Part III Interim Progress Report

Project 39: Bouncing Hamiltonian Monte Carlo for supernovae cosmology and beyond

Harvey T-Williams (hw562)

Michaelmas 2024

Motivation

Physics is fitting models to data (and ideally making quantifiable predictions of future data). It is usually straightforward to calculate the *likelihood* of observing a set of data, assuming a given model. That is $P(D|\theta)$.

However, the quantity we really care about is the *posterior* $P(\theta|D)$. Which is a direct measure of how "good" the model is given data we have actually observed already. We can use Bayes' theorem to relate the two:

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} \tag{1}$$

We refer to $P(\theta)$ as the *prior* as it comes from any pre-existing beliefs we have about the likelihood of the candidate models. This is often taken to be uniform in lieu of any more reasoned judgment. The denominator $P(D)$ is referred to as the *evidence* and may be calculated from the likelihood and prior via the law of total probability/marginalisation:

$$P(D) = \int P(D|\theta)P(\theta)\, d\theta \tag{2}$$

Physical Example

Here we provide a sketch of such a situation. Consider that we are observing a supernova, which is of interest as a "standard candle". One measurement we might take is its brightness over time. See figure 1.

We can't make truly continuous measurement, and instead must take a discrete number of observations over time. Each of these has some uncertainty associated, in part due to stochastic processes in the supernova and in part due to error in our measuring devices. We appeal to the central limit theorem and assume that the total uncertainty σ_t on each measurement follows a Gaussian distribution. For simplicity and depending on our goals, we might also assume that our knowledge of the physics of supernovae is correct and that

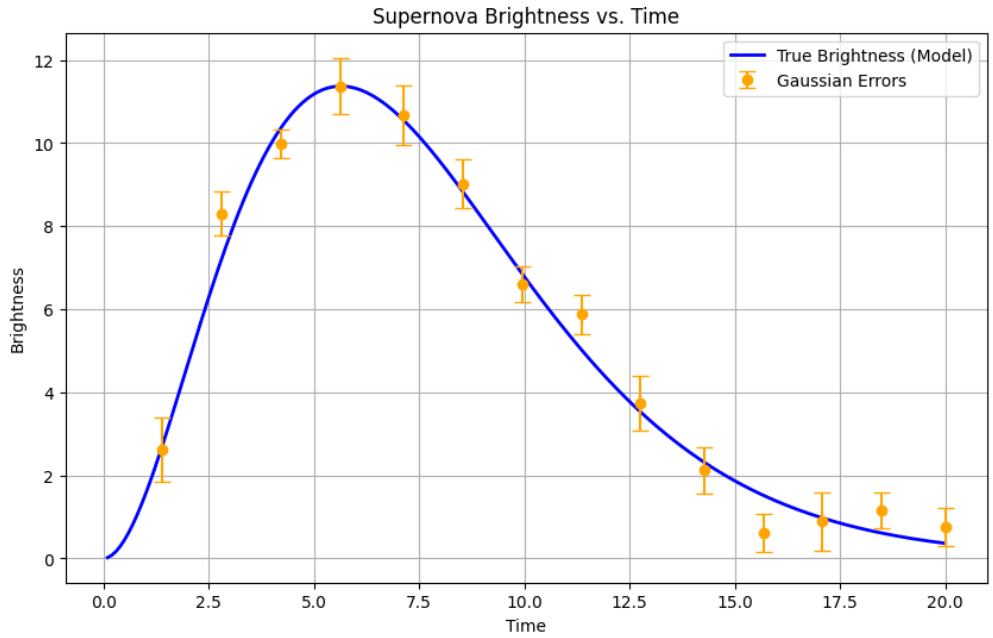

Figure 1: A rough sketch of supernova brightness over time, with errors. Data is fictitious.

the unknowns are simply some set of parameters (e.g. progenitor mass, temperature, etc.) which we shall label with the vector $\boldsymbol{\theta}$ (with one component for each parameter).

Thus we have a set of observational data D_t for the brightness at different times t and can calculate expected brightness $I_t(\boldsymbol{\theta}_i)$ as a function of parameters for each time t. Making the Gaussian approximation (valid by central limit theorem) we can calculate the likelihood as

$$P(D|\theta) = \text{normalisation} \times \prod_t \exp\left(\frac{(D_t - I_t(\boldsymbol{\theta}))^2}{2\sigma_t^2}\right) \tag{3}$$

We can then plug this into Bayes' theorem to calculate the posterior and judge how probable any set of parameters might be, given the data we have observed. If we calculate the evidence via marginalisation, then we do not need to know the normalisation factor as it will cancel.

The "obvious" thing to do would be to incrementally adjust the parameters (via gradient descent) until we find a maximum in the posterior and declare that this is our best guess for the properties of the supernova progenitor. In this case, we would not need to evaluate the normalisation nor the evidence — which is good because this is very difficult!

The drawback of the gradient descent approach is that it does not give us any information about the uncertainty on our parameter estimation (it might also fail to hit the global maximum). This is where sampling becomes advantageous. In general $I(\boldsymbol{\theta})$ will be some complicated function that does not have an analytic integral—so we cannot directly calculate the evidence nor higher order moments such as variance.

If we can obtain a (large) sample of points from our probability distribution, then we can appeal to the law of large numbers and approximate any property of the distribution. E.g., as the number of samples becomes large, the sample mean and variance converge to

the mean and variance of the underlying distribution.

Markov Chain Monte Carlo

If we want to sample from a known probability distribution $P(\boldsymbol{\theta})$ then a naive approach would be to form a uniform grid of points and "accept" them or "reject" them according to the probability distribution. The computational cost of doing this scales exponentially with the number of parameters in $\boldsymbol{\theta}$ — and this is usually quite a high number for physical applications.

A common means to avoid this "curse of dimensionality" is to perform a random walk that will reflect the underlying distribution. One well-studied form of random walk is the Markov process, where each step X_{t+1} depends only on the previous step X_t. Formally this is written as

$$P(X_{t+1}|X_t) = P(X_{r+1}|X_t, X_{t-1}, \ldots, X_0) \tag{4}$$

Markov processes are characterised by their transition matrix Q_{ij}, which defines the probability that $X_{t+1} =$ state j given $X_t =$ state i [1]. A given transition matrix might have an associated *stationary distribution(s)* $\mathbf{s}$ such that

$$\sum_i \mathbf{s}_i = 1 \quad \text{and} \quad \mathbf{s}Q = \mathbf{s} \tag{5}$$

That is to say $\mathbf{s}$ is a left-eigenvalue of the transition matrix whose entries represent probabilities of being in each state. Notably if X_0 is distributed according to a stationary distribution, all following X_t will be distributed according to the stationary distribution.

Existence and uniqueness I

In finite state space, any irreducible (meaning all states always have some probability of being visited eventually) Markov chain has a unique stationary distribution. Moreover, if the chain is aperiodic, X_t will eventually converge to this distribution, regardless of X_0 [1]. This motivates the continuous case but we really ought to find a corresponding theorem.

Existence and uniqueness II

An alternative condition is detailed balance. Namely if we can find a distribution $\mathbf{s}$ (with $\sum_i \mathbf{s}_i = 1$) such that

$$s_i q_{ij} = s_j q_{ji} \quad \text{for all } i, j \quad \text{(no summation)} \tag{6}$$

then this is a (not necessarily unique) stationary distribution.

Metropolis-Hastings

The goal of Markov Chain Monte Carlo is to engineer the transition matrix Q such that it has a stationary distribution matching the PDF we want to sample from. The basic idea is captured in the Metropolis-Hastings algorithm:

1. Start at position $\theta_t =$ state i.
2. Propose moving to a state j, according to some (unimportant) proposal probability matrix P_{ij}.
3. Calculate the acceptance probability
$$a_{ij} = \min\left(\frac{\mathbf{S}_j P_{ji}}{\mathbf{S}_i P_{ij}}, 1\right),$$
where $\mathbf{S}_n$ is the probability of state n according to the desired underlying distribution
4. Set $\theta_{t+1} =$ state j with probability a_{ij}, otherwise set $\theta_{t+1} =$ state i.
5. Repeat

Here the acceptance probability a_{ij} acts as the transition matrix Q_{ij} we desire. It is fairly straightforward to show that this Q_{ij} obeys detailed balance where $\mathbf{s}$ is the underlying distribution (see Hwang Chapter 12). Thus we have constructed a transition matrix such that its stationary distribution is the one we want to sample from.

The beauty of this algorithm is that it is mathematically agnostic to our choice of proposal matrix P. In a vanilla implementation of Metropolis-Hastings, we often take it to be a multivariate-normal which is symmetric and therefore cancels when calculating the acceptance probability. We can be sure that the denominator P_{ij} is never 0 as the algorithm would never propose such a step. However, in the proof of detailed balance, we also rely on P_{ji} never being 0. This is not easy to guarantee and we will see it have a strong influence on more sophisticated algorithms.

Hamiltonian Monte Carlo (choosing a sensible P)

The detailed-balance property of the propose-accept method guarantees that our Markov Chains will eventually converge to the desired distribution. However, our choice of the proposal matrix P influences how quickly this will happen. Basic Metropolis-Hastings using a multivariate-normal proposal probability performs a Brownian random walk on the state-space and moves a distance proportional to $\sqrt{N}$ with N the number of steps.

One improvement on this is the Hamiltonian Monte Carlo (HMC) method. This makes use of Hamiltonian dynamics as follows:

1. Start at "position" $\boldsymbol{\theta}_t =$ state i.
2. Randomly draw a "momentum" $\mathbf{p}$ from a 0-centered Gaussian with covariance matrix M (usually equal to identity)
3. Compute the "Hamiltonian"
$$H = \frac{1}{2}\mathbf{p}M^{-1}\mathbf{p} - \log(P(\boldsymbol{\theta}))$$
and integrate its motion according to Hamiltonian dynamics. This is done using Størmer–Verlet integration with step size ϵ and number of steps L.

4. Calculate the acceptance probability a_{ij} as

$$\min\left(\exp(H(\text{previous step}) - H(\text{proposed step}), 1\right).$$

5. Set $\theta_{t+1} =$ state j with probability a_{ij}, otherwise set $\theta_{t+1} =$ state i.

6. Repeat

Note that the acceptance probability in HMC is exactly the same as in the Metropolis-Hastings algorithm: when one works through everything the logs, exponentials and inner-products give us $a_{ij} = \min(\frac{S_j P_{ji}}{S_i P_{ij}}, 1)$ as before. The difference now is that our step size is typically much larger and we no longer move diffusively. Conservation of the Hamiltonian during motion means that, provided our momentum variance is low, we expect a high acceptance rate for proposed steps.

The HMC method has two major drawbacks: practitioners must manually choose the "hyperparameters" ϵ and L and there might be "wasted effort" by integrating paths of motion that loop back on themselves. The second of these issues is addressed by the No-U-Turn sampler.

The No-U-Turn Sampler

The goal of the No-U-Turn sampler is to avoid U-Turns. Duh! This is defined by stopping the integration of motion when the distance between the two ends starts to decrease. In practice this is calculated by checking the sign of the dot product between the current momentum and the end-to-end displacement vector. Supposedly [2] this breaks detailed balance — presumably by causing situations where $P_{ji} = 0$ and $P_{ij} \neq 0$. The solution is to integrate the motion with respect to time in both directions, alternating direction with a 50% chance of moving forward in time from the most advanced end and a 50% chance of moving backward in time from the least advanced (in time) end. Between each potential change of direction, the integrator proceeds for twice as many integration steps building a balanced binary-tree. When certain stopping conditions are met, we stop the integration and are left with a binary tree of candidate sample points as leaf-nodes. For any of these candidates, the tree we have is one of 2^k possible unique trees that we could have formed if we had started at that node (where k is the depth of the tree). To see this, note that any possible tree formed from a starting node is uniquely characterized by the k forward or backward directions we integrate. From this we can form a binary number using 1 for forward and 0 for backward. Therefore each tree has probability $1/(2^k)$ of occurring given we started at one of its leaf nodes. All we need to do to ensure detailed balance is to exclude any leaf-nodes that would have resulted in earlier termination had we started there. This is kept track of while the tree is built. Finally we draw a proposed step uniformly from these leaf nodes and calculate the acceptance probability as usual.

What about the bouncing?

Some regions of parameter space can be excluded from our search (they have 0 probability) for physical or symmetry or observational reasons. Traditionally this is included into HMC

using "soft" barriers which smoothly decrease the probability to 0 in the excluded regions. An alternative is for our Markov Chain to "bounce" off these regions analogously to a particle bouncing off a hard barrier. Because of the finite integration size, there is 0 probability of the chain landing exactly on the boundary. Instead, it will get sufficiently close that the next proposal lies beyond the boundary. Therefore we calculate reflection according to the nearest surface-normal.

If we label the forward direction as North and the reverse direction as South, and label reflection in the usual sense as one of East or West then reflection in the perpendicular plane will be West/East respectively. Supposing a forward (North) proposal would be in an excluded region, we hope [3] (but have not yet proved) that at most **one** of East or West will be a valid proposal direction. Otherwise a chain passing through the same point in the East-West direction would never be diverted to the North-South direction and this would clearly break detailed balance. If possible, we move East/West when North is excluded. However, it might be that neither East nor West is a valid proposal direction, in which case we propose a Southward step (doubling back on ourselves).

Work done so far

See the below images and captions for a discussion of work completed so far implementing a HMC sampler.

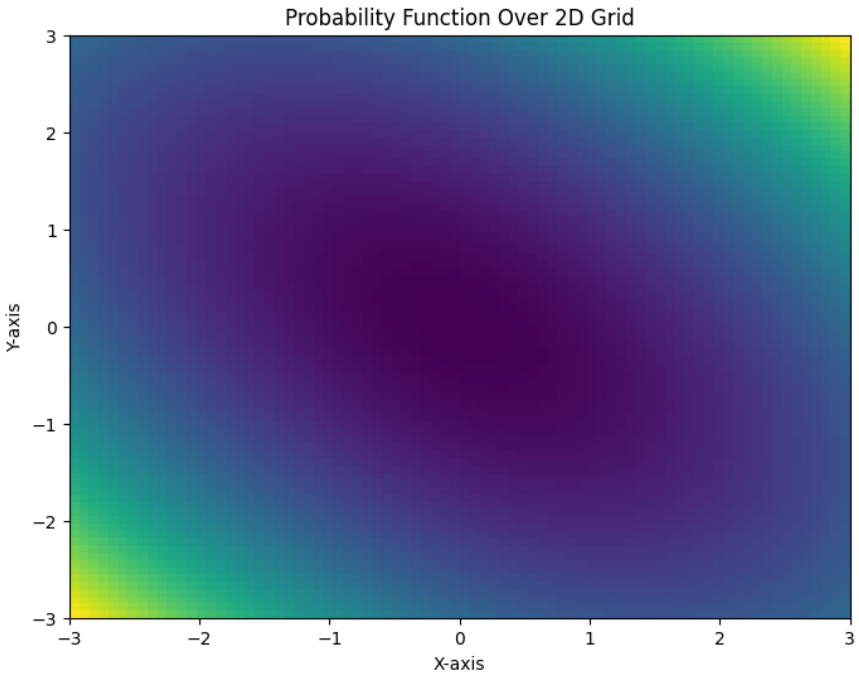

Figure 2: Here we see the underlying distribution we want to sample from. It is a 2D Gaussian with variances 1 and covariance 0.5.

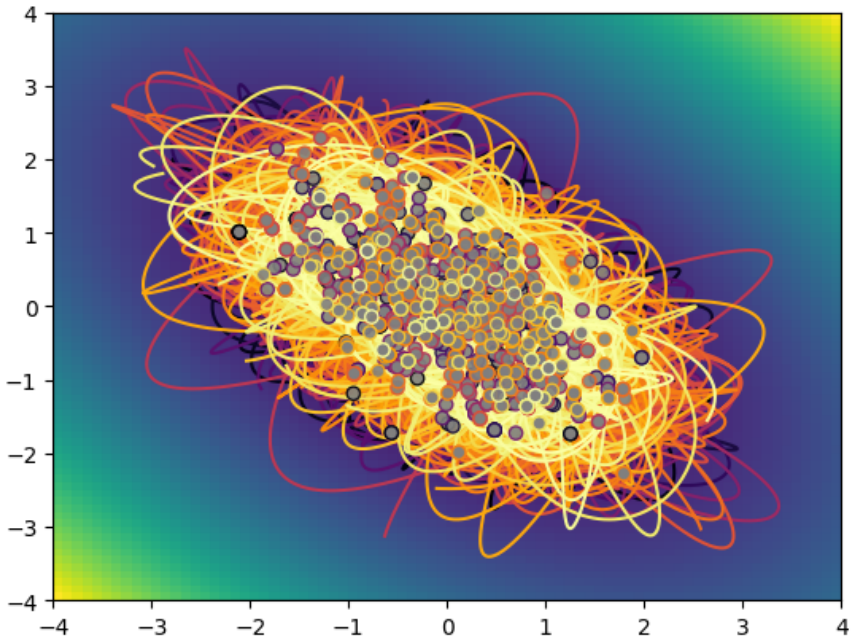

Figure 3: Here we see the result of HMC sampling generating 1000 samples. The circles are our sample points and the curves are the Hamiltonian trajectories followed in the proposal step. Darker colours represent the early steps in the chain and lighter colours later steps. We can see that all colours are spread across a wide range of parameters.

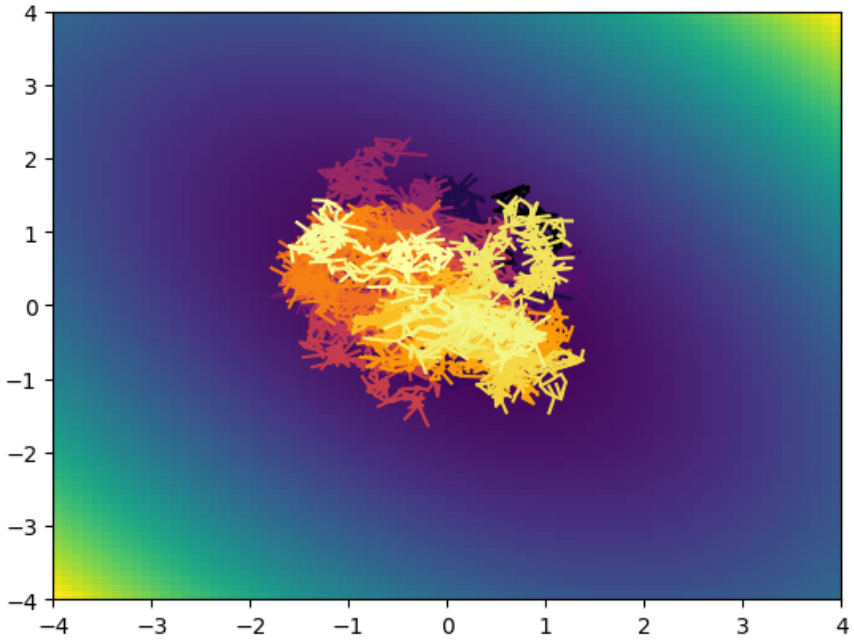

Figure 4: Here we see the limiting case as integration length L is reduced. We effectively have standard Metropolis-Hastings. Notice that colours are clumped as the chain slowly diffuses.

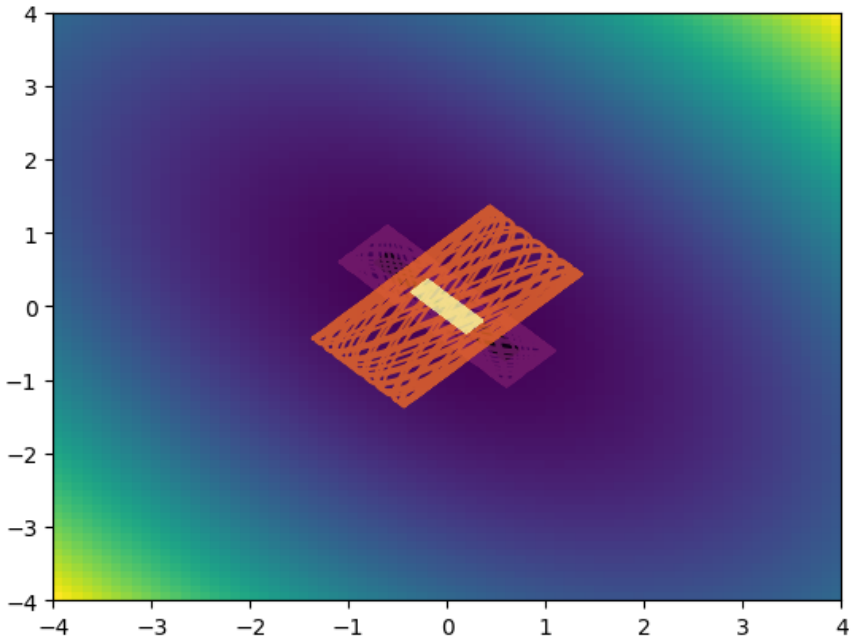

Figure 5: Here we see the limiting case as integration length L is increased. Notice the doubling-back. We might naively expect elliptical outlines but we are reminded of Lissajous figures which arise from independent oscillators.

Next steps

1. Implement bouncing
2. Implement No U-Turn
3. Find continuous version of detailed balance theorem
4. Derive why a naive version of the No-U-Turn breaks detailed balance
5. Mathematically confirm that we expect to see Lissajous figures for large L
6. Implement an adaptive method to tune ϵ
7. Test more-physical examples — e.g. masked supernovae problems, which induce sharp cutoff regions in parameter fits

References

[1] Joseph K. Blitzstein and Jessica Hwang. *Introduction to Probability*. Chapman and Hall/CRC, Feb. 2019. ISBN: 9780429428357. DOI: 10.1201/9780429428357. URL: http://dx.doi.org/10.1201/9780429428357.

[2] Matthew D. Hoffman and Andrew Gelman. "The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo". In: *Journal of Machine Learning Research* 15.47 (2014), pp. 1593–1623. URL: http://jmlr.org/papers/v15/hoffman14a.html.

[3] John Skilling. "Galilean and Hamiltonian Monte Carlo". In: *The 39th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering*. MaxEnt 2019. MDPI, Dec. 2019, p. 19. DOI: 10.3390/proceedings2019033019. URL: http://dx.doi.org/10.3390/proceedings2019033019.