rewrite both the certificate and the package.

The default prior of Eq. (1) is uniform in $x = e^{-4t/3}$ for each branch length t. When a JC69 or K2P estimate is requested, `px.evidence` also accepts the experimental option `prior=px.BetaPrior(a, b)`, which places a Beta density with shape parameters from 1 to 10^6 on each of the estimator's unit-cube coordinates (for JC69 the x_e of Eq. 1); the resulting estimate carries no error figure. For example, $a = 3\lambda/4$ with $b = 1$ gives an exponential prior of rate $\lambda \geq 4/3$ on each JC69 branch length.

PRELIMINARY

Table 1: Entry points of phyloexact 0.2.3 (`import phyloexact as px`). "Kinds" are the kinds of answer a call can return; a tie is sought first wherever two topologies are compared. Module `pail` holds the tree-free calls and module `hill` the window-scan tools. †Experimental in 0.2.3: available and tested, but the interface may change in release 0.2.4, and verification of certificates across package versions is promised only for the unmarked (core) entries; known issues are listed in `ISSUES.md`.

Call	Input	Returns	Kinds
`px.Dataset.from_fasta(path, taxa=)`, `.from_rows`, `.from_counts`	FASTA file; taxon→sequence mapping; site-pattern counts	a `Dataset`; `ds.stats(topology)` gives N and m	–
`px.evidence(ds, topology, model, register=)`	`Dataset`; split key, Newick or `"all"`; model name	report: value, $\ln Z$, interval, error, kind, certificate; with `"all"`, the three marginal likelihoods, their sum, the posterior and the winner	all four
`px.decidability(ds)`	`Dataset`, 4–32 taxa	symmetry group of the pattern counts, the topologies it exchanges, and the ties this implies	tie
`px.compare(ds, A, B, model=)`	two topologies (split keys or Newick, ≤ 32 taxa)	log Bayes factor A over B, the exact ratio when both terms are exact; the tie proof instead when a symmetry implies a tie	all four
`px.compare_models(ds, topology, models=(A, B))`	one topology, two models	log Bayes factor A over B under proper priors, both terms of one kind	exact, interval, estimate
`px.grade_model(ds, rep)`; `px.check_adequacy(ds, topology, model)`	`Dataset`; an evidence report, or a topology and model	ceiling gap in log units per site against the multinomial ceiling (see Models); exact posterior-predictive table over pattern classes with its discrepancy	that of `rep`; exact, interval (JC69), estimate†
`px.pail_evidence(ds, model)`†; `px.pail_row(ds, models=)`†	`Dataset`; one model, or a model list	the marginal likelihood averaged over the three topologies; per model with the differences; no tree assumed (see Tree-free model choice across loci)	exact, estimate (one kind across models unless `allow_mixed_registers=True`)
`python -m phyloexact.hill.scan --rows W.json --out DIR`†	genome windows as row sets or as per-window pattern counts	one JSON line per window: posterior over topologies, tie status, ceiling gap, m; a manifest; weight files for wASTRAL (Zhang and Mirarab, 2022) and wQFM (Mahbub et al., 2021) via `hill.weights`	estimate (posterior, ceiling gap), tie, exact (ceiling)
`px.hill.sup_ceiling_from_counts(counts)`	pattern counts of any alignment, any alphabet	the multinomial ceiling as an exact integer ratio	exact
`px.PlugIn(loglik_fn, d, name=)`†	a vectorized log-likelihood of the alignment on $[0,1]^d$	a model object accepted by the calls above (see Adding a model)	estimate (no error figure); tie
`px.verify_certificate(cert, dataset=, recompute=, strict=)`	a certificate (dict or JSON file); the alignment as a `Dataset`, or as a path or mapping†	`ok`, the list of tests performed, the named omissions	all

PRELIMINARY

(a)

```
>>> import phyloexact as px
>>> ds = px.Dataset.from_fasta("pins/phylo_gkz/realdata/trnaq_quartet.fasta")    # mouse, rat, chicken, Xenopus mt tRNA-Gln, included with the package; N=68, m=25 for "12|34"
>>> rep = px.evidence(ds, topology="12|34", model="JC69", register="R1")           # 356 s, one process, host under load; "auto" returns the same exact value here (m=25 <= 25 at N=68)
>>> rep.register, rep.logZ, rep.error_nats
('R1', -261.40306415034735, 0.0)
>>> rep.value
Fraction(64822528584113983746...5736865245420889, 21758766967780217805...7019648000000000)    # 188-digit numerator, 302-digit denominator
>>> est = px.evidence(ds, "12|34", "JC69", register="R3", budget=100000, seed=0)    # 6.2 s
>>> est.logZ, est.error_nats, est.certificate["replicates"]["spread_nats"]         # error_nats is None: N=68 lies outside the classes with a measured error figure (N <= 32)
(-261.40315247963173, None, 0.00028203467712728525)
>>> alltop = px.evidence(ds, topology="all", model="JC69", register="R3", budget=100000, seed=0)   # 13 s
>>> alltop.winner, {t: float(p) for t, p in alltop.posterior.items()}
('12|34', {'12|34': 0.9999993409513044, '13|24': 4.843602971272815e-07, '14|23': 1.746883985758681e-07})
>>> hky = px.evidence(ds, "12|34", "HKY85", register="R2", minutes=3, procs=1)       # 184 s; kappa = 413/100 and base frequencies held at the rationalized ML fit
>>> hky.register, hky.interval
('R2-one-sided', (-inf, -230.79193341269783))
>>> v = px.verify_certificate(rep.certificate, dataset=ds)                          # 0.06 s
>>> v["ok"], v["n_performed"], v["n_skipped"]                                        # the one named omission: 'full exact recomputation ... N=68 ... RECOMPUTE_N_MAX = 40'; recompute=True performs it
(True, 18, 1)
```

(b)

```
{"register": "R1",  "model": "JC69",  "topology": "12|34",  "suite_version": "phyloexact-0.2.3",
 "prior": "per-edge uniform x_e ~ U(0,1) (JC x-parameterization; equals Exp(4/3) branch-length prior)",
 "taxa": ["V00711.1/3842-3772", "X14848.1/3820-3750", "X52392.1/5172-5102", "M10217.1/5909-5841"],
 "dataset": {"kind": "sequences", "N_used": 68, "dropped_columns": 0, "alphabet": "nucleotide"},
 "dataset_sha256": "825982082dce20d8...53438d1f",
 "value": {"num": "64822528584113983746...5736865245420889",      (188 digits)
           "den": "21758766967780217805...7019648000000000"},     (302 digits)
 "denominator_bound": {"bound": "den(Z) | 256^68 * lcm(1..69)^5 ...", "divides": true},
 "engine": {"entry": "jc_evidence", "sha256": "18d044c929168a54...78427d2b", "stats": {"N": 68, "m": 25,
     ...}, ...},
 "wall_seconds": 355.877,  ...}
```

Figure 2: A phyloexact 0.2.3 session on the mitochondrial tRNA-Gln quartet (68 sites). (a) Exact JC69 marginal likelihood of the mouse–rat topology, an estimate of it, the three-topology posterior, an HKY85 interval with only its upper endpoint proven, and the default `px.verify_certificate` run; `...` is an elision. (b) The certificate from line 3, abridged (more in Supplement S4).

PRELIMINARY

4 Models

Table 2 lists the built-in models and the kinds of answer available for each on a four-taxon alignment of N usable columns, with size limits stated partly in terms of the topology-dependent column count m defined above. Exact values require the site-pattern probabilities to be polynomials in transformed branch lengths such as the x_e (Schwartz et al., 2026). This condition holds under JC69 (Jukes and Cantor, 1969), including its application to protein recoded to four states (SR4; Susko and Roger, 2007) and its extension by integer site-rate classes (JC+R), and under Kimura's K2P and K3ST (Kimura, 1980, 1981). In JC+R each site evolves at an integer multiple $1, \ldots, r$ of a common rate, with Dirichlet-distributed class weights (Supplement S3), and the model is unrelated to the +R rate-heterogeneity model of ModelFinder (Kalyaanamoorthy et al., 2017). Under HKY85 (Hasegawa et al., 1985) the transition probabilities instead have the form $A + Bx_e^a + Cx_e^b$ with non-integer exponents set by the transition-transversion ratio κ and the base frequencies, and under GTR+Γ_4 (Tavaré, 1986; Yang, 1994) even the eigenvalues of the rate matrix are irrational. Hence the integer computation applies to neither model (although at rational κ and base frequencies the HKY85 marginal likelihood is still a rational number). For JC69 with integer site-rate classes and for K3ST, which do satisfy the polynomial condition, exact values are so far limited to toy alignment sizes.

Under K2P we give each branch its own transition and transversion parameters $\alpha_e t_e$ and $\beta_e t_e$ (Kimura, 1980), and $\kappa_e = \alpha_e/\beta_e$ may then differ between branches, whereas the usual form of the model has a single κ for the whole tree. The prior is uniform on the transformed branch lengths $e^{-4\beta_e t_e}$ and $e^{-2(\alpha_e+\beta_e)t_e}$ over their attainable region (area 2/3 per branch). Equivalently, the density of $(\alpha_e t_e, \beta_e t_e)$ is 2/3 times a product of exponential densities with means 1/2 and 1/6. The ratio κ_e then has prior median 3 on each branch. The exact integer form requires these branch-specific parameters, because the prior must be uniform on each branch's own transformed variables. This prior is left unnormalized, with total mass $(2/3)^5$, and phyloexact divides that mass out of every Bayes factor between K2P and another model. Bayes factors between K2P and JC69 still depend on the diffuse priors of the two models, under which the five extra K2P parameters carry an Occam penalty that can be decisive for alignments as short as a single tRNA gene (Kass and Raftery, 1995). For the usual single-κ model (K80), `model="HKY85-free-kappa"` (below) fixes equal base frequencies, integrates the single κ under a log-normal prior, and returns a proven interval or an estimate.

For the HKY85 and GTR+Γ_4 proven intervals we round the fitted or user-supplied substitution parameters to rationals and bound the branch-length integral at those fixed values in ball arithmetic (Johansson, 2017). The option `model="HKY85-free-kappa"` instead integrates κ under a log-normal prior at fixed base frequencies. Only this form is directly comparable with a JC69 or K2P marginal likelihood, because Eq. (1) defines the marginal likelihood as an integral over both the branch lengths and the substitution parameters, whereas an interval computed at fitted parameters is conditional on them. Integrating over the base frequencies as well would multiply the computing time by about 10^5. For the 63-parameter general Markov model (Barry and Hartigan, 1987) the term-by-term sum for the exact marginal likelihood has at least 16^K terms for K distinct site patterns and is therefore out of reach at locus scale. A Monte Carlo estimator for this model is implemented but not offered in this release, because its run-to-run variation between installations has not yet been bounded.

In addition to the marginal likelihood, phyloexact computes two measures of model fit: a ceiling gap and a posterior-predictive adequacy test. The ceiling gap, computed by `px.grade_model`

PRELIMINARY

for any marginal likelihood (plug-ins included), is

$$g_M \;=\; \frac{1}{N}\left(\sum_k n_k \ln \frac{n_k}{N} \;-\; \ln Z_M\right). \tag{2}$$

Here, n_k columns show pattern k and Z_M is the marginal likelihood of Eq. (1) under M. The sum, which we call the multinomial ceiling, is the unconstrained multinomial log-likelihood (Goldman, 1993), the best fit any independent-sites model can reach, and g_M is therefore nonnegative for every model and prior. Because the ceiling is evaluated exactly at any alignment size, g_M is of the same kind as $\ln Z_M$ (exact value, proven interval or estimate). Zero is not attainable, however: the ceiling fits every pattern frequency freely, so even an alignment simulated under M itself leaves a positive gap at finite N, from the sampling scatter of the counts and the spread of the prior (cf. Goldman, 1993), and g_M approaches zero only when a tree model reproduces the observed pattern frequencies exactly, in practice for alignments that are constant or saturated and therefore uninformative about the tree. The gap is therefore read comparatively. Two models' gaps on one alignment differ by their log Bayes factor per site; gaps from alignments of different lengths are not directly comparable; and `px.grade_model` reports beside g_M the same difference taken against the marginal likelihood of the unconstrained multinomial model under a uniform Dirichlet prior on all 256 pattern probabilities of a quartet, which is a log Bayes factor per site between M and the hypothesis of no structure common to the sites and can take either sign. For the tRNA-Gln quartet under JC69, $g_M = 0.91$ log units per site, while the multinomial model's marginal likelihood lies 0.99 log units per site below $\ln Z_M$: JC69 misses the ceiling, yet is preferred to the structureless alternative by 67.5 log units over the 68 sites. Whether the misfit that remains matters for a given question is what the adequacy test addresses.

The adequacy test (`px.check_adequacy`) is an exact form of the posterior-predictive test of Bollback (2002): for each class of site patterns that the model treats alike, it computes, as a ratio of two marginal likelihoods, the probability that one further column would show a pattern of that class. It then reports the Kullback–Leibler divergence D of observed from predicted class frequencies, in log units per site. When the JC69 or K2P marginal likelihood is itself an exact value, those probabilities and e^{ND} are exact rationals as well. The test is also available in two cheaper forms. These are a proven interval for JC69 up to 149 columns (a limit set by cost) and an estimate (experimental in this release, Table 1) for JC69, HKY85 and GTR+Γ_4 at any size. No form of the test is offered for JC+R, K3ST, the general Markov model or plug-ins, or for K2P beyond the size limits of its exact value. Supplement S1.7 lists every interface marked experimental in Table 1; it states what the label means and where known issues are listed.

PRELIMINARY

Table 2: Models in phyloexact 0.2.3 and the kinds of answer available for each on a four-taxon alignment of N usable columns; m and the error figure are defined under Four kinds of answer; every model is on a fixed unrooted topology without a clock. Priors are those of Eq. (1), Supplement S3 and, for K2P, the text.

Model	Exact value (`R1`)	Proven interval (`R2`)	Estimate (`R3`)	Tie proof (`R0`)
JC69 (Jukes and Cantor, 1969), also on SR4-recoded protein (Susko and Roger, 2007)	yes, $N \leq 150$; by default while $m \leq 25$ ($N \leq 68$), $m \leq 20$ ($N \leq 100$), $m \leq 10$ ($N \leq 150$); larger m on a GPU	yes, $N \leq 150$; two-sided, width $\sim 10^{-35}$ log units	yes, any N, ≥ 4 taxa on a fixed topology; error figure for $N \leq 32$, $m \leq 9$ (0.014 log units at 10^5 evaluations), none beyond	yes
JC69 + r integer rate classes	toy sizes only, $N \leq 8$ at $r = 2$	yes, two-sided; $N \leq 75$, 28, 15, 9 at $r = 2$, 3, 4, 5	user plug-in only	yes
K2P (Kimura, 1980), branch-specific κ	short alignments only: $m \leq 11$ and $N - m \leq 24$ (hence $N \leq 35$); minutes to hours near the limit	not built	yes, any N; error figure for $N \leq 35$, $m \leq 11$ (below 0.02 log units at 5×10^5 evaluations), none beyond	yes
K3ST (K81) (Kimura, 1981)	toy sizes only, $N \leq 5$	not built (possible in principle)	user plug-in only	yes
HKY85 (Hasegawa et al., 1985)	no (transition probabilities not polynomial in x_e)	yes at fitted or user-supplied κ and base frequencies, not integrated over them; or κ integrated under its prior with the frequencies fixed; not with both free ($18^4 \approx 10^5$ boxes)	yes, any N; no error figure	yes
GTR+Γ_4 (Tavaré, 1986; Yang, 1994)	no (not polynomial)	yes at fitted or user-supplied exchangeabilities, base frequencies and gamma shape (four rate categories), not integrated over them; not with exchangeabilities, frequencies and shape all free	yes, any N; no error figure	yes
General Markov (Barry and Hartigan, 1987), 63 parameters	no at locus scale ($\geq 16^K$ terms for K distinct patterns)	no ($\geq 2^{63}$ boxes before the interval tightens)	withheld in 0.2.3	no
User plug-in, `px.PlugIn` (see Adding a model)	no	no	yes, with the spread of four replicates; no error figure	yes, if leaves are exchangeable under the model

PRELIMINARY

5 Accuracy and speed against existing estimators

Measured against exact values on nine small problems, the phyloexact 0.2.3 estimate has a mean absolute error of 0.0001 log units at 2.5×10^6 likelihood evaluations (Table 3 and Fig. 3). The most accurate of the nineteen estimators benchmarked by Fourment et al. (2020), generalized stepping-stone sampling in physher (Fan et al., 2011; Fourment and Holmes, 2014), has an error of 0.0026 log units at 5.1×10^7 evaluations, which is 26 times our error at twenty times our budget.

Six of the nine problems are the three unrooted topologies of each of two quartet alignments, the mouse-to-*Xenopus* mitochondrial tRNA-Gln (68 gapless sites) and the hominid tRNA-Phe (69 sites). The other three are tRNA-Phe with gibbon added (67 sites) on three of the fifteen five-taxon topologies. Because two pairs of these problems tie exactly, the nine amount to seven distinct computations (Supplement S3.7). Each problem is the JC69 (Jukes and Cantor, 1969) marginal likelihood over five or seven branch lengths with independent exponential priors of rate 4/3, and every program was given this model and prior. The nine reference values are exact and distributed with the package. The six quartet values, computed by the package's exact routine and checked by its test suite, are given as ratios of two integers. The three quintet values, computed independently in the companion paper (Schwartz et al., 2026), are given as 60-digit logarithms. The package itself computes no exact five-taxon value and offers only the JC69 estimate for five or more taxa (see Costs and limits). We ran the nineteen methods of Fourment et al. (2020) in physher at that study's defaults. BEAST 2 (Bouckaert et al., 2019), whose path-sampling and nested-sampling packages implement algorithms that physher rows represent here, was not run separately. Stepping-stone sampling in MrBayes 3.2 and RevBayes (Xie et al., 2011; Ronquist et al., 2012; Höhna et al., 2016), and LoRaD (Wang et al., 2023), run through its R package lorad (Milkey et al., 2023)„ were run on the tRNA-Gln and quintet topologies at two budgets, one matched to ours by proposal count and one eight times larger. Supplement Tables S14 and S15 list the version and settings of every program. All timed runs used one thread of one Linux server (Intel Xeon Platinum 8581C, 2.3 GHz, 96 logical cores, 732 GB of memory), referred to throughout as the benchmark server. The comparison programs were timed separately, with the server under partial load, as medians of four runs (Supplement S3). The phyloexact 0.2.3 times are means over four seeds with the server under full load, which lengthens them (see Example analyses).

MrBayes stepping-stone sampling deviated from the exact values by about 0.024 log units after 7.2 s per run on the three tRNA-Gln topologies, and RevBayes by about 0.017 log units after 13.0 s. The phyloexact estimate on the same topologies had a mean error of 0.0003 log units and took 31 s per run without the optional compiled likelihood kernel. LoRaD was more accurate than either stepping-stone program, with an error of about 0.005 log units on the tRNA-Gln and tRNA-Phe topologies at 6.8 s per run. At that budget of 2.5×10^6 evaluations MrBayes, RevBayes and LoRaD are, per run, two to ten times faster than uncompiled single-thread phyloexact and one to two orders of magnitude less accurate. The two like-for-like comparisons favor phyloexact more strongly. At its default budget of 10^5 evaluations the estimate takes 3.8 s per quartet run and 4.4 s per quintet run uncompiled (1.1 and 1.3 s with the compiled kernel), less than one run of any of the three programs, with a mean error of 0.0006 log units over the nine problems (maximum 0.0024; Table 3). At equal run time, about 7 s, its mean error is 0.0003 uncompiled and 0.0001 compiled, 20 to 44 times below LoRaD's and 67 to 213 times below the two stepping-stone programs'. At equal accuracy the gap is larger still: by the $B^{-1/2}$ scaling of the sampling error, confirmed between the two budget levels of Supplement Table S16 for RevBayes, LoRaD would need about 500 s and RevBayes about 1.2×10^4 s per run to reach the default-budget error of phyloexact, and MrBayes does not reach it at any budget because of the offset described below.

PRELIMINARY

The physher methods other than generalized stepping-stone sampling had errors of 0.018 to 16 log units at their default budgets. Because the reference values are exact, a systematic offset in an estimator, such as the discretization bias of stepping-stone sampling (Xie et al., 2011), can be distinguished from its run-to-run scatter. With the budget raised eightfold, the MrBayes error on the quintet topologies kept its sign and size while its scatter shrank (Supplement Table S16; on one topology the two levels share a seed, note a). That left MrBayes with an offset of 3.2 to 3.8 standard deviations on every topology at the larger budget, whereas the RevBayes errors there differed in sign between topologies. This offset is unchanged, within its standard error, by further stones, longer chains, doubled burn-ins, the direction of the path and the precision of the likelihood. It disappears when one of the three default branch-length proposals, the node slider, is switched off: over eight of the exact problems the mean deviation is -0.026 log units (standard error 0.003) with the default proposals and -0.003 (0.002) without the node slider, from 72 estimates per setting (Supplement S3.9). The proposal ratio coded for this move from MrBayes 3.2.4 onward contains one more factor of the ratio of new to old length sum than the value that leaves the branch-length prior invariant, which is the value versions 3.1.2 to 3.2.3 used. Run without data and with the node slider as its only proposal, MrBayes 3.2.7a accordingly samples trees 1.4 to 3 times longer than the prior mean, depending on the prior (Supplement S3.9). In the default mixture of proposals the posterior tree length changes by about 0.1 per cent on our problems, too little to affect inferred trees. We compared MrBayes and RevBayes with phyloexact only through their stepping-stone estimators on small fixed topologies, although both are general programs built for large trees.

For one Bayes factor the stepping-stone and LoRaD errors above are negligible compared with $\ln \mathrm{BF} = 3$, the conventional threshold for strong evidence (Kass and Raftery, 1995). However, thousandths of a log unit can matter where two marginal likelihoods nearly tie, as they may for short loci and genome windows, and where errors of one sign accumulate over many windows. Where log marginal likelihoods are summed over many loci, as in the tree-free model choice example below, a per-locus offset that differs between the models compared accumulates in the total.

Under K2P (Kimura, 1980) with branch-specific κ we have run no full multi-program comparison. There the phyloexact estimate deviated from exact values on alignments of up to 35 columns by at most 0.0053 log units over 310 runs at 5×10^5 evaluations. The K2P error figure of Table 2 was derived from these measurements as three times their largest deviation. Supplement S3 gives the reference values, all settings, the K2P measurements, the results of the earlier release 0.1.0 (Table 3, note a) and the full twenty-one-row table from which that table is drawn.

PRELIMINARY

Table 3: Accuracy and speed on the nine tRNA problems (JC69, fixed topology). Error: mean absolute deviation from the exact log marginal likelihood, all nine problems unless marked. Wall-clock (tRNA-Gln quartet, tRNA-Phe quintet): phyloexact, mean over four seeds, benchmark server under full load (note a); other programs, median of four one-thread runs, same server under partial load. Parenthesized phyloexact times are with the optional compiled likelihood kernel. The physher methods (defaults of Fourment et al., 2020) were not timed.

method	program	evaluations	mean |error| (log units)	wall-clock (s) quartet	quintet
Importance sampling, randomized quasi-Monte Carlo	phyloexact 0.2.3[a] (this paper), default budget	1.0×10^5	0.0006	3.8 (1.1)	4.4 (1.3)
same, benchmark budget	phyloexact 0.2.3[a]	2.5×10^6	0.0001	31 (23)	39 (28)
Generalized stepping-stone (Fan et al., 2011)	physher	5.1×10^7	0.0026	–	–
LoRaD (Wang et al., 2023)	LoRaD (R package)[c]	2.0×10^6	∼0.005	6.8	6.2
Stepping-stone (Xie et al., 2011)	RevBayes (Höhna et al., 2016)[b]	2.5×10^6	∼0.017	13.0	15.2
Bridge sampling	physher	1.0×10^6	0.018	–	–
Stepping-stone (Xie et al., 2011)	MrBayes 3.2 (Ronquist et al., 2012)[b]	2.6×10^6	∼0.024	7.2	3.9
Stepping-stone and path sampling (Xie et al., 2011; Lartillot and Philippe, 2006)	physher	5.0×10^7	0.027–0.030	–	–
Nested sampling (Skilling, 2006; Maturana Russel et al., 2019)	physher	1.2×10^6	0.29	–	–
Naive Monte Carlo	physher	1.0×10^4	4.2	–	–
Harmonic mean (Newton and Raftery, 1994)	physher	1.0×10^6	∼7.5	–	–

[a] Without the optional compiled kernel, server under full load; on the tRNA-Gln topologies alone 0.0003 log units. Release 0.1.0 used a single fitted proposal, which 0.2.3 replaces by a mixture over every posterior mode and its images under the symmetries of the tree to remove an underestimate by $\ln 2$ on near-symmetric quartets (Supplement S3). The earlier release had a mean error of 0.00006 log units and, with the compiled kernel on sixteen threads, ran in 4.9 s on the quartet and 7.0 s on the quintet. [b] tRNA-Gln topologies only, twelve runs; MrBayes reports two estimates per invocation, RevBayes one (settings in Supplement S3). [c] tRNA-Gln and tRNA-Phe topologies; two estimates per invocation.

PRELIMINARY

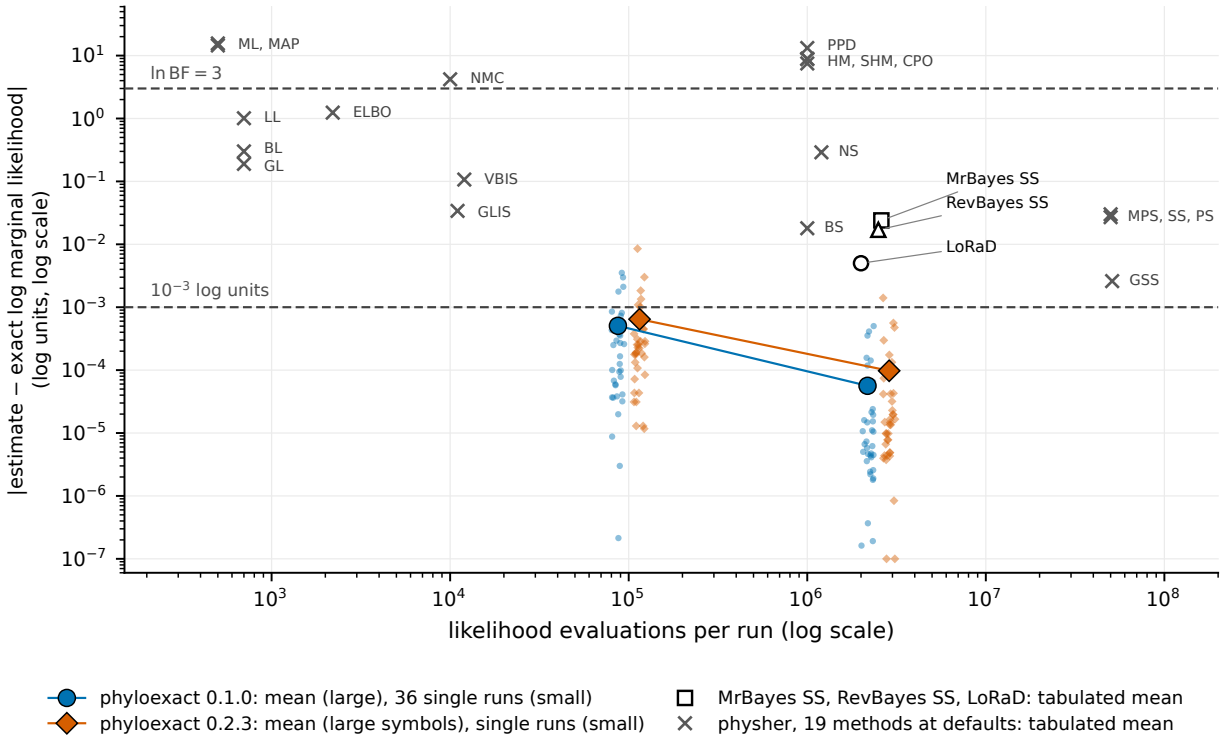

Figure 3: Absolute error against the exact log marginal likelihood versus likelihood evaluations per run, nine tRNA problems; at the larger budget phyloexact is one to two orders of magnitude more accurate than MrBayes, RevBayes and LoRaD. Filled, phyloexact 0.1.0 (circles) and 0.2.3 (diamonds; Table 3 note a), means and single runs, offset sideways; open, MrBayes, RevBayes, LoRaD; crosses, physher methods abbreviated as in Fourment et al. (2020).

6 Example analyses

We illustrate phyloexact with four analyses and give, for each, the commands a user types together with the results. Each call ran as one process on the benchmark server under full load, which lengthens the run times: the slowest call, 356 s here, takes 295 s on an idle machine. The alignments are distributed with the package and a longer session is reproduced in Supplement S1. The tree-free (module `pail`), window-scan (module `hill`) and plug-in entry points used below are experimental (Table 1).

6.1 A single tRNA quartet

The first example uses the mitochondrial tRNA-Gln quartet introduced above (68 sites, none dropped), and Fig. 2 reproduces the Python session in which several of this example's calls were run. For 12|34, which pairs mouse with rat, phyloexact returns the exact JC69 (Jukes and Cantor, 1969) marginal likelihood, $\ln Z = -261.40306$, in 356 s. The run time is governed by the topology-dependent column count m, which is 25 here but 32 and 33 for the alternatives 13|24 and 14|23, for which the exact computation takes half an hour or more (see Costs and limits). Hence the default call computes 12|34 exactly and estimates the other two. The call with `topology="all"` and `register="R3"` computes estimates of all three marginal likelihoods at 10^5 evaluations in 13 s, and, under a uniform prior on the three topologies, these estimates give 12|34 a posterior probability of $1 - 6.6 \times 10^{-7}$ and a log Bayes factor of 14.5 log units over 13|24. The 12|34 estimate lies 8.8×10^{-5} log units from the exact value, and because the alignment is larger than any size class with a measured error, the only uncertainty reported for it is the replicate spread.

An exact K2P value is out of reach at 68 sites, and a request for one raises the error `RegisterUnavailable` with a message giving the size limit and the requests that would succeed. The call `px.compare_models(ds, "12|34", models=("K2P","JC69"))` instead computes both marginal likelihoods as estimates in 12 s, and the resulting log Bayes factor favors K2P by 3.79 log units (K2P with a separate κ on every branch; see Models). For HKY85 (Hasegawa et al., 1985) the call with `register="R2"` and `minutes=3` runs for 184 s and returns a proven upper bound at the fitted parameters rounded to rationals ($\kappa = 413/100$), $\ln Z \leq -230.7919$. This bound lies above both estimates and is therefore uninformative about how HKY85 compares with either model on this alignment. On the first 16 columns of the tRNA-Gln quartet (the object `d16`, built with `px.Dataset.from_rows`; $m = 9$) the comparison of K2P with JC69 can be exact. There `px.compare_models(d16, "12|34", models=("K2P","JC69"), register="R1")` returns $\ln \mathrm{BF} = 0.6921$ (K2P over JC69), the logarithm of a ratio of two 80-digit integers, in 139 s, and `px.verify_certificate(..., strict=True)` recomputes both marginal likelihoods in 127 s.

6.2 Tree-free model choice across loci

ModelTest and its successors (Posada and Crandall, 1998; Darriba et al., 2012, 2020) and ModelFinder (Kalyaanamoorthy et al., 2017) score substitution models on a guide tree. For four taxa the tree can be left out: each model's marginal likelihood is averaged over the three topologies and the logarithms are summed over loci. Our example data set consists of eighteen mitochondrial tRNA loci of human, chimpanzee, gorilla and orangutan (64–75 usable sites, 3–15 variable; package directory `pins/pail_demo`; the panel was fixed before any marginal likelihood was computed, Supplement S3.9). For each locus `px.pail_evidence(ds, "JC69")` returns the exact topology-averaged marginal likelihood, and `px.pail_evidence(ds, "K2P",`

PRELIMINARY

locus	sites	variable	$\ln \bar{Z}$, JC69 (exact)	K2P−JC69 (log units)	preferred	topology under JC69
Ala	69	4	−134.937	−9.38	JC69	HC|GO*
Arg	64	6	−139.638	−8.67	JC69	HC|GO
Asn	73	10	−173.790	−1.50	JC69	HC|GO
Asp	67	13	−174.477	+0.54	K2P	HO|CG
Cys	66	8	−147.512	−7.82	JC69	HG|CO
Gln	72	7	−155.798	−5.96	JC69	HC|GO
Glu	69	15	−178.617	+0.77	K2P	HC|GO*
Gly	68	9	−160.495	−4.77	JC69	HC|GO
His	69	9	−156.958	−6.23	JC69	HC|GO
Ile	68	8	−159.193	−5.90	JC69	HG|CO
Leu(CUN)	71	3	−135.479	−12.06	JC69	HC|GO
Leu(UUR)	75	7	−160.408	−6.11	JC69	HG|CO
Met	68	3	−129.231	−10.42	JC69	HG|CO*
Pro	68	11	−175.336	−6.81	JC69	HG|CO
Ser(UCN)	69	6	−146.554	−6.92	JC69	HG|CO*
Trp	66	9	−159.199	−2.10	JC69	HC|GO = HG|CO
Tyr	65	12	−162.697	−1.32	JC69	HC|GO
Val	69	11	−165.147	−5.46	JC69	HG|CO
all 18 loci			−2815.467	-100.13 ± 0.0024	JC69 (16 of 18)	10 / 7 / 1

Table 4: Tree-free model choice on eighteen mitochondrial tRNA loci (human H, chimpanzee C, gorilla G, orangutan O). $\ln \bar{Z}$: log topology-averaged marginal likelihood, exact under JC69. K2P−JC69 (branch-specific κ): four-seed mean at 5×10^5 evaluations, seed s.d. < 0.003 log units. Last column: best JC69 topology (* other two tie; Trp's best two tie). Bottom: sums; loci per best topology HC|GO, HG|CO, HO|CG (Trp as HC|GO).

PRELIMINARY

`register="R3", budget=500000, seed=s)` an estimate in about a minute on one thread. With each K2P estimate averaged over four independent runs (Table 4), JC69 is preferred at sixteen loci by 1.3 to 12.1 log units, and K2P at tRNA-Asp and tRNA-Glu by under 1 log unit. Over all eighteen loci JC69 is favored by 100 log units, the difference between the exact JC69 total and the estimated K2P total (see Schwartz et al., 2026, for a larger data set and a split-half test).

6.3 A window scan

The command `python -m phyloexact.hill.scan --rows windows.json --out scan/` compares the three topologies of one quartet in every window of a genome alignment, reading four taxon names and, per window, an identifier and four aligned rows or site-pattern counts. For each window it writes the estimated JC69 posterior over topologies, exact ties, m, and the exact multinomial ceiling together with the ceiling gap of Eq. (2). We ran it on six hominid tRNA loci distributed with the package, treating each locus as a window, and the scan took 14 s with one process at the default 20,000 evaluations per window (Fig. 4). At tRNA-Phe the three posterior probabilities lie within 0.001 of 1/3, and HC|GO and HO|CG are returned as an exact tie. The command `python -m phyloexact.hill.weights` then converts the scan output into input files for wASTRAL and wQFM (Zhang and Mirarab, 2022; Mahbub et al., 2021).

Because a 2-kb hominid genome window (m roughly 120 to 190) is in practice too large for the exact computation, its posterior is an estimate without an error figure, while its ties and its multinomial ceiling remain exact. An earlier version of the tool scanned 243,339 window–quartet combinations of an *Anopheles* genome alignment (Fontaine et al., 2015) in 105 minutes on 48 cores. The routine `px.hill.sup_ceiling_from_counts`, which computes only the multinomial

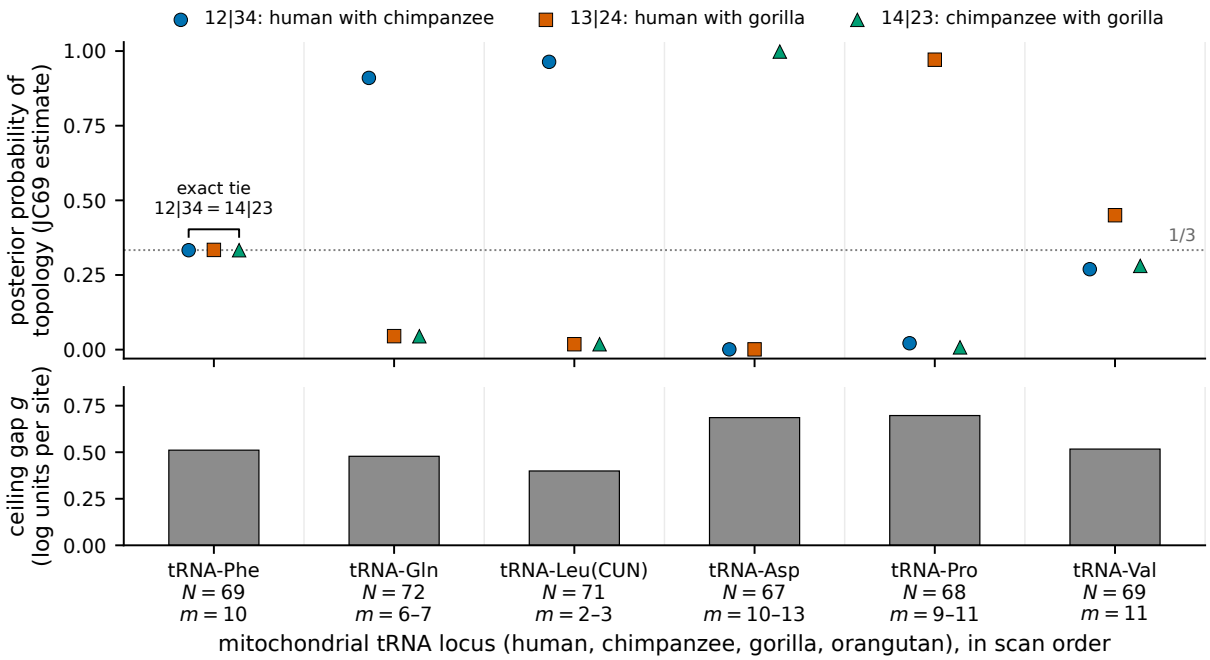

Figure 4: Window scan of six hominid mitochondrial tRNA loci (one process, 14 s). Top: estimated JC69 topology posteriors (2×10^4 evaluations per window; 12|34 is HC|GO as in Table 4); the bracket marks the exact tRNA-Phe tie, proved by a taxon relabeling. Bottom: each window's ceiling gap (Eq. 2, log units per site), reported to flag ill-fitting windows.

ceiling and accepts the site-pattern counts of an alignment with any number of taxa, takes seconds on a 2,000-taxon alignment (Supplement S4).

6.4 Adding a model

A user adds a model by wrapping its likelihood in `px.PlugIn(loglik_fn, d, name)`. The function maps an (n, d) array of points in the unit cube to n log-likelihoods of the alignment, and because phyloexact integrates this likelihood against the uniform measure on the cube, the user's map from cube coordinates to branch lengths and parameters defines the prior. We wrote such a function for JC69 with an invariant-sites fraction in about 25 lines of NumPy ($d = 6$). Applied to the hominid tRNA-Phe quartet, this plug-in gives the estimate $\ln Z = -158.286$ at 20,000 evaluations in 1.6 s with replicate spread 0.031 log units but without an error figure, which would require exact values under the user's model. The same function with the invariant fraction held at zero yields an estimate within 0.003 log units of the exact JC69 value. Ties between topologies can also be proved for a plug-in whose likelihood treats the leaves exchangeably, whereas an exact value would require a proof that the model's marginal likelihood is a rational number under its prior (Supplement S3). Three worked plug-ins are distributed in `examples/`, and one of them, `covarion_plugin.py`, computes the marginal likelihood of the tRNA-Gln quartet under a seven-parameter covarion model (Tuffley and Steel, 1998) in 7 s.

PRELIMIN

7 Costs and limits

Run time depends on the kind of answer, and for the exact JC69 value it grows steeply with both N and m (Fig. 5a). A synthetic worst-case quartet with $m = 5$ takes 0.6 s at $N = 32$ and 341 s at $N = 150$ on one thread of the benchmark server, and at $N = 68$ the time rises from 15 s at $m = 10$ to 15 min at $m = 30$. Because N is fixed for a given alignment, the exact run time depends on the topology only through m. The same thread, on an otherwise idle server, takes five minutes for the tRNA-Gln 12|34 topology ($m = 25$) and roughly half an hour for each alternative ($m = 32$ and 33). We based the 8 min threshold in the default rule for choosing the kind of answer on these timings. Where that rule selects the estimate, the result includes a note giving the measured run time of the nearest benchmarked (N, m) cell as a guide to the exact call's cost. However, `register="R1"` forces the exact JC69 computation at any m for $N \leq 150$ (timings in Supplement S3). The exact value can also be computed on a GPU. On a rented 80 GB NVIDIA H100, timing runs of an earlier release reached $m = 191$ (a projected 11 h per topology) and ran out of memory at $m = 215$.

The cost of the estimate depends little on N or m. For the benchmark problems and the worked tRNA-Gln example we timed single calls on one thread of the benchmark server under full load. They took roughly 3 to 6 s at the default 10^5 evaluations and about 26 to 40 s at 2.5×10^6. Every likelihood evaluation in a call, those of the pilot search included, is counted against the budget (see Interface), and each call also spends about 2 s on setup. Proven intervals cost the most: on tRNA-Gln a two-sided GTR+Γ_4 interval 0.20 log units wide took 9.5 h on 24 processes under release 0.2.0, while a short run proves only the upper endpoint. We have not profiled memory on the CPU; in the one-thread timing sweep of Supplement S3 every program, the release-0.1.0 estimator included, ran inside a 16 GB address-space limit.

Because each marginal likelihood is computed on a fixed topology, phyloexact neither searches tree space nor samples under the multispecies coalescent. Programs for tree search and for larger trees include MrBayes, RevBayes, BEAST 2 and PhyloBayes (Ronquist et al., 2012; Höhna et al., 2016; Bouckaert et al., 2019; Lartillot et al., 2013) for Bayesian inference, and IQ-TREE 2 and RAxML-NG (Minh et al., 2020; Kozlov et al., 2019) for maximum likelihood. For five or more taxa phyloexact offers only the JC69 estimate (with no error figure, because the size classes with a measured error cover only quartets) and tie proofs for up to 32 taxa. A request for an exact five-taxon value is refused. The separate call `px.quintet.adjudicate` ranks the fifteen trees on five taxa by JC69 maximum likelihood with proven bounds but computes no marginal likelihood. We have run the weight-file writers for wASTRAL (Zhang and Mirarab, 2022) and wQFM (Mahbub et al., 2021) on synthetic data only, apart from the window-scan example above. A call beyond any of these size or taxon limits, or one that needs a missing optional component (the `python-flint` library or a GPU), raises `RegisterUnavailable` before computing anything.

PREPRINT

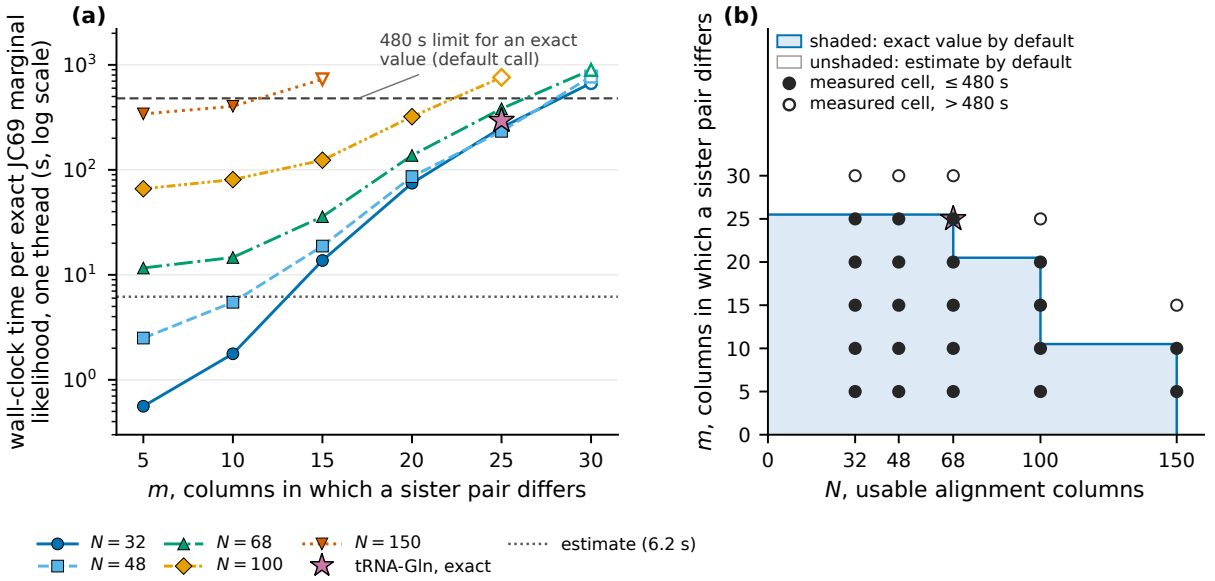

Figure 5: Run time of the exact JC69 quartet marginal likelihood, one thread. (a) Seconds (log scale) against m for $N = 32$ to 150 (synthetic); star, tRNA-Gln 12|34 (295 s); dashed, the default call's 480 s limit for an exact value; dotted, the estimate at 10^5 evaluations (6.2 s). (b) Shaded, the (N, m) region where the default call returns the exact value; filled symbols, measured cells inside it; open symbols, cells outside it (both panels).

8 Validation

The phyloexact package includes a suite of 38 tests run by one command,

```
python -m phyloexact.validate
```

which needs no GPU or external data and writes a JSON report. The tests compare each exact value with an independent evaluation of the same integral, for example by direct symbolic integration in SymPy (Meurer et al., 2017) on small alignments. In particular, the JC+R and K3ST routines share no arithmetic with the JC69 routine and, at the parameter values that reduce each to JC69, must return its numerator and denominator. The K2P routine, whose JC69 reduction is the package's exact JC69 routine, is itself tested against an independent JC69 implementation. Every proven interval must contain the exact value where one exists, and every estimate must lie within its error figure of that value. The suite also corrupts, in turn, each quantity in a certificate that the independent routine described under Interface recomputes, and requires that routine to reject the corrupted certificate and identify the altered quantity. The benchmark's exact reference values are included with the package, and the suite optionally recomputes the four-taxon values among them through the public interface. We run the suite before each release under Python 3.10 and 3.12 (with and without python-flint).

9 Availability

JaCK & Jill is distributed as the Python package `phyloexact`, version 0.2.3, under the MIT license (manual pages CC BY 4.0). Source code, a versioned archive and the issue tracker will

PRELIMINARY

be linked from `https://bootloops.ai` on release. Running phyloexact requires Python 3.10 or later with NumPy, SciPy, mpmath and SymPy (Harris et al., 2020; Virtanen et al., 2020; The mpmath development team, 2023; Meurer et al., 2017). Optional libraries extend phyloexact: python-flint (Arb; Johansson, 2017) enables proven intervals, Numba (Lam et al., 2015) compiles the likelihood kernel of the importance-sampling estimator, and PyTorch with CUDA (Paszke et al., 2019) runs large exact computations on a GPU. The package installs from its source tree (`pip install -e .`) or runs from that tree on `PYTHONPATH`. No prebuilt distribution (Python wheel) is provided, because the package reads reference files that sit beside the package directory and a wheel would not copy them. We develop and test on Linux; Windows is untested, and on macOS some interval routines need the `fork` start method (otherwise they raise an error whose message states the fix).

The Supplement is the user manual, compiled from the repository pages `GUIDE.md`, `MODELS.md` and `SPEC.md`; `GUIDE.md` marks the interfaces that may still change, `ISSUES.md` lists known issues, and a tool page is at `https://bootloops.ai/tools/jackandjill.html`. We ask that published analyses report the phyloexact version used; Schwartz et al. (2026) describe the underlying mathematics.

Summary

JaCK & Jill returns a four-taxon marginal likelihood as an exact value, a proven interval, an estimate with a measured error or a proof of a tie, each with a testable certificate. On nine tRNA problems with exact answers its estimate was one to two orders of magnitude more accurate than MrBayes, RevBayes and LoRaD, at two to ten times their single-thread run time. The exact values also serve as ground truth for tuning any sampling estimator, and extending the measured-error classes to alignments of tRNA length is the next use we intend for them.

Acknowledgments

Funding for this project was provided by Anthropic. This is not an official Anthropic project and results and views are not endorsed by Anthropic.

References

Baele, G., Lemey, P., Bedford, T., Rambaut, A., Suchard, M. A., and Alekseyenko, A. V. (2012). Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty. *Mol. Biol. Evol.*, 29:2157–2167.

Barry, D. and Hartigan, J. A. (1987). Statistical analysis of hominoid molecular evolution. *Stat. Sci.*, 2:191–207.

Bollback, J. P. (2002). Bayesian model adequacy and choice in phylogenetics. *Mol. Biol. Evol.*, 19:1171–1180.

Bouckaert, R., Vaughan, T. G., Barido-Sottani, J., Duchêne, S., Fourment, M., Gavryushkina, A., Heled, J., Jones, G., Kühnert, D., De Maio, N., Matschiner, M., Mendes, F. K., Müller, N. F., Ogilvie, H. A., du Plessis, L., Popinga, A., Rambaut, A., Rasmussen, D., Siveroni, I., Suchard, M. A., Wu, C.-H., Xie, D., Zhang, C., Stadler, T., and Drummond, A. J. (2019).

PRELIMINARY

BEAST 2.5: an advanced software platform for Bayesian evolutionary analysis. *PLoS Comput. Biol.*, 15:e1006650.

Darriba, D., Posada, D., Kozlov, A. M., Stamatakis, A., Morel, B., and Flouri, T. (2020). ModelTest-NG: a new and scalable tool for the selection of DNA and protein evolutionary models. *Mol. Biol. Evol.*, 37:291–294.

Darriba, D., Taboada, G. L., Doallo, R., and Posada, D. (2012). jModelTest 2: more models, new heuristics and parallel computing. *Nat. Methods*, 9:772.

Dick, J., Kuo, F. Y., and Sloan, I. H. (2013). High-dimensional integration: the quasi-Monte Carlo way. *Acta Numer.*, 22:133–288.

Fan, Y., Wu, R., Chen, M.-H., Kuo, L., and Lewis, P. O. (2011). Choosing among partition models in Bayesian phylogenetics. *Mol. Biol. Evol.*, 28:523–532.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol. Evol.*, 17:368–376.

Felsenstein, J. (2004). *Inferring Phylogenies*. Sinauer Associates, Sunderland, MA.

Fontaine, M. C., Pease, J. B., Steele, A., Waterhouse, R. M., Neafsey, D. E., Sharakhov, I. V., Jiang, X., Hall, A. B., Catteruccia, F., Kakani, E., Mitchell, S. N., Wu, Y.-C., Smith, H. A., Love, R. R., Lawniczak, M. K., Slotman, M. A., Emrich, S. J., Hahn, M. W., and Besansky, N. J. (2015). Extensive introgression in a malaria vector species complex revealed by phylogenomics. *Science*, 347:1258524.

Fourment, M. and Holmes, E. C. (2014). Novel non-parametric models to estimate evolutionary rates and divergence times from heterochronous sequence data. *BMC Evol. Biol.*, 14:163.

Fourment, M., Magee, A. F., Whidden, C., Bilge, A., Matsen, IV, F. A., and Minin, V. N. (2020). 19 dubious ways to compute the marginal likelihood of a phylogenetic tree topology. *Syst. Biol.*, 69:209–220.

Goldman, N. (1993). Statistical tests of models of DNA substitution. *J. Mol. Evol.*, 36:182–198.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585:357–362.

Hasegawa, M., Kishino, H., and Yano, T. (1985). Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.*, 22:160–174.

Hendy, M. D. and Penny, D. (1989). A framework for the quantitative study of evolutionary trees. *Syst. Zool.*, 38:297–309.

Hendy, M. D. and Penny, D. (1993). Spectral analysis of phylogenetic data. *J. Classif.*, 10:5–24.

Höhna, S., Landis, M. J., Heath, T. A., Boussau, B., Lartillot, N., Moore, B. R., Huelsenbeck, J. P., and Ronquist, F. (2016). RevBayes: Bayesian phylogenetic inference using graphical models and an interactive model-specification language. *Syst. Biol.*, 65:726–736.

PRELIMINARY

Johansson, F. (2017). Arb: efficient arbitrary-precision midpoint-radius interval arithmetic. *IEEE Trans. Comput.*, 66:1281–1292.

Jukes, T. H. and Cantor, C. R. (1969). Evolution of protein molecules. In Munro, H. N., editor, *Mammalian Protein Metabolism*, volume III, pages 21–132. Academic Press, New York.

Kalyaanamoorthy, S., Minh, B. Q., Wong, T. K. F., von Haeseler, A., and Jermiin, L. S. (2017). ModelFinder: fast model selection for accurate phylogenetic estimates. *Nat. Methods*, 14:587–589.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *J. Am. Stat. Assoc.*, 90:773–795.

Kimura, M. (1980). A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.*, 16:111–120.

Kimura, M. (1981). Estimation of evolutionary distances between homologous nucleotide sequences. *Proc. Natl. Acad. Sci. USA*, 78:454–458.

Kozlov, A. M., Darriba, D., Flouri, T., Morel, B., and Stamatakis, A. (2019). RAxML-NG: a fast, scalable and user-friendly tool for maximum likelihood phylogenetic inference. *Bioinformatics*, 35:4453–4455.

Lam, S. K., Pitrou, A., and Seibert, S. (2015). Numba: a LLVM-based Python JIT compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC (LLVM '15)*, pages 7:1–7:6, New York. ACM.

Lanfear, R., Frandsen, P. B., Wright, A. M., Senfeld, T., and Calcott, B. (2017). PartitionFinder 2: new methods for selecting partitioned models of evolution for molecular and morphological phylogenetic analyses. *Mol. Biol. Evol.*, 34:772–773.

Lartillot, N. and Philippe, H. (2006). Computing Bayes factors using thermodynamic integration. *Syst. Biol.*, 55:195–207.

Lartillot, N., Rodrigue, N., Stubbs, D., and Richer, J. (2013). PhyloBayes MPI: phylogenetic reconstruction with infinite mixtures of profiles in a parallel environment. *Syst. Biol.*, 62:611–615.

Mahbub, M., Wahab, Z., Reaz, R., Rahman, M. S., and Bayzid, M. S. (2021). wQFM: highly accurate genome-scale species tree estimation from weighted quartets. *Bioinformatics*, 37:3734–3743.

Martin, S. H. and Van Belleghem, S. M. (2017). Exploring evolutionary relationships across the genome using topology weighting. *Genetics*, 206:429–438.

Maturana Russel, P., Brewer, B. J., Klaere, S., and Bouckaert, R. R. (2019). Model selection and parameter inference in phylogenetics using nested sampling. *Syst. Biol.*, 68:219–233.

Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J. K., Singh, S., Rathnayake, T., Vig, S., Granger, B. E., Muller, R. P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., Curry, M. J., Terrel, A. R., Roučka, Š., Saboo, A., Fernando, I., Kulal, S., Cimrman, R., and Scopatz, A. (2017). SymPy: symbolic computing in Python. *PeerJ Comput. Sci.*, 3:e103.

PRELIMINARY

Milkey, A., Korte, E., and Lewis, P. O. (2023). *lorad: Lowest Radial Distance Method of Marginal Likelihood Estimation*. R package version 0.0.1.0, https://CRAN.R-project.org/package=lorad.

Minh, B. Q., Schmidt, H. A., Chernomor, O., Schrempf, D., Woodhams, M. D., von Haeseler, A., and Lanfear, R. (2020). IQ-TREE 2: new models and efficient methods for phylogenetic inference in the genomic era. *Mol. Biol. Evol.*, 37:1530–1534.

Newton, M. A. and Raftery, A. E. (1994). Approximate Bayesian inference with the weighted likelihood bootstrap. *J. R. Stat. Soc. Ser. B*, 56:3–26.

Oaks, J. R., Cobb, K. A., Minin, V. N., and Leaché, A. D. (2019). Marginal likelihoods in phylogenetics: a review of methods and applications. *Syst. Biol.*, 68:681–697.

Owen, A. B. (1997). Scrambled net variance for integrals of smooth functions. *Ann. Stat.*, 25:1541–1562.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). PyTorch: an imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates. arXiv:1912.01703.

Pearson, W. R. and Lipman, D. J. (1988). Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA*, 85:2444–2448.

Posada, D. and Crandall, K. A. (1998). MODELTEST: testing the model of DNA substitution. *Bioinformatics*, 14:817–818.

Ronquist, F., Teslenko, M., van der Mark, P., Ayres, D. L., Darling, A., Höhna, S., Larget, B., Liu, L., Suchard, M. A., and Huelsenbeck, J. P. (2012). MrBayes 3.2: efficient Bayesian phylogenetic inference and model choice across a large model space. *Syst. Biol.*, 61:539–542.

Schwartz, M. D., Edwards, S. V., and Lewis, P. O. (2026). Exact Bayesian evidence for phylogenetic trees. Manuscript in preparation.

Skilling, J. (2006). Nested sampling for general Bayesian computation. *Bayesian Anal.*, 1:833–859.

Sturmfels, B. and Sullivant, S. (2005). Toric ideals of phylogenetic invariants. *J. Comput. Biol.*, 12:204–228.

Susko, E. and Roger, A. J. (2007). On reduced amino acid alphabets for phylogenetic inference. *Mol. Biol. Evol.*, 24:2139–2150.

Székely, L. A., Steel, M. A., and Erdős, P. L. (1993). Fourier calculus on evolutionary trees. *Adv. Appl. Math.*, 14:200–216.

Tavaré, S. (1986). Some probabilistic and statistical problems in the analysis of DNA sequences. *Lectures on Mathematics in the Life Sciences*, 17:57–86.

The mpmath development team (2023). mpmath: a Python library for arbitrary-precision floating-point arithmetic (version 1.3.0). https://mpmath.org/.

PRELIMINARY

Tuffley, C. and Steel, M. (1998). Modeling the covarion hypothesis of nucleotide substitution. *Math. Biosci.*, 147:63–91.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and SciPy 1.0 Contributors (2020). SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nat. Methods*, 17:261–272.

Wang, Y.-B., Milkey, A., Li, A., Chen, M.-H., Kuo, L., and Lewis, P. O. (2023). LoRaD: marginal likelihood estimation with haste (but no waste). *Syst. Biol.*, 72:639–648.

Xie, W., Lewis, P. O., Fan, Y., Kuo, L., and Chen, M.-H. (2011). Improving marginal likelihood estimation for Bayesian phylogenetic model selection. *Syst. Biol.*, 60:150–160.

Yang, Z. (1994). Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *J. Mol. Evol.*, 39:306–314.

Zhang, C. and Mirarab, S. (2022). Weighting by gene tree uncertainty improves accuracy of quartet-based species trees. *Mol. Biol. Evol.*, 39:msac215.

Zhang, C., Rabiee, M., Sayyari, E., and Mirarab, S. (2018). ASTRAL-III: polynomial time species tree reconstruction from partially resolved gene trees. *BMC Bioinformatics*, 19(Suppl. 6):153.

PRELIMINARY