Bayesian model testing

- How do we know which models (sequence evolution models, clock models, coalescent models, …) we should use?
- Use a well-developed statistical theory (model selection and model averaging) that enables models to be compared according to objective criteria (Steel, Trends Genet., 2005)
- Such approaches penalize the addition of extra parameters, unless there is a sufficiently impressive improvement in fit between model and data
- The approaches discussed here enable the comparison of any two models, even if they are not nested

Bayesian model testing

- Goal: finding the most appropriate model for your data
- Over-fitting: too many parameters, the model is too complex
- Under-fitting: too few parameters, the model is too simple
- Don't compare all possible model combinations (evolutionary model, clock models, coalescent tree prior, …) to one another!
- Test/compare those models that relate to (or might have an impact on) the hypothesis you are interested in testing

Bayesian model testing

The aim of model selection is not to find the 'true model' but to find a model with sufficient parameters to capture the key features of the data.

'Better, more realistic models' should not mean 'more parameter-rich models'!

A Bayesian alternative to classical hypothesis testing: the Bayes factor (a summary of the evidence provided by the data in favor of one scientific theory, represented by a statistical model, as opposed to another; Kass & Raftery, 1995).

Model testing using Bayes factors

The evaluation of Bayes factors has become a standard approach to perform model selection in a Bayesian phylogenetic framework.

The Bayes factor is a ratio of two marginal likelihoods (i.e. two normalizing constants of the form $p(D \mid M)$, with D the observed data and M an evolutionary model under evaluation) obtained for the two models, M_0 and M_1, under comparison (Jeffreys, 1935):

- Bayes factor $$B_{01} = \frac{p(D|M_1)}{p(D|M_0)}$$

Model testing using Bayes factors

- posterior $$p(\theta|D,M) = \frac{p(D|\theta,M)\, p(\theta|M)}{p(D|M)}$$
- marginal likelihood $$p(D|M) = \int_\theta p(D|\theta,M)\, p(\theta|M)\, d\theta$$
- Bayes factor $$B_{01} = \frac{p(D|M_1)}{p(D|M_0)}$$

- normalizing constant
- quantity of interest for model selection
- very difficult to compute

Reminder: MHG MCMC Sampling

More formally, we want to explore the posterior distribution in an efficient manner

D = Data
θ = Model parameters (model notation M omitted here)

Posterior distribution | Prior distribution | "Likelihood"

$$p(\theta \mid D) = \frac{p(\theta)\, p(D \mid \theta)}{\int p(\theta)\, p(D \mid \theta)\, d\theta}$$

Usually fairly easy to calculate

Very ugly integral/sum
Extremely tedious to calculate even once

Normalizing constant

The algorithm starts from a random state (θ) and 'proposes' a new state (θ^*)

The new state is accepted with probability: $R = \min\left(1, \frac{p(\theta^*|D)}{p(\theta|D)} \times \frac{p(\theta|\theta^*)}{p(\theta^*|\theta)}\right)$

Reminder: MHG MCMC Sampling

The algorithm starts from a random state (θ) and 'proposes' a new state (θ^*)

The new state is accepted with probability:

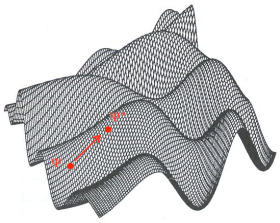
$$R = \min\left(1, \frac{p(\theta^*|D)}{p(\theta|D)} \times \frac{p(\theta|\theta^*)}{p(\theta^*|\theta)}\right)$$

$$= \min\left(1, \frac{p(D|\theta^*)\, p(\theta^*)\, p(D)}{p(D|\theta)\, p(\theta)\, p(D)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)}\right)$$

the two marginal likelihoods cancel out and don't have to be computed !

$$= \min\left(1, \frac{f(D|\theta^*)}{f(D|\theta)} \times \frac{f(\theta^*)}{f(\theta)} \times \frac{f(\theta|\theta^*)}{f(\theta^*|\theta)}\right)$$

Likelihood ratio | Prior ratio | Proposal ratio



Model testing using Bayes factors

- for model fit, the marginal likelihood p(D|M) is of primary importance (to calculate the Bayes factor)
- among several models, one is led to **choose the model with the highest marginal likelihood**
- when calculated correctly/accurately: takes into account differences in dimensions, so higher dimensional models are not automatically preferred
- most/all software packages (including BEAST) estimate the log marginal likelihood: log(p(D | M))

Calculating marginal likelihoods

Methods of general applicability:

- the posterior arithmetic mean estimator (pAME; Aitkin, 1991)
- the arithmetic mean estimator (AME/ILP; but a misnomer)
- the importance sampling estimators, and particularly the harmonic mean estimator (HME) (Newton and Raftery, 1994)
- the stabilized harmonic mean estimator (sHME) (Redelings and Suchard, 2005)

No additional analysis required

- path sampling (Gelman, 1998; Ogata, 1989), applied in phylogenetics (Lartillot and Philippe, 2006)
- stepping-stone sampling (Xie et al., 2011)
- generalised stepping-stone sampling (Fan et al., 2011; Baele et al., 2016)

Additional analysis required

The arithmetic mean estimator (AME)

$$p(D \mid M) = E_{\text{prior}}[p(D \mid \theta, M)]$$
$$\simeq \frac{1}{K} \sum_{k=1}^{K} p(D \mid \theta_k, M).$$

- a.k.a. the prior arithmetic mean estimator (but a misnomer)
- integrates the likelihood against the model prior (unbiased)
- does **not** use samples from the likelihood obtained from an MCMC analysis used to estimate parameters
- the high-likelihood region can be very small, hence unless K is very large, the sample drawn from the prior will contain virtually no points from the high-likelihood region, resulting in a (very) poor estimate of p(D | M)

What about our MCMC output?

How does this relate to your (regular) MCMC analysis (which can already be quite time-consuming, taking days or even weeks)?

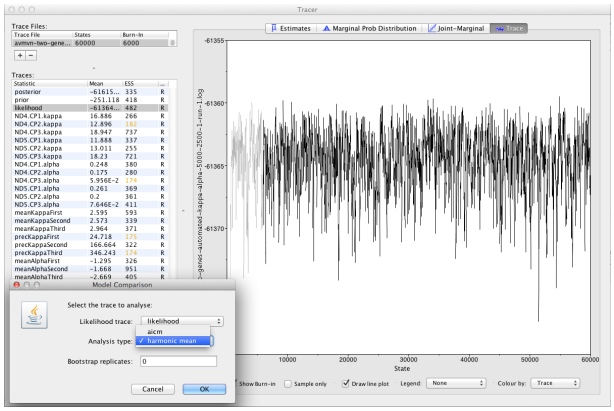
Can you use your MCMC output to compute marginal likelihoods?

Yes: some marginal likelihood estimators (HME and sHME) and the AICM use the likelihood samples collected during an MCMC run.

No: their accuracy and performance is poor and their outcomes unreliable, thereby completely invalidating their computational advantage. **Hence, their use should be avoided.**

Accurate marginal likelihood estimation hence requires an additional MCMC analysis for each model!

What can we do in Tracer?



What can we do in Tracer?

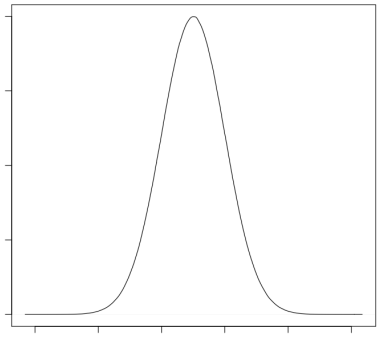
- you can load the .log file from your parameter estimation run into Tracer (by default, this file will contain a column of likelihood values)
- using this likelihood trace/column, you can estimate the HME (but currently not the sHME) and the AICM in Tracer
- PS/SS and GSS are much more accurate marginal likelihood estimators but require additional calculations/analyses
- as such, the PS/SS and GSS estimators can **NOT** be estimated in Tracer (not what Tracer is for)

DO NOT load the .log output file generated by PS/SS and GSS into Tracer

Avoid performing model selection in Tracer altogether

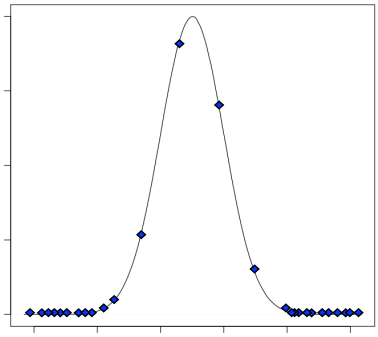
Intuitive reasoning

suppose this is the likelihood (curve) we need to explore to obtain an accurate estimate of the marginal likelihood



Intuitive reasoning: ILP

integrating the likelihood over the prior (ILP) draws random values for the parameters and then calculates the corresponding likelihood

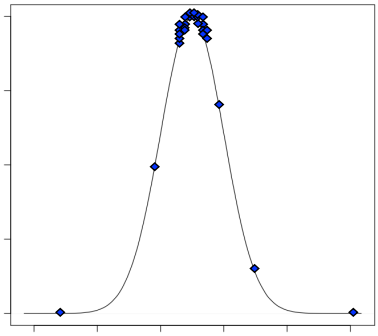


most of the likelihoods will hence be (very) low

but for the ILP to work accurately, we need a fair amount of high likelihood values

Intuitive reasoning: HME/sHME

the HME/sHME use samples from the posterior, which tend to be mostly high-likelihood values



only once every so often does the MHG algorithm accept a low value

for the HME/sHME to work well, samples from the low likelihood region are needed

Path sampling

requires samples from a series of power posteriors, not just the posterior, along a path between prior and posterior:

$$q_\beta(\theta) = p(Y \mid \theta, M)^\beta p(\theta \mid M)$$

- reduces to the posterior when $\beta = 1$
- reduces to the prior when $\beta = 0$

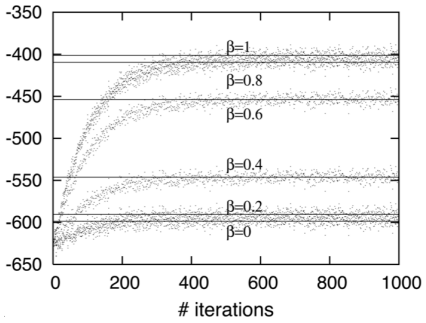
- slow / computationally demanding
- slower convergence than HME/sHME/AICM

Rationale of path sampling

run several power posteriors, including the posterior and the prior, and collect samples from each power posterior

Example: run 6 power posteriors, for different values of beta: 1.0 (**posterior**), 0.8, 0.6, 0.4, 0.2 and 0.0 (**prior**).

After each power posterior has converged, collect samples.



Path sampling: the estimator

- points θ_k are saved before each update of β; let us denote $(\beta_k , \theta_k)_{k=0..K}$ the series of points obtained this way
- one can start at $\beta = 1$ (i.e. the posterior), equilibrate the MCMC, and then **progressively decrease β** (BEAST), while sampling along the path down to $\beta = 0$ (i.e. the prior)
- one has in particular $\beta_0 = 0$, $\beta_K = 1$, and $\forall k\ 0 \le k < K$, $\beta_{k+1} - \beta_k = \delta\beta$ (i.e. the original approach assumes equidistant β's; Lartillot and Philippe, 2006)
- the log marginal likelihood estimator is given by:

$$\hat{\mu}_{qs} = \frac{1}{K}\left(\frac{1}{2}U(\theta_0) + \sum_{k=1}^{K-1} U(\theta_k) + \frac{1}{2}U(\theta_K)\right)$$

Rationale of path sampling

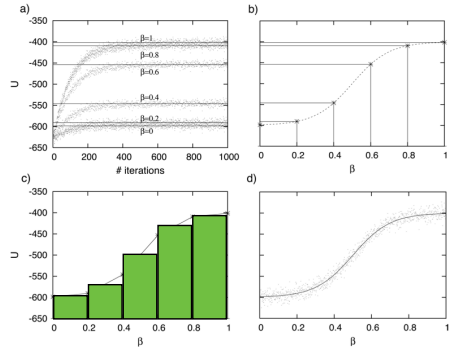
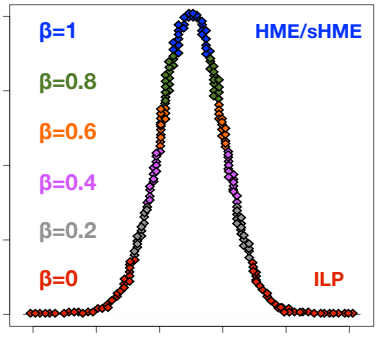


FIGURE 1. Rationale of the thermodynamic integration method. a, A series of independent chains are run under different values of β, and for each of them, the mean posterior expectation of the potential $U = \partial \ln q_\beta / \partial \beta$ is computed (horizontal lines). b, These mean posterior expectations are plotted against β. c, The integral of the curve is estimated by the Simpson procedure. d, Illustration of the quasistatic version, in which β moves continuously from 0 to 1 during MCMC (see text for details).

Intuitive reasoning: Path Sampling

PS combines the sampling regions from ILP and HME/sHME, and adds samples from a series of power posteriors

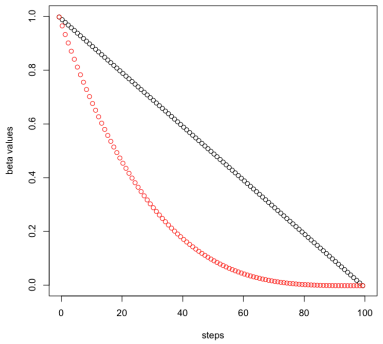


PS (and later SS) hence sample the likelihood curve more accurately

leading to PS/SS drastically outperforming the other marginal likelihood estimators

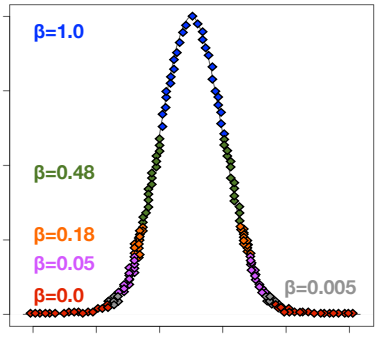
Path sampling: a better path

- in the common situation where the likelihood is much more concentrated than the prior, the shape of the power posterior is relatively stable except near β = 0
- placing more computational effort near 0 is hence sensible and leads to a substantial increase in the efficiency of the estimator
- use a Beta(1.0 ; α) distribution to select values of β
- shown here: α = 0.3 (Xie et al., 2011)
- different approaches available in BEAST



Path sampling: a better path

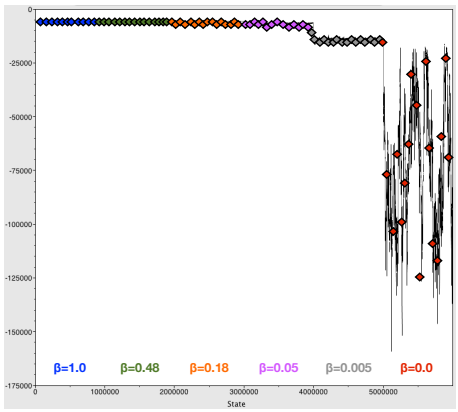
PS/SS can use a Beta(1.0 ; 0.3) distribution to determine the series of power posteriors from which to sample



more power posteriors closer to the prior are being sampled from

leading to more accurate ML estimates with PS/SS using the same amount of computation power

Path sampling: a better path



gradual decrease of the sampled likelihood values with decreasing β

only when a small portion of the data (i.e. β close to zero) remains are very low likelihood values observed

'long way' from posterior to prior

Path sampling: conclusions

- a striking discrepancy between the HME and PS, due to a lack of reliability of the HME
- the HME **overestimates** the marginal likelihood
- this overestimation is more pronounced in the case of higher dimensional models, which implies that the harmonic estimator will be effectively biased in favor of such models

- PS is included in more software packages, such as BEAST, nowadays and should hence be used instead of the HME
- one downside: PS is computationally (much) more demanding than the HME, sHME and AICM

Stepping-stone sampling

requires samples from a series of power posteriors, like path sampling, along a path between prior and posterior:

$$q_\beta(\theta) = p(Y \mid \theta, M)^\beta p(\theta \mid M)$$

- PS and SS traverse the same path between posterior and prior, the same samples can be used for both estimators (i.e. buy one, get one for free)
- differs from path sampling **in the way the collected samples are used to estimate the log marginal likelihood**
- does not need to sample the posterior, an initial run that converges towards the posterior is used to burn-in the MCMC chain

PS/SS: two for the price of one

- when performing PS/SS in a software packages (e.g. BEAST), collecting samples from the path between posterior and prior is the computationally demanding step
- from this one collection, two marginal likelihood estimators (i.e. PS and SS) are computed, which only takes a few minutes at most

- **BEAST prints both estimates to the screen**, i.e. the actual log marginal likelihoods are not stored in a file
- save the screen output when submitting such calculations to a server/ computer cluster/grid system

Stepping-stone sampling: the estimator

$$\log \hat{r}_{SS} = \sum_{k=1}^{K} \log(\hat{r}_{SS,k})$$
$$= \sum_{k=1}^{K} [(\beta_k - \beta_{k-1}) \log L_{\max,k}]$$
$$+ \sum_{k=1}^{K} \log \left(\frac{1}{n} \sum_{i=1}^{n} \exp \left\{ (\beta_k - \beta_{k-1}) \right. \right.$$
$$\left. \left. \times [\log f(\mathbf{y}|\boldsymbol{\theta}_{k-1,i}) - \log L_{\max,k}] \right\} \right)$$

- log($\hat{r}_{SS}$) is biased, and its bias appears to be directly proportional to its variance, which can be alleviated by increasing K (i.e., the number of power posteriors)

Stepping-stone sampling: phylogenetic example

- HME converges much faster than PS and SS, but to the wrong value
- variance of PS and SS decreases with increasing K
- SS converges much faster to the marginal likelihood than PS
- given sufficiently large K, PS and SS converge to the same result

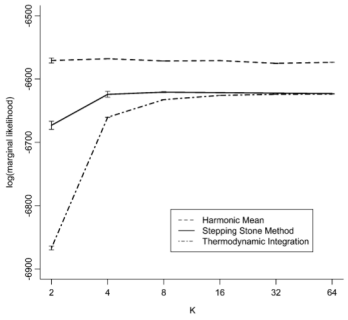


FIGURE 5. Log marginal likelihood for three estimation methods as a function of the number of β intervals, *K*, for the green plant Ribulose Bisphosphate Carboxylase/Oxygenase large subunit (*rbc*L) example. β values are evenly spaced quantiles from a Beta(0.3,1.0) distribution. Error bars represent ±1 standard error based on 30 independent MCMC analyses.

Stepping-stone sampling: conclusions

- SS is a more efficient and less biased estimator for the (log) marginal likelihood than PS
- less computational effort is required for SS compared to PS
- which settings, i.e. the number of β values and the length of the chain at each value of β, produce consistent estimates is subject to debate
- like PS, SS outperforms the HME, sHME and AICM in selecting the best model from a collection of candidate models

PS/SS/GSS: suggestions

- which settings, i.e. the number of path steps and the length of the chain at each step, produce a good estimate of the log marginal likelihood?
- when performing model selection, I suggest to **use a total number of iterations equal to the standard MCMC run** used to estimate parameters
- for example, if it takes your standard MCMC run 100 million iterations to yield good ESS values, try running 100 path steps of 1 million iterations each
- tutorial online (XML code; for BEAST 1.7.x and up): http://rega.kuleuven.be/cev/ecv/tutorials/model-selection-tutorial

Model testing problems: HIV-1 example

- HIV-1 data: 162 taxa, 997 bp (Worobey et al., Nature, 2008)
- 'The inability to strongly reject the model with a constant population size prior is counterintuitive because it is clear that the HIV-1 population size has increased notably. We speculate that this finding might be due to the simplest model providing a good fit to a relatively short, information-poor alignment, in comparison with more parameterized models.'

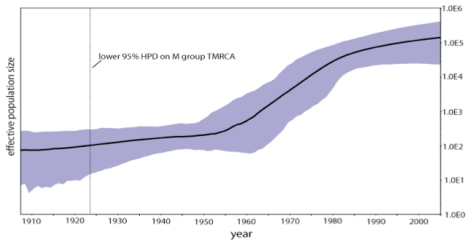
Table 1 | HIV-1 M group TMRCA estimates from BEAST analyses under different coalescent tree priors

Coalescent tree prior	DRC60 and ZR59 excluded*	DRC60 and ZR59 included
Constant	**1933 (1919–1945)**†, 0.0	**1921 (1908–1933)**†, 0.0
Exponential	1907 (1874–1932), −3.5 ± 0.8	1914 (1891–1930), −2.1 ± 1.5
Expansion	1882 (1834–1917), −2.7 ± 0.8	**1902 (1873–1922)**†, −1.6 ± 1.5
Logistic	1913 (1880–1937), −2.3 ± 0.8	1913 (1891–1930), −3.2 ± 1.5
Bayesian skyline plot	1882 (1831–1916), −2.7 ± 0.8	**1908 (1884–1924)**†, −0.4 ± 1.5

Shown for each coalescent tree prior is the median, with the 95% highest probability distribution of TMRCA in parentheses. Also shown is the $\log_{10}$ Bayes factor difference in estimated marginal likelihood (± estimated standard error) compared with the coalescent model with strongest support.
*Concatenated *gag-pol-env* fragments available for either or both of ZR59 and DRC60 (994 nucleotides total, 507 from DRC60).
†TMRCAs for the best-fit model and models not significantly worse than it are written in bold.

Model testing problems: HIV-1 example

- Bayesian skyline plot of HIV-1 group M. The plot begins at the median posterior TMRCA (1908). The bold line traces the inferred median effective population size over time with the 95% HPD shaded in blue



Model testing problems: HIV-1 example

- Analysis of the HIV-1 data set using sHME, AICM, PS and SS

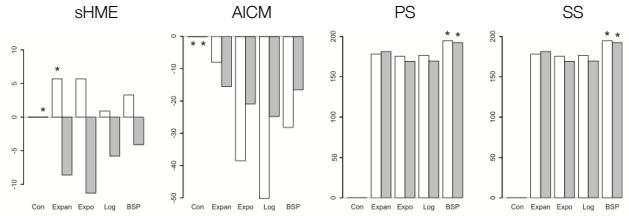


FIG. 1. Differences in log-marginal likelihood estimates and AICM for two independent fittings (first fitting shown in white and second in gray) of the HIV data set using the HME, posterior-simulation Akaike information content (AICM), PS, and SS sampling. For each estimator, the constant population size model (Con) was used as the reference model, and the top-performing model for each fitting is indicated with a star (*). For all estimators, we employ equal amounts of computational work (MCMC iterations) as well as an equal numbers of samples from which to estimate the marginal likelihood. The HME shows drastic differences in the overall ranking of the demographic models and, depending on the fitting, may very well select a constant population size as the preferred coalescent prior. The AICM is consistent across both fittings but selects a constant population size above all other coalescent priors. PS and SS consistently select the BSP coalescent prior as the optimal choice and put the constant population size far behind the other coalescent priors. PS and SS indicate that the expansion growth model (Expan) yields the second highest fit, whereas the exponential (Expo) and logistic (Log) growth models yield similar performance.

Model testing: simulation results

Table 1. Marginal Likelihood Estimator Performance for 100 Simulated Data Sets under Various Coalescent Priors Using the HME, AICM, PS, and SS.

Coalescent Prior	Growth Rate	HME	AICM	PS	SS	Log BF HME	ΔAICM	Log BF PS	Log BF SS
Constant	—	48	59	72	72	0.61	0.57	1.76	1.76
Exponential	0.010	50	45	57	57	0.28	0.20	−0.81	−0.80
Exponential	0.025	59	73	92	92	−1.33	−1.36	−6.81	−6.81
Exponential	0.050	80	99	100	100	−4.43	−4.34	−12.54	−12.54
Exponential	0.100	78	100	100	100	−7.75	−7.66	−18.24	−18.24

We employed equal amounts of computational work (MCMC iterations) for all estimators as well as an equal number of posterior samples being used to estimate the marginal likelihood. The HME, PS, and SS columns report the number of correct classifications obtained out of 100 simulations. The log BF HME, log BF PS, and log BF SS report the mean log BF over all replicates between the constant population size and exponential growth coalescent priors (a positive number indicates a preference for the constant population size), whereas ΔAICM reports the mean difference of the AICM values across all replicates.

Conclusions:

- an exponential demographic prior with a growth rate of 0.01 is a difficult case for each estimator
- HME is unable to reach an accuracy higher than 80%
- AICM outperforms HME in all but one case
- PS/SS outperform HME and AICM in all cases

now in BEAST!
(Baele et al., MBE, 2012)

Model testing: simulation results

Table 3.1: model selection performance for 100 simulated datasets, consisting of 32 taxa, under either a balanced or Yule tree and two relaxed molecular clock models using HME, sHME, AICM, PS, SS and the maximum *a posteriori model* estimated under BMA. The columns report the number of correct classifications obtained out of 100 simulations.

Tree	Clock	Length	HME	sHME	AICM	PS	SS	MAP
Balanced	UCED	1.000	92	100	100	94	94	90
Balanced	UCLD	1.000	28	5	1	99	99	99
Yule	UCED	1.000	92	100	100	99	99	97
Yule	UCLD	1.000	11	1	1	61	61	65
Yule	UCED	2.500	89	100	100	98	98	99
Yule	UCLD	2.500	26	5	9	83	82	81
Yule	UCED	5.000	78	99	99	98	98	98
Yule	UCLD	5.000	38	11	12	82	82	82

Conclusions:

- HME and AICM perform poorly, almost no correct classifications when simulated under a relaxed lognormal clock model
- PS/SS outperform HME and AICM in all cases

Generalised stepping-stone sampling

requires samples from a series of power posteriors, along a path between reference/working prior and posterior:

$$q_\beta(\theta) = [p(Y \mid \theta,M)p(\theta \mid M)]^\beta p_0(\theta \mid M)^{1-\beta}$$

- reduces to the original SS method if the reference/working distribution is equal to the actual prior
- in practice, samples from the posterior distribution ($\beta = 1$) are used to parameterize the joint reference/working distribution $p_0(\theta|M)$
- we will use kernel density estimation (KDE) to construct reference/ working priors for each of the parameters being estimated

GSS: the estimator

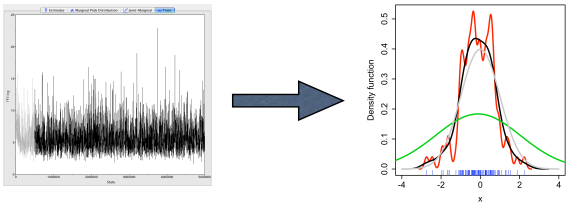
Numerical stability can be improved by factoring out the largest sampled term, $\eta_k = \max_{1 \leqslant i \leqslant n} \{f(\mathbf{y}|\theta_{k-1,i}, M)\, \pi(\boldsymbol{\theta}_{k-1,i}|M)/\pi_0(\boldsymbol{\theta}_{k-1,i}|M)\}$:

$$\log \hat{r} = \sum_{k=1}^{K} \log \hat{r}_k$$

$$= \sum_{k=1}^{K} [(\beta_k - \beta_{k-1}) \log \eta_k]$$

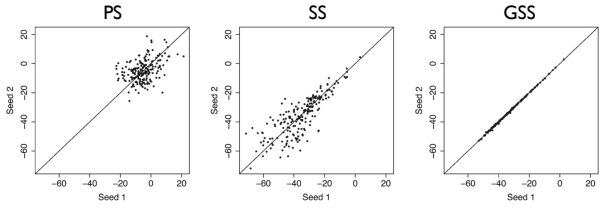
$$+ \sum_{k=1}^{K} \log \left\{ \frac{1}{n} \sum_{i=1}^{n} \left[\frac{f(\mathbf{y}|\boldsymbol{\theta}_{k-1,i}, M)\, \pi(\boldsymbol{\theta}_{k-1,i}|M)}{\eta_k\, \pi_0(\boldsymbol{\theta}_{k-1,i}|M)} \right]^{\beta_k - \beta_{k-1}} \right\}$$

GSS: working priors



- reduces to the original SS method if the reference/working distribution is equal to the actual prior
- in practice, samples from the posterior distribution (β = 1) are used to parameterize the joint reference/working distribution $p_0(\theta|M)$
- we will use kernel density estimation (KDE) to construct reference/ working priors for each of the parameters being estimated

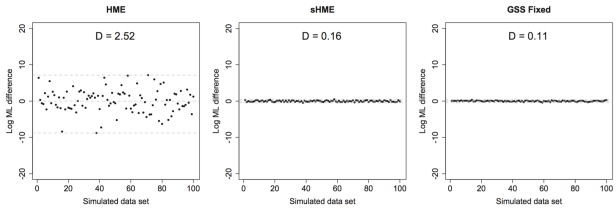
Original GSS: fixed tree topology



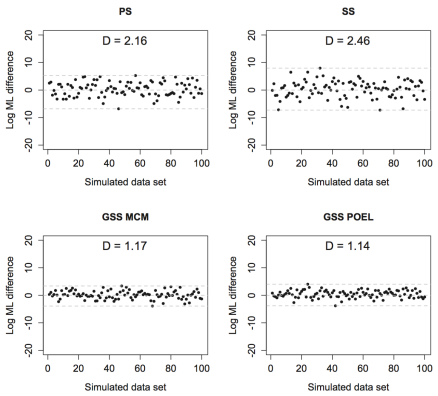
- original GSS publication (Fan et al., 2011): fixed tree topology
- GSS analyses with different starting seeds yield almost identical MLEs
- hence much lower variance compared to PS/SS

GSS: phylogenetic uncertainty

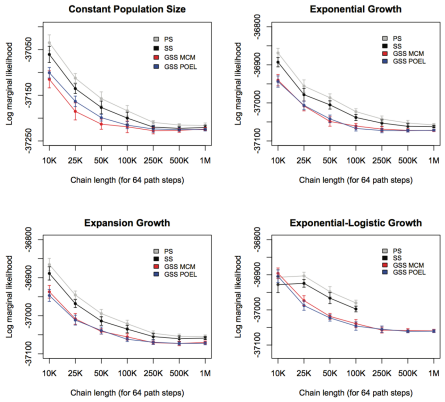
- the fixed tree topology restriction has been relaxed
- e.g. for use in a coalescent-based framework such as BEAST
- 2 working priors for coalescent models proposed (Baele et al., 2016)
  - matching coalescent model (MCM): for simple parametric models
  - product of exponential distributions (POEL): general use



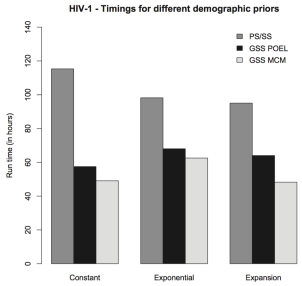
GSS: phylogenetic uncertainty (variance)



GSS: coalescent models



GSS: decreased run time



- GSS does not need to explore the prior, which avoids computing the likelihood for highly unlikely parameter values, which may lead to numerical instabilities
- combined with a "shorter" path to be traversed, this leads to a drastic performance increase (dependent on the actual reference/working prior)

Bayesian model testing: priors

- a key aspect of any Bayesian analysis is setting priors on the parameters being estimated in the process
- unless there is a priori knowledge concerning some of the parameters, uninformative (but proper) priors are used

- **improper priors should be avoided**, although many analyses are still performed with such priors
- more importantly: improper priors are to be avoided at all cost when performing path sampling (PS) and stepping-stone sampling (SS) as these approaches gradually decrease the contribution of the data to the posterior and include an exploration of the prior

Bayesian model testing: priors

A proper prior is a probability distribution that integrates to 1.

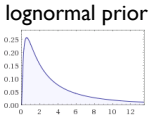
The frequently used constant function on an infinite interval is often inaccurately called a uniform distribution, although it is actually an example of an improper prior.

When no prior is specified in certain software packages, an improper prior may be assumed (e.g. BEAST).

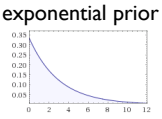
Bayesian model testing: proper priors

A proper prior is a probability distribution that integrates to 1.

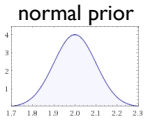
lognormal prior



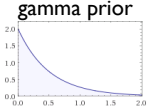
exponential prior



normal prior



gamma prior



uniform prior

but not Uniform[-∞,+∞]

End of lecture…

End of lecture / start of practical

Main papers discussed (1)

- Newton, M.A., Raftery, A.E. (1994) Approximating Bayesian inference with the weighted likelihood bootstrap. J. R. Stat. Soc. B. 56:3-48.
- Kass, R.E., Raftery, A.E. (1995) Bayes factors. J. Am. Stat. Assoc. 90:773-795.
- Lartillot, N., Philippe, H. (2006) Computing Bayes factors using thermodynamic integration. Syst. Biol. 55:195-207.
- Xie, W., Lewis, P.O., Fan, Y., Kuo, L., Chen, M.H. (2011) Improving marginal likelihood estimation for Bayesian phylogenetic model selection. Syst. Biol. 60:150-160.
- Fan, Y., Wu, R., Chen, M.H., Kuo, L., Lewis, P.O. (2011) Choosing among partition models in Bayesian phylogenetics. Mol. Biol. Evol. 28:523-532.

Main papers discussed (2)

- Baele, G., Lemey, P., Bedford, T., Rambaut, A., Suchard, M. A., Alekseyenko, A. V. (2012) Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty. Mol. Biol. Evol. 29(9), 2157-2167.
- Li, W. L. S., Drummond, A. J. (2012) Model averaging and Bayes factor calculation of relaxed molecular clocks in Bayesian phylogenetics. Mol. Biol. Evol. 29:751-761.
- Baele, G., Li, W. L. S., Drummond, A. J., Suchard, M. A., Lemey, P. (2013) Accurate model selection of relaxed molecular clocks in Bayesian phylogenetics. Mol. Biol. Evol. 30(2), 239-243.
- Baele, G., Lemey, P., Suchard, M. A. (2016) Genealogical working distributions for Bayesian model testing with phylogenetic uncertainty. Syst. Biol. 65:250-264.