

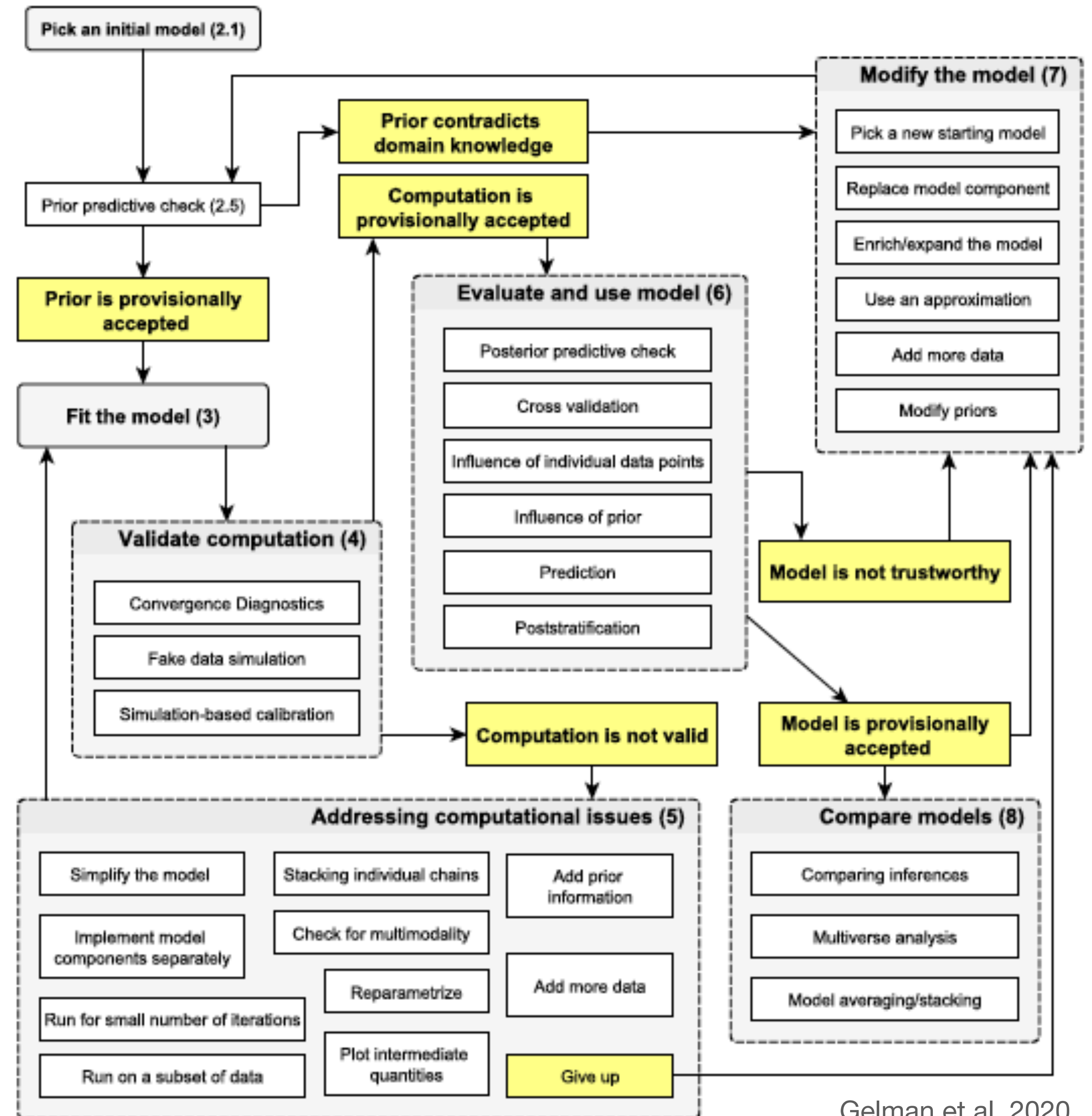
Applied Bayesian Modeling module 12: **Bayesian workflow**

Lecture material (slides, notes, videos) are licensed under CC-BY-NC 4.0. Code is licensed under BSD-3

Leontine Alkema (lalkema@umass.edu), Fall 2022

Background

- Setting:
 - You have a research question and a data set
 - Example: forecast COVID19 in Switzerland
 - Goal: Let's develop a Bayesian model to answer the question!
- How to go about model development?
 - Consider an iterative process...
 - Contrast this with eg clinical trial research: Sample size calculation & analysis plan -> data collection -> analysis as per plan

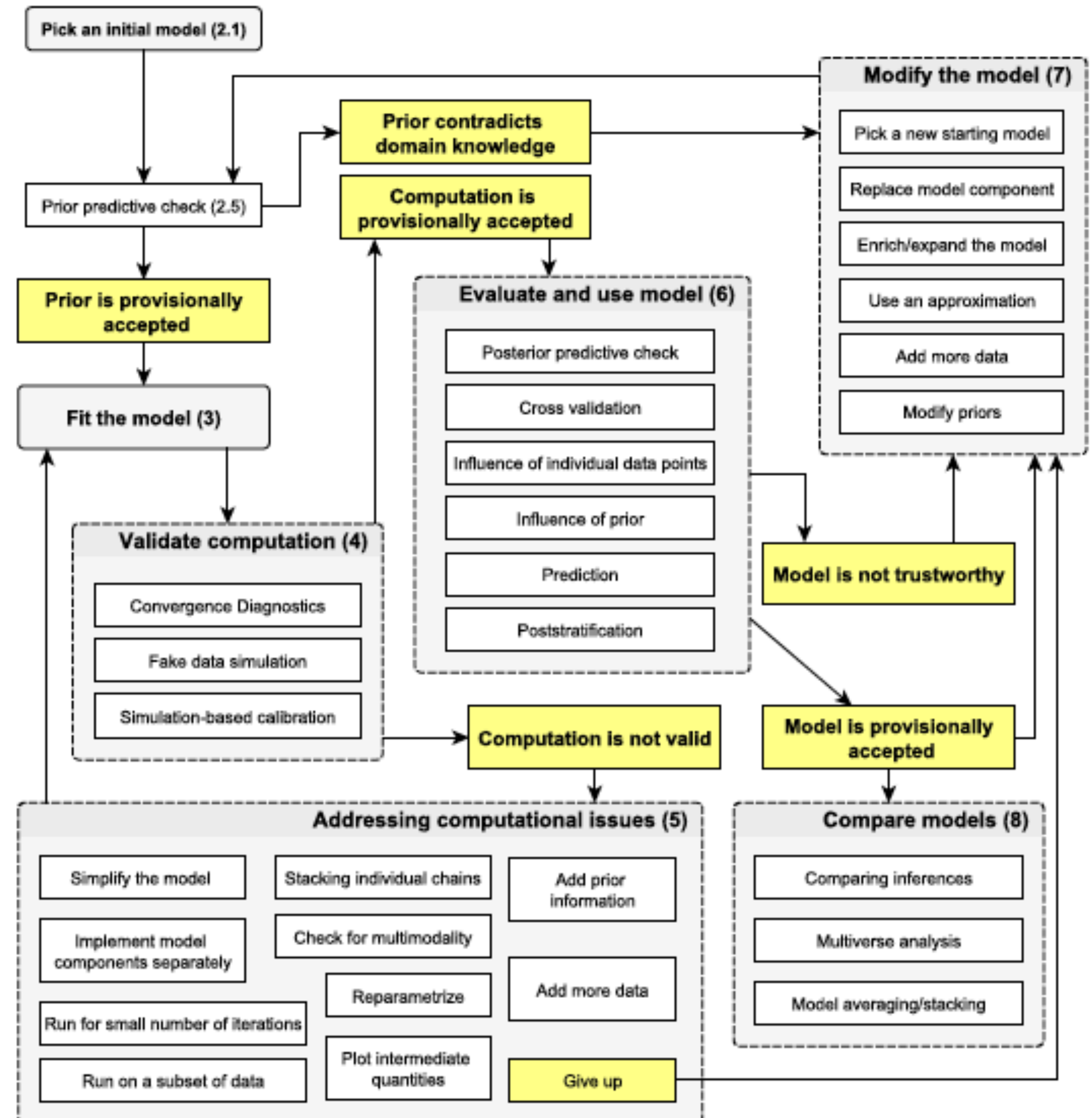


Bayesian workflow

- Bayesian workflow refers to the steps/iterative process used to develop a model
 - Build, debug, improve, expand
- References
 - Gelman et al (2020). “Bayesian Workflow.” <http://arxiv.org/abs/2011.01808>
 - Gabry et al (2019). “Visualization in Bayesian Workflow.” JRSS Series A 182, no. 2 (2019): 389–402. <https://doi.org/10.1111/rssa.12378>
 - Grinsztajn et al (2021). “Bayesian Workflow for Disease Transmission Modeling in Stan.” Statistics in Medicine 40, no. 27 (2021): 6209–34. <https://doi.org/10.1002/sim.9164>
- By writing about Bayesian workflows, the authors (Gelman et al in particular)
 - Share guidance to help with model development
 - Help work towards formalize/systematize Bayesian model development
- Opinionated?

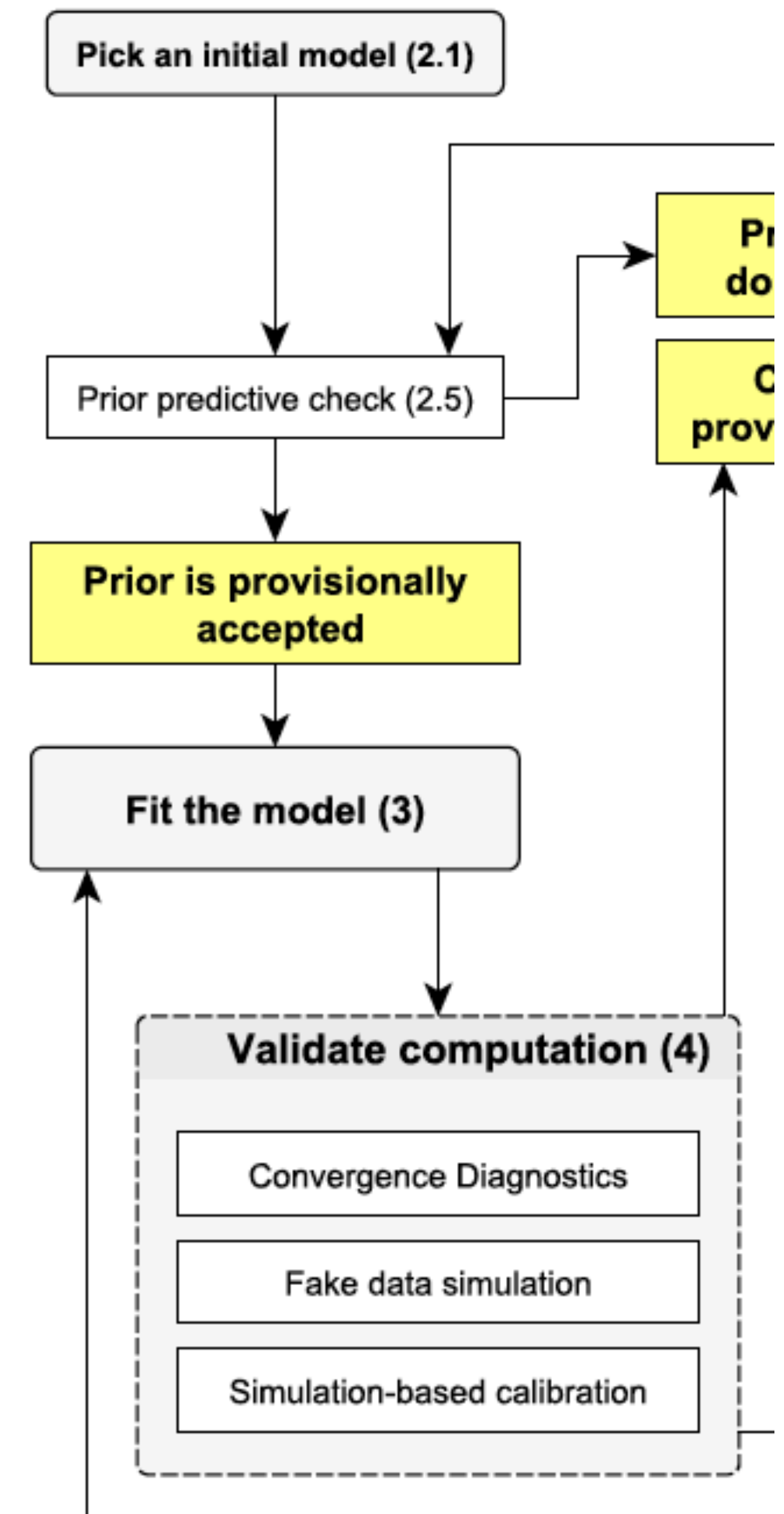
Elements

- (+) *Exploratory data analysis*
- Choose initial model(s)
- Troubleshoot before fitting
 - Prior predictive check
- Validate computation
 - Address computational issues
- Evaluate and use model
- Modify the model
- Model comparison



Getting started

- Choose initial model(s)
 - Adopt a model from a previous analysis
 - Consider starting simple, to then expand
- Troubleshoot before fitting to actual data
 - Prior predictive check
- Fit the model and validate computation
 - Consider first running the model with a smaller number of iterations (fit fast, fail fast)
 - Then consider MCMC diagnostics
 - So far, we discussed general MCMC diagnostics to check mixing of chains (Rhat and effective sample size)
 - When using default computation in Stan, some additional warnings may come up that are helpful to detect issues

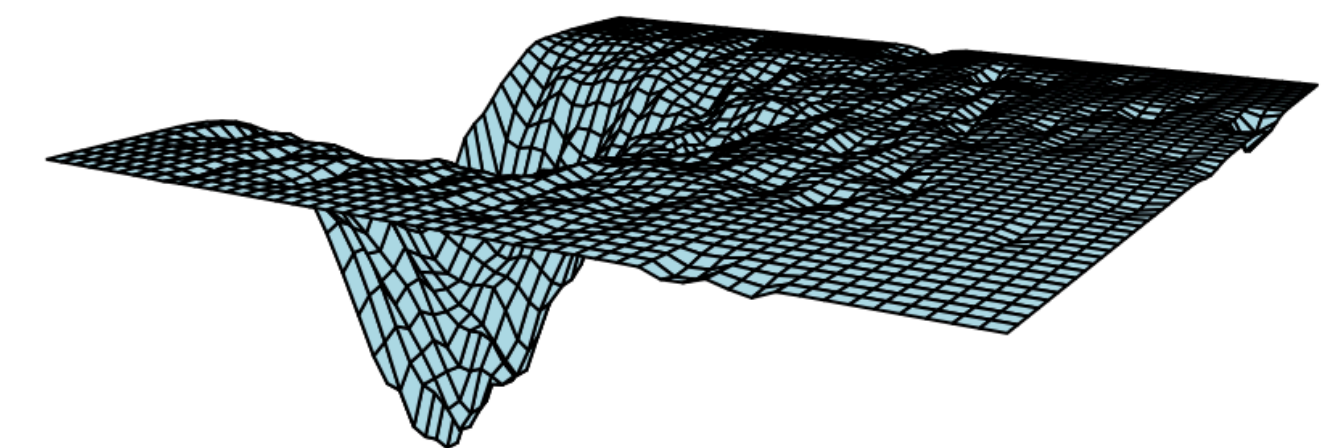
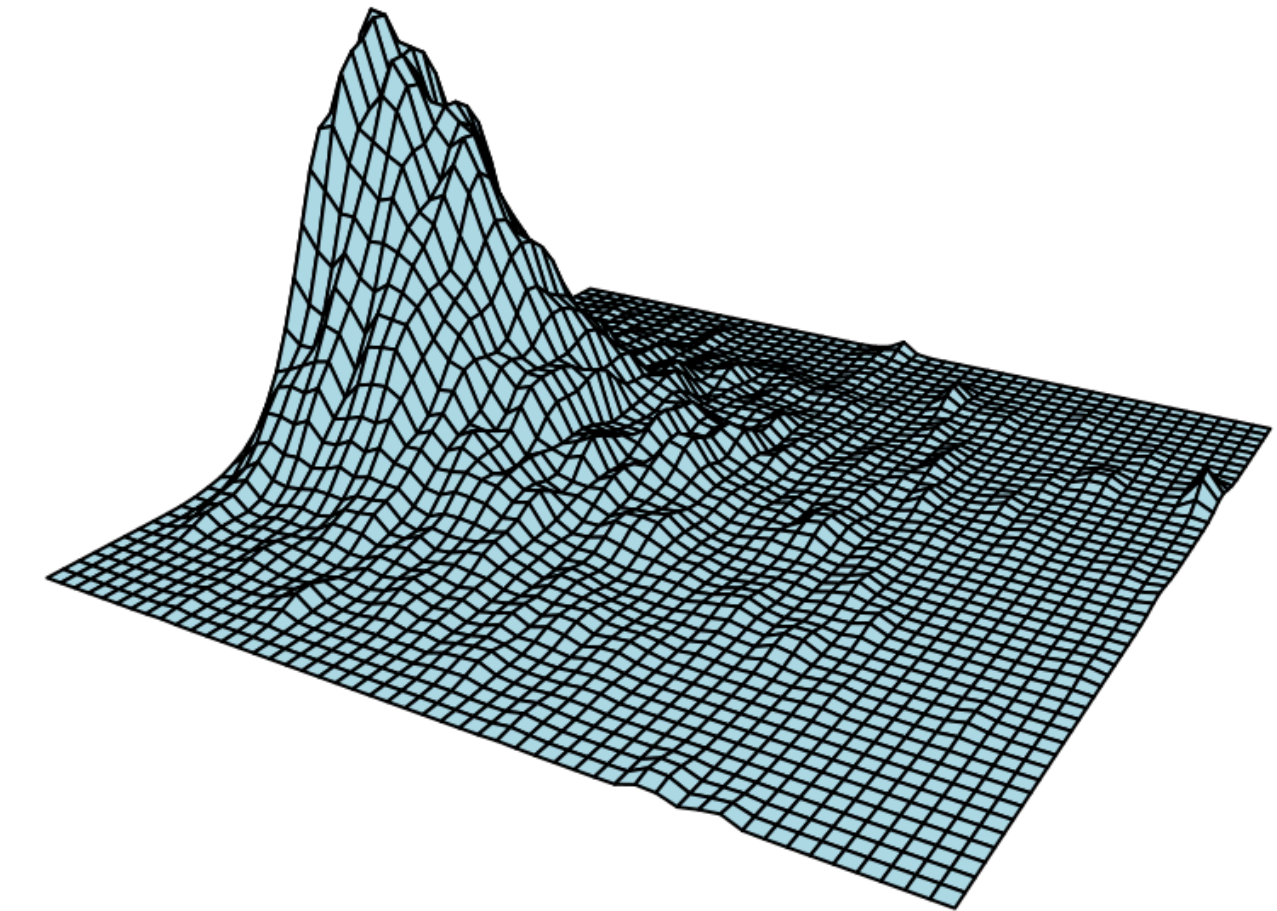


Computation

- More details on Stan's default MCMC algorithm (HMC/NUTS)
 - Focus on main ideas, to understand the warnings and tuning parameters
- References
 - Nice intro to HMC: https://bookdown.org/marklhc/notes_bookdown/markov-chain-monte-carlo.html#hamiltonian-monte-carlo
 - Warnings: <https://mc-stan.org/misc/warnings.html>

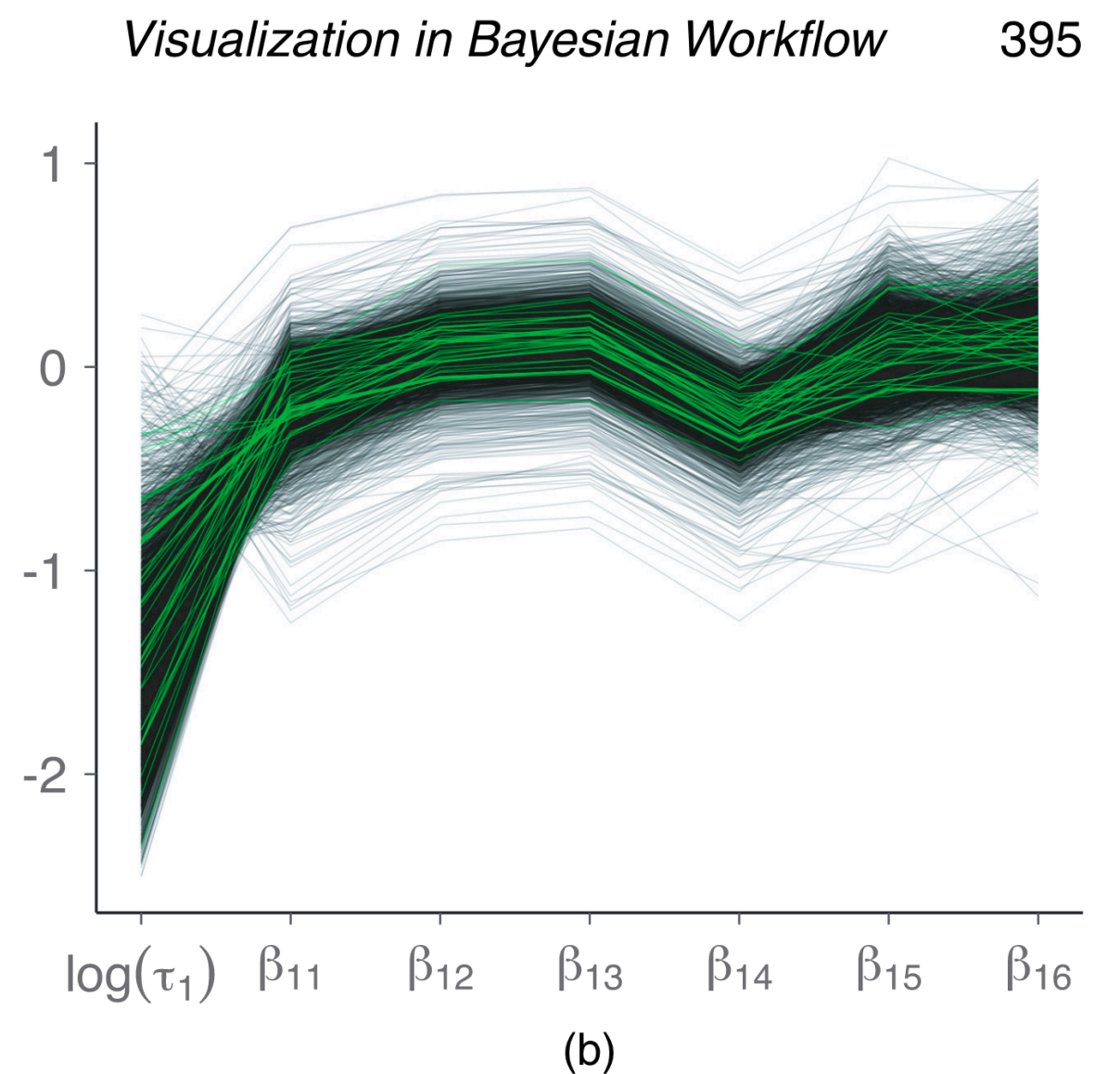
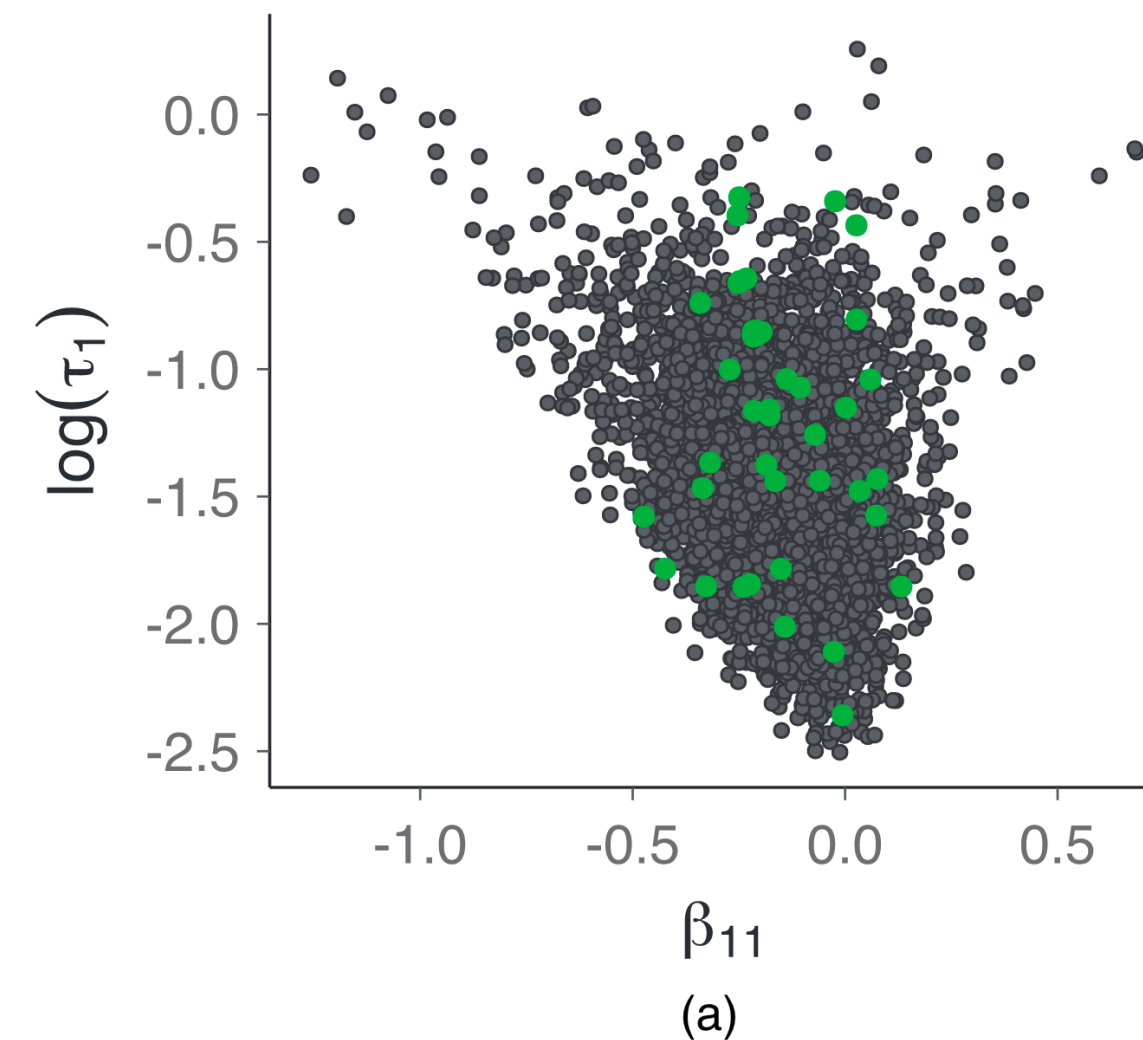
Under the hood

- Stan implements gradient-based Markov chain Monte Carlo (**MCMC**) algorithms for Bayesian inference, with the default algorithm being the No-U-Turn sampler (**NUTS**), a variant of Hamilton Monte Carlo (**HMC**)
- HMC: sampling from posterior == “considering the inverted posterior as a park for ice skating” (Lai, 2019)
 - You are randomly given a direction and an energy level
 - Where you end up (taking account of gravitational force [derivatives]) == a sample from the posterior
- In practice: use numerical integration (leapfrog integrator)
 - Break path into discrete segments, referred to as leapfrog steps
 - Need to choose a step size and number of steps
 - NUTS version of HMC automates the algorithm
 - Once you stop somewhere, do a Metropolis step (accept or reject the proposed value), to make up for issues with numerical integration



Stan warnings: divergences

- A divergence arises when the simulated Hamiltonian trajectory departs from the true trajectory as measured by departure of the Hamiltonian value from its initial value.
 - E.g. when log posterior doesn't have a continuous derivative
- This is a problem because it may mean that posterior distribution isn't thoroughly explored
- Check where divergences they occur to see if there is a problematic area (combination of parameter values)
- ShinyStan or Bayesplot visualization



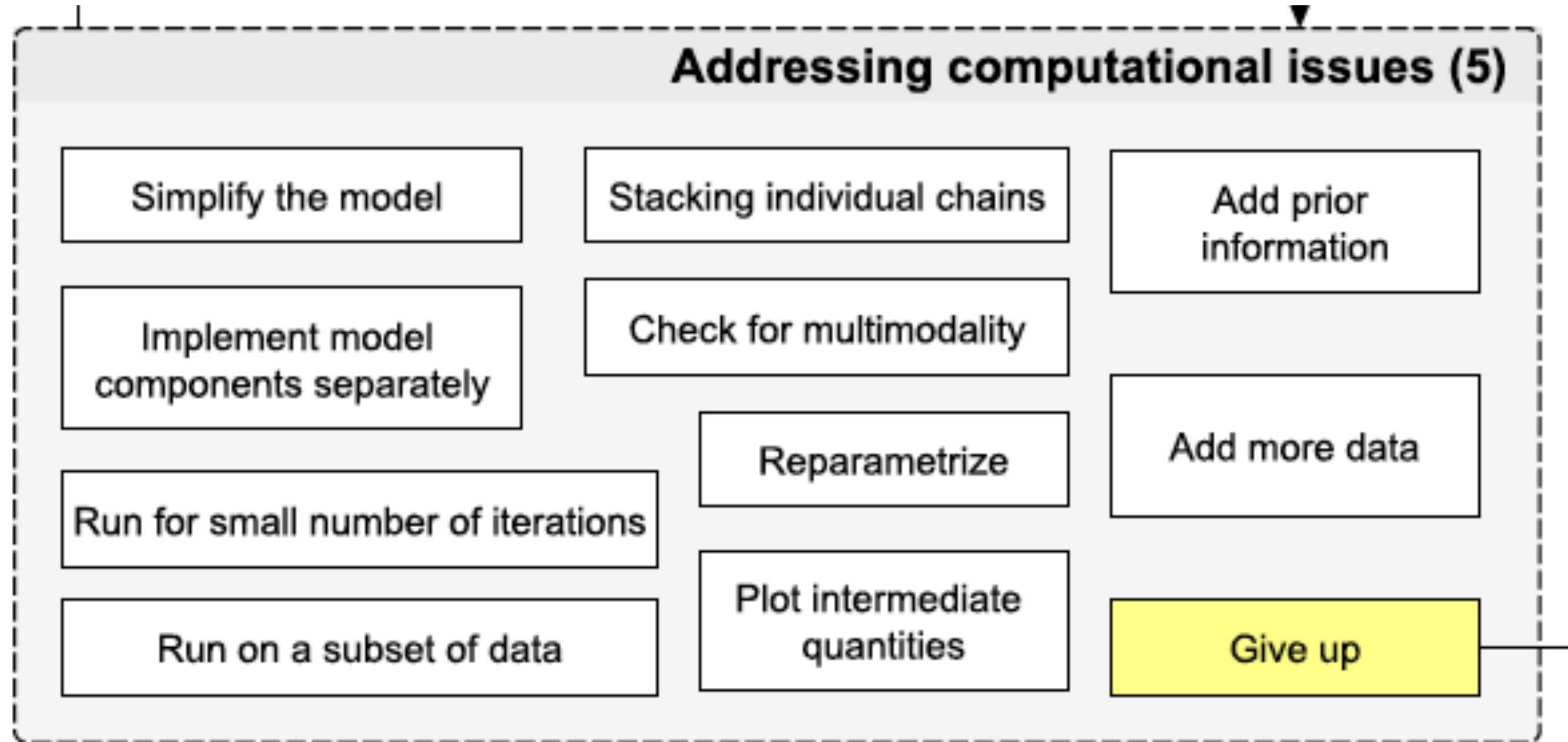
Stan warnings: divergences (ctd)

- Only a few divergences in different places and good other diagnostics (Rhat and ESS), often a ok
 - Could be falsely labeled as a divergence
- If there is a systematic issue:
 - Reparametrize
 - Consider the “folk theorem of statistical computing”:
 - When you have computational problems, often there’s a problem with your model (Yao et al, 2020). Not always—sometimes you will have a model that is legitimately difficult to fit—but many cases of poor convergence correspond to regions of parameter space that are not of substantive interest or even to a nonsensical model (Gelman et al, 2020).
 - Reconsider your model
- If you do want to throw more computational power at it: consider Stan tuning parameter
 - Adapt_delta = target acceptance rate in metropolis step
 - Defaults at 0.8
 - Increase to reduce the stepsize so algorithm won’t go too far away in each jump (and is more likely to accept proposed value)

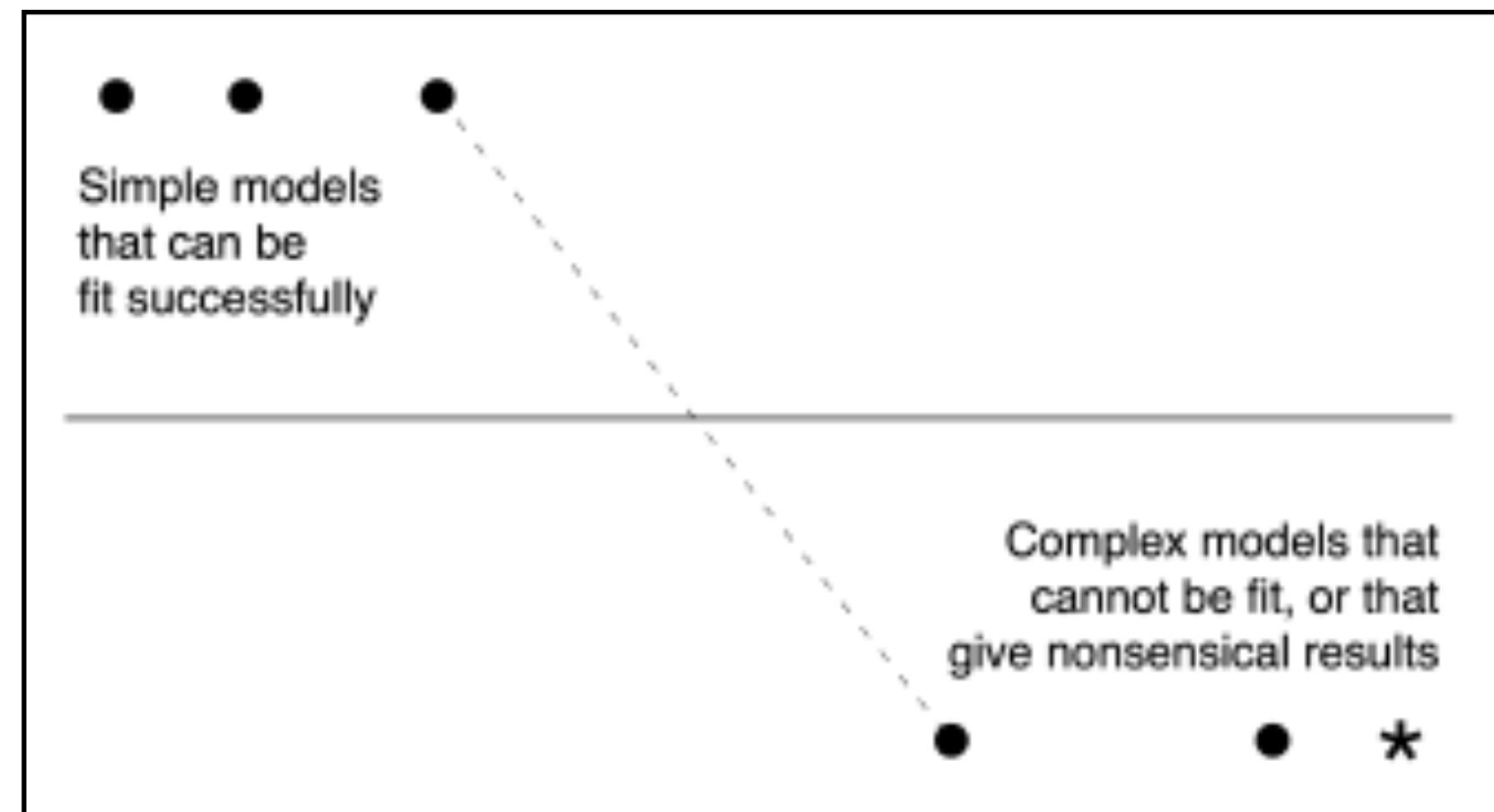
Treedepth

- You may get a warning on max treedepth
 - An efficiency concern as compared to a validity concern
 - But reaching max treedepths may (again) suggest a model specification issue
- Tuning parameter “max_treedepth” controls maximum number of leapfrog steps
 - Not typically recommended to increase as per stan warning reference

Addressing computational issues

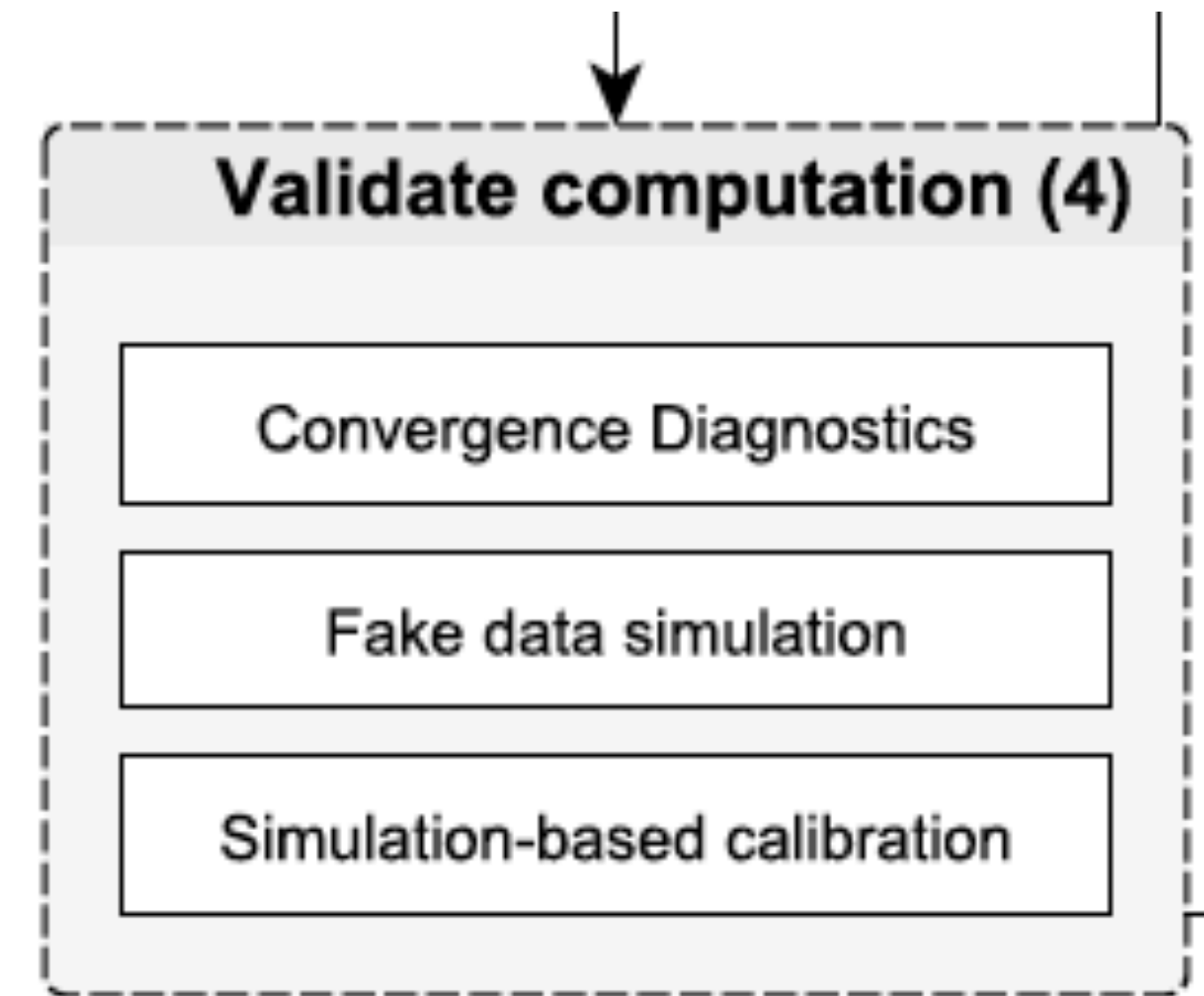


- Reparametrize
- Consider simpler models/model components
- For debugging, fit to
 - Data that's simulated from the model
 - Subset of data



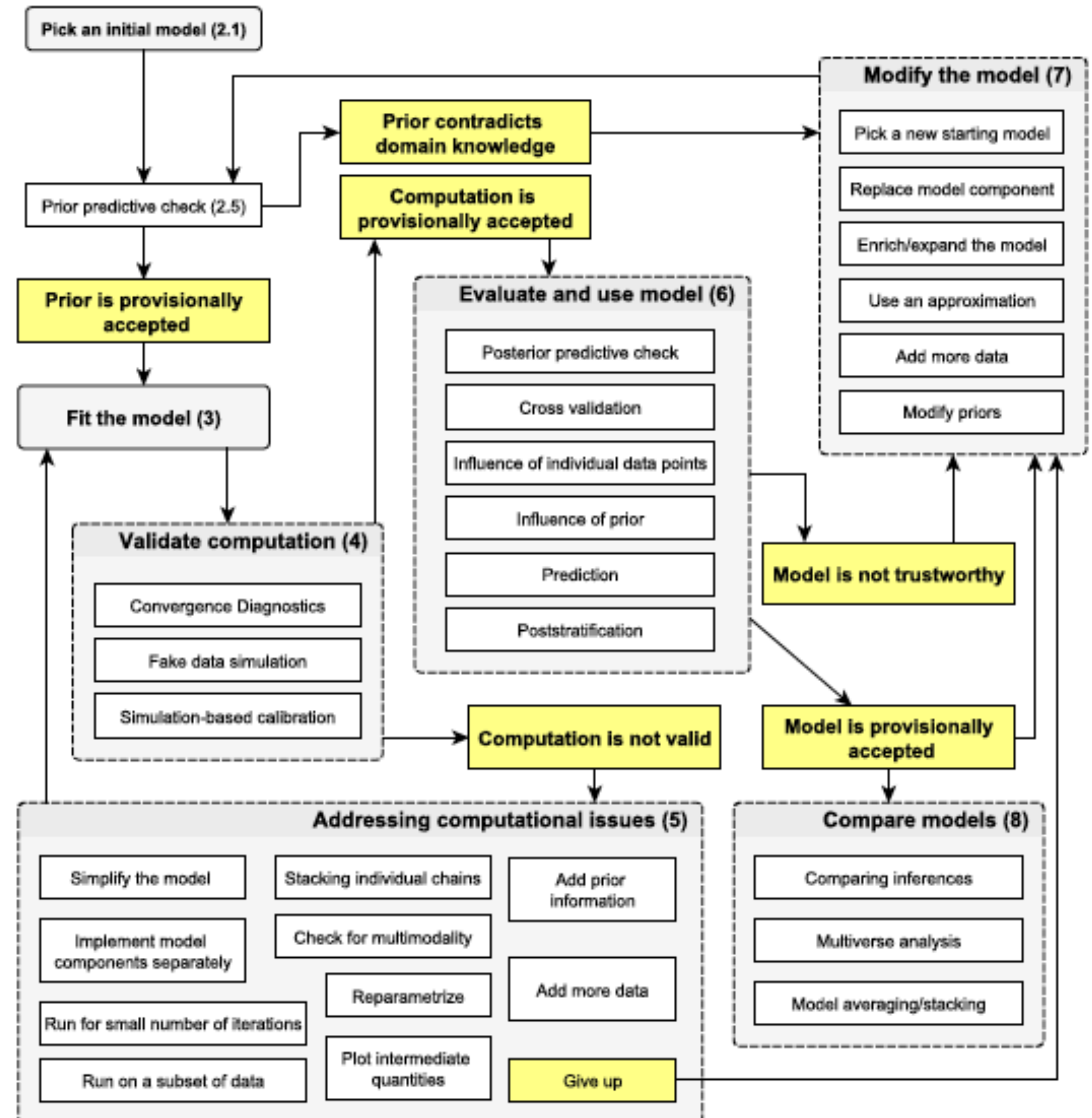
Weehaa “it works”!

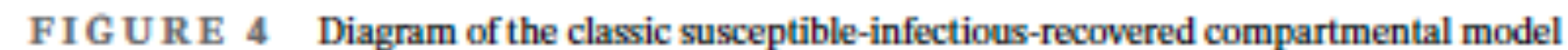
- ... right?
- MCMC diagnostics only tell you whether computation for the probabilistic model is valid, not if the computation reflects the model you intended to specify
- Consider “fake data simulation” or simulation-based calibration
 - When fixing the truth (e.g., a draw from prior)
 - Can we reproduce that truth?



Weehaa “it works”!

- But how useful is it for our context?
 - -> Evaluate and use model
- Use strategies from Module 11: model checking
 - Posterior predictive checks
 - Cross-validation
- Good enough?
 - No -> Modify the model
 - Informed by issues that were identified
 - Yes -> Model comparison
 - Compare (and consider reporting on) inferences from different models, combine models

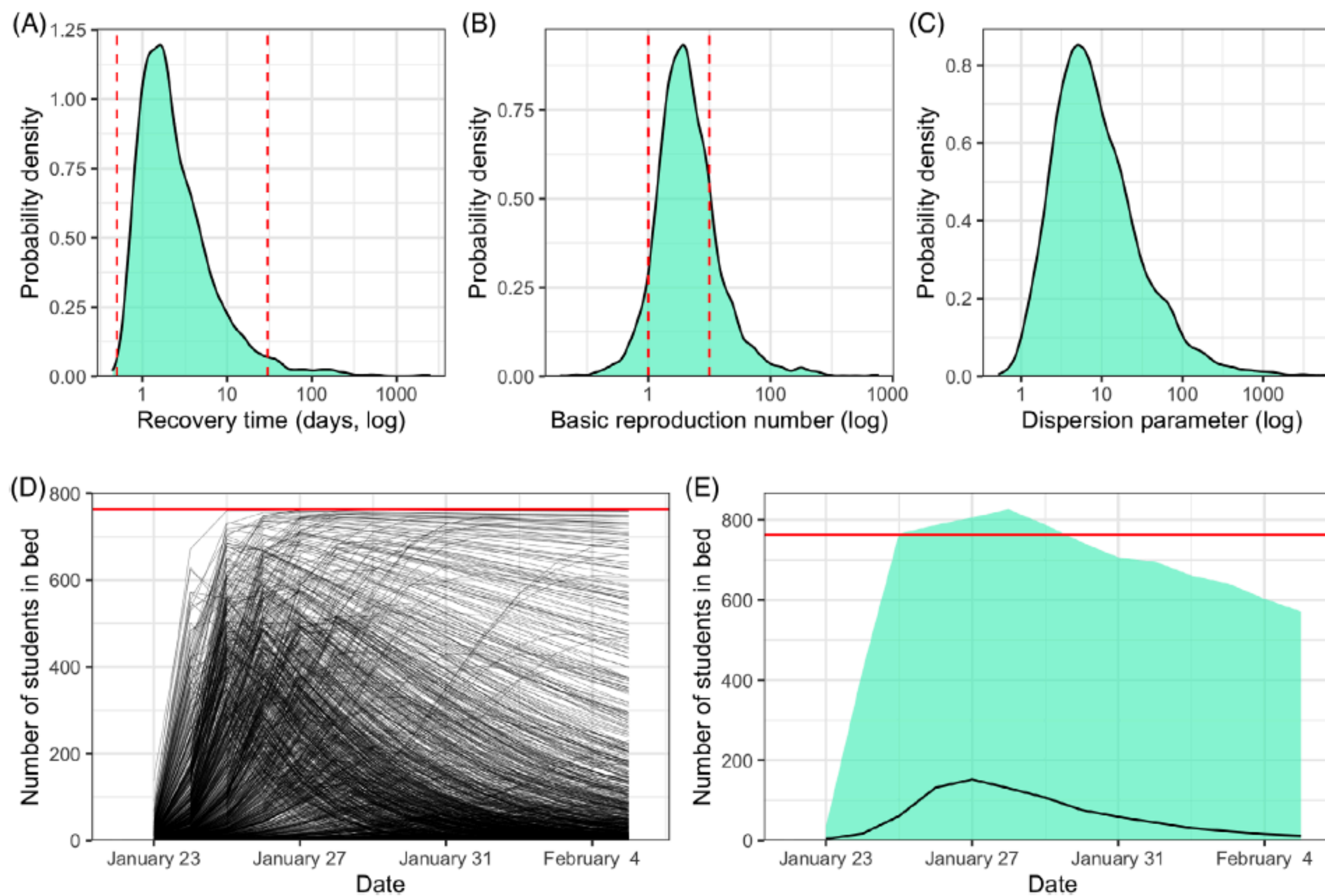




```
real[] f(real t, real[] y, real[] vartheta, real[] x_r, int[] x_i) {
    real S = y[1];
    real I = y[2];
    real R = y[3];
    real beta = vartheta[1];
    real gamma = vartheta[2];
    real N = x_i[1];
    real dS_dt = -beta * I * S / N;
    real dI_dt = beta * I * S / N - gamma * I;
    real dR_dt = gamma * I;
    return {dS_dt, dI_dt, dR_dt};
}
```

[illegible]

Prior predictive check



Computational and model checks

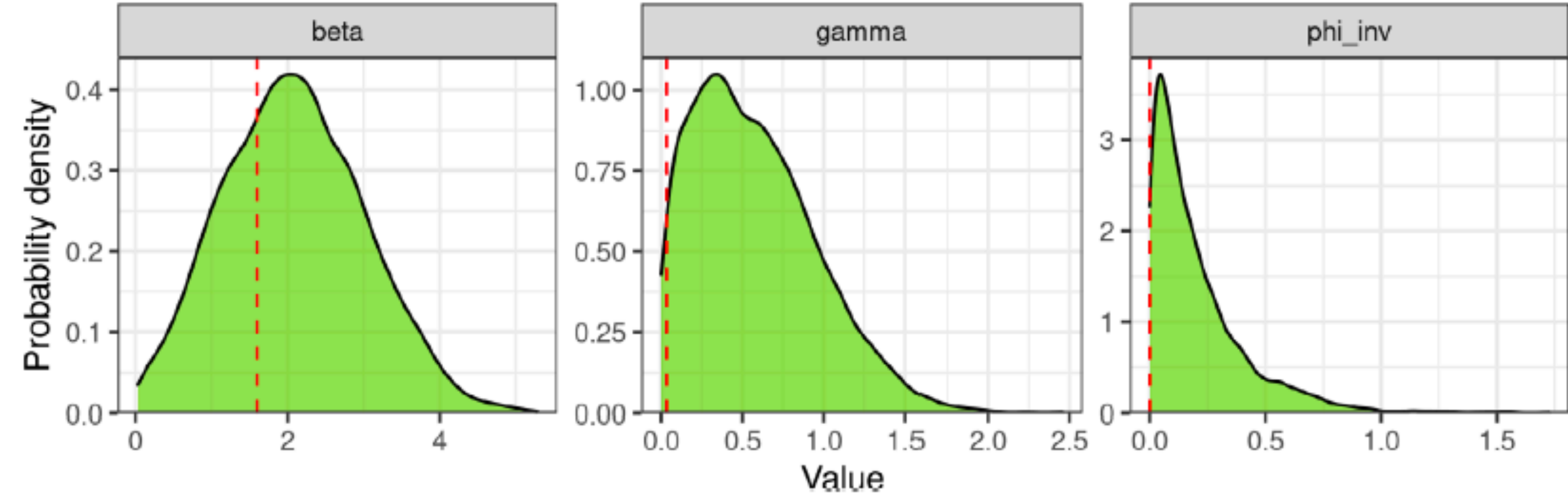


FIGURE 7 Marginal posterior densities for the transmission rate (beta or β), the recovery rate (gamma or γ) and the inverse dispersion parameter (phi_inv or $1/\phi$) obtained when fitting the model to simulated data. The red dashed lines show the fixed parameter values used for simulating the data [Colour figure can be viewed at [wileyonlinelibrary.com](#)]

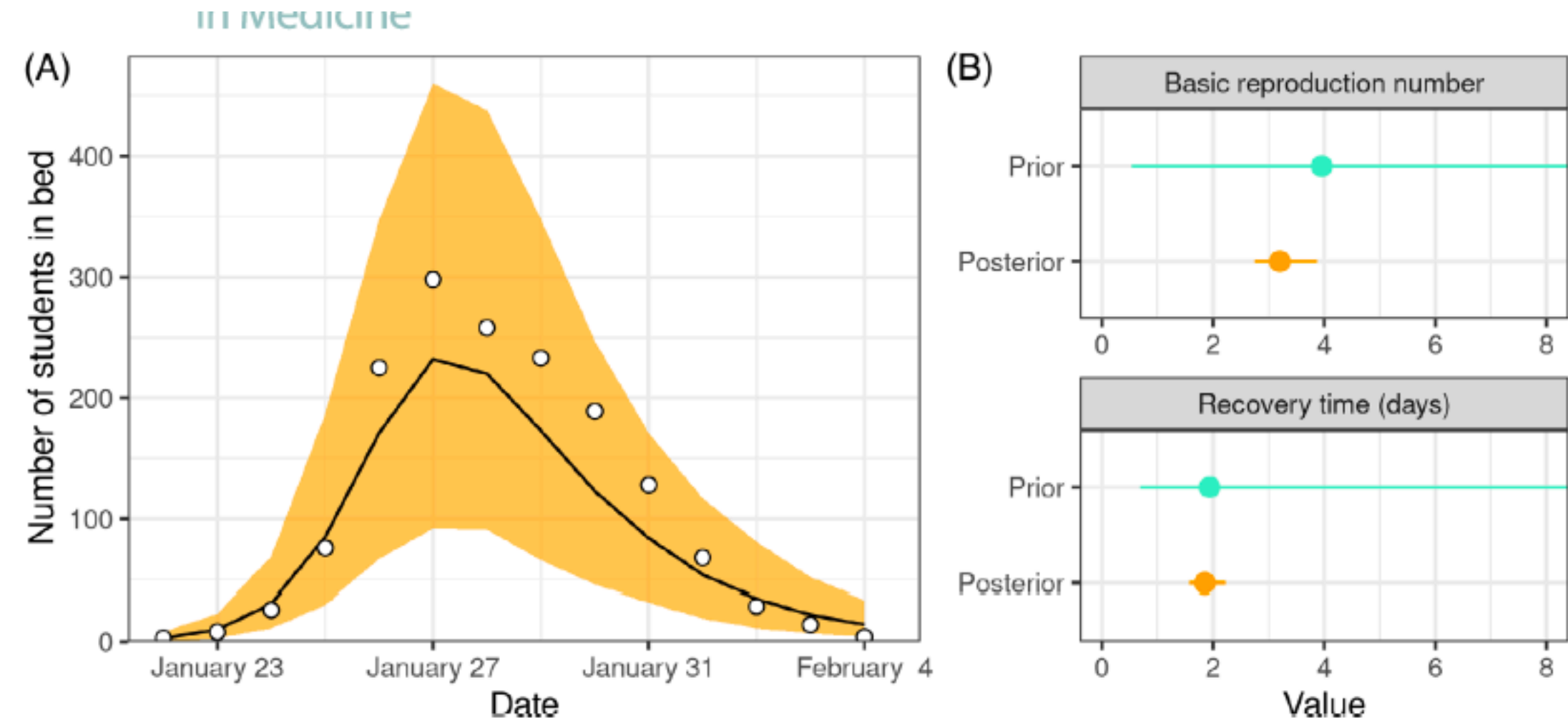
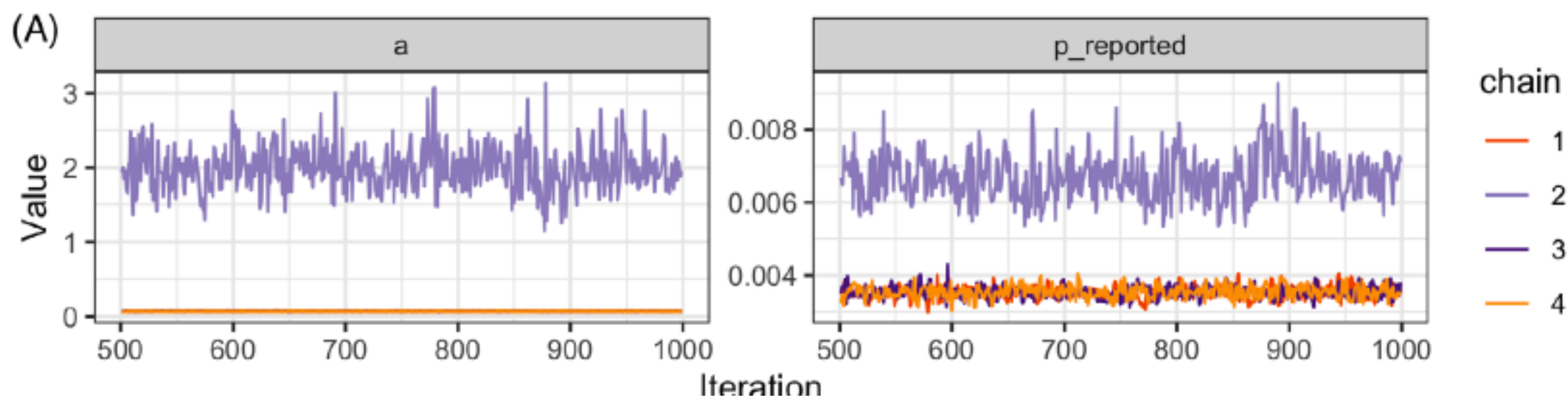
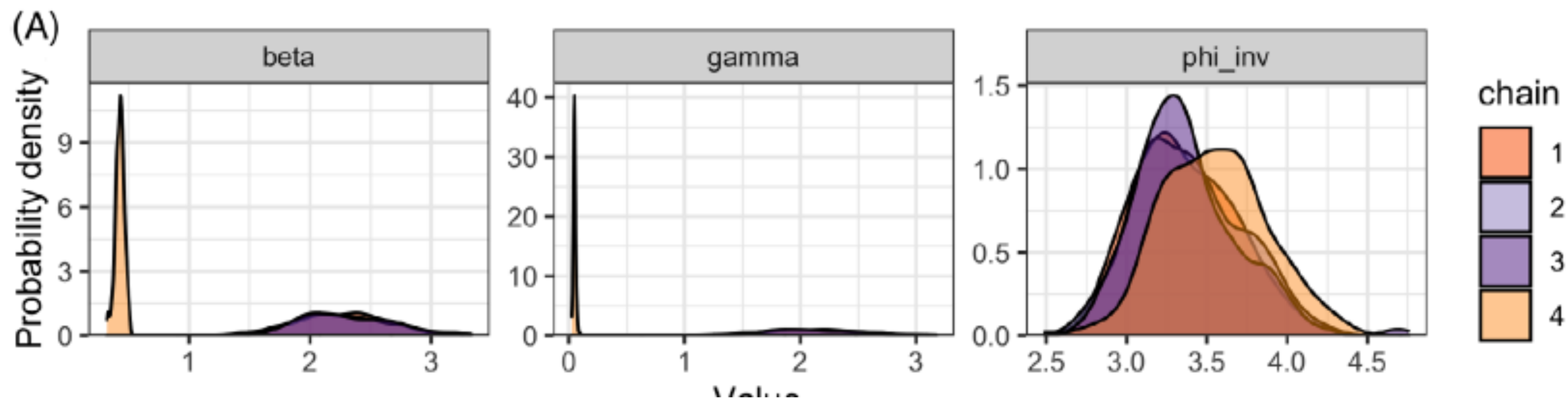


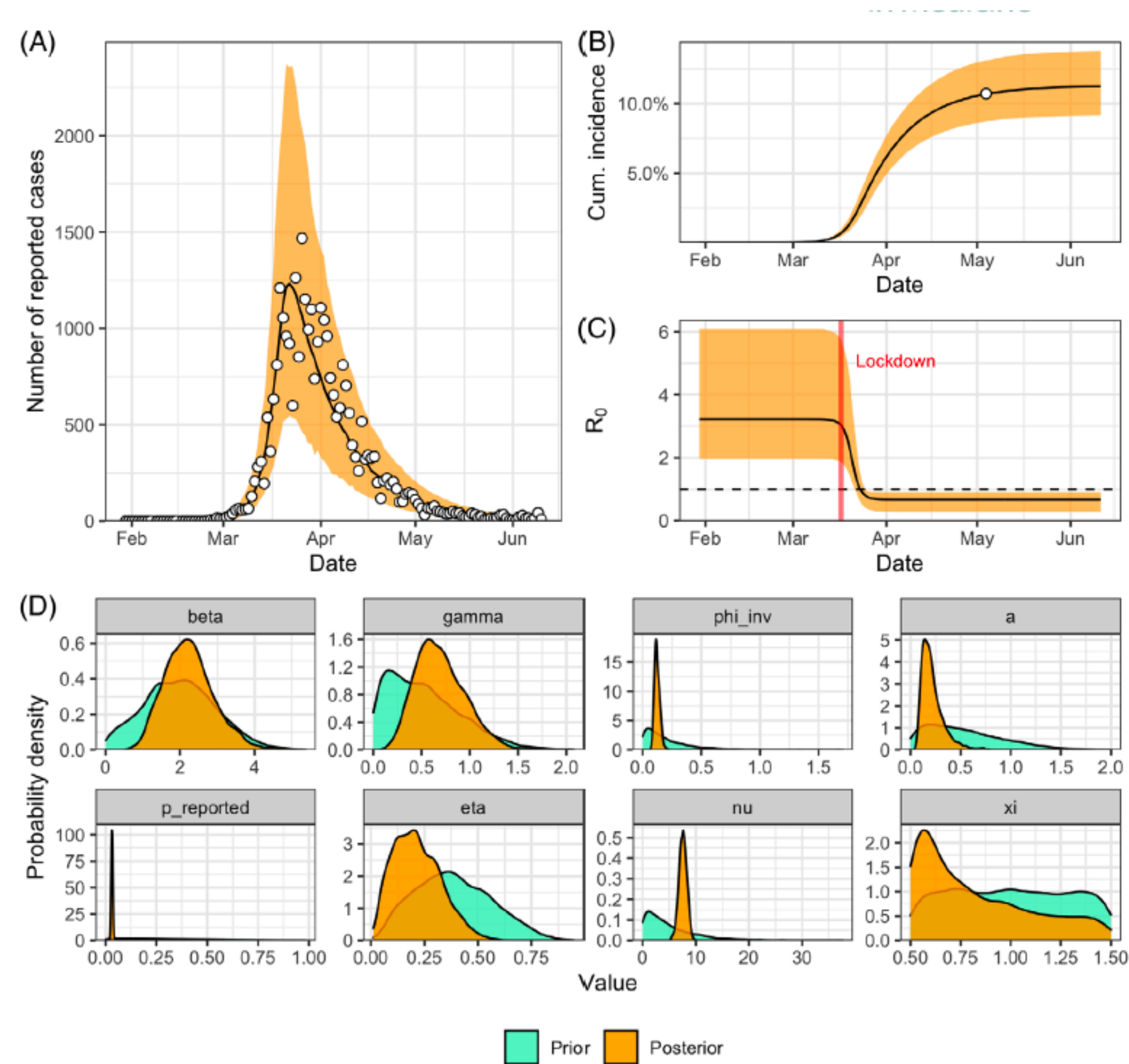
FIGURE 8 A, Posterior predictive check of the number of students in bed each day during an influenza A (H1N1) outbreak at a British boarding school. The line shows the median and the orange area the 90% prediction interval. B, Prior and posterior predictive checks of the basic reproduction number R_0 and of the recovery time (both truncated at 8). The dot shows the median posterior and the line shows the 95% credible interval [Colour figure can be viewed at [wileyonlinelibrary.com](#)]

COVID 19 application

Oops!



After extending the model and adding data...



Discussion of Bayesian workflow (Gelman et al. 2020)

- Potential issues with the Bayesian workflow:
 - Using the data multiple times, overfitting
 - Gelman et al advocate for reporting based on several models, models that encompasses as much information as possible, NOT to search for an optimally fitting model
 - Subjectivity, more of an art than a science?
- On the upside:
 - Iterative model building = gradual learning
 - The workflow allows for severe testing of model assumptions (contrast with pre-registered model)
 - This is only the start...
- Transformative steps in data science (Gelman et al):
 - Data summaries
 - Modeling
 - Computation (ongoing)
 - Towards formalizing the process of model navigation (exploring model space)