Hamiltonian Monte Carlo

Thomas Hamelryck
PLTC section
PML 2022

In a nutshell

- Hamiltonian Monte Carlo / NUTS is an efficient way to approximate the **Bayesian posterior** by **a set of samples**.

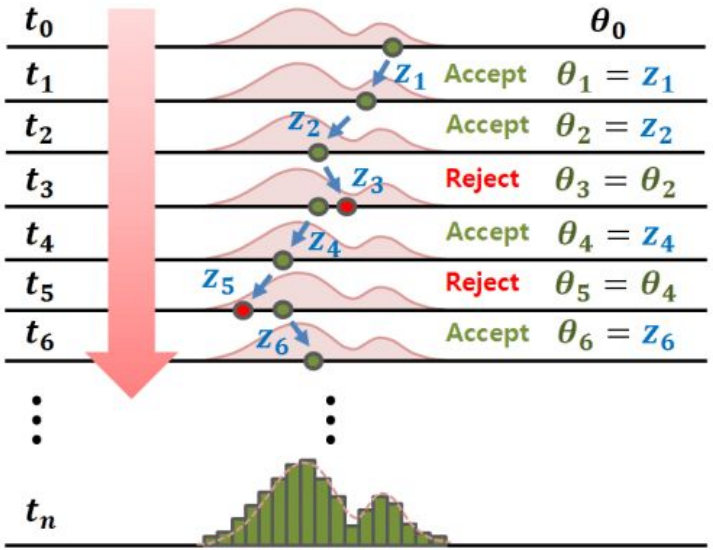


Figure reference

Overview

- Limitations of standard MCMC
  - Poor exploration of the **typical set**
- Hamiltonian equations of motion
  - Parameters are coordinates of a particle moving in a force field
    - Kinetic energy (momentum)
    - Potential energy (-log posterior)
  - Symplectic integrators
- This is used as a superior proposal in HMC
  - NUTS: automated HMC
- Bonus topic: Diagnostic of convergence

Reading material

- A conceptual introduction to Hamiltonian Monte Carlo
  - Betancourt, 2017
- Rank-normalization, folding, and localization: An improved R-hat for assessing convergence of MCMC
  - Vehtari, Gelman, Simpson, Carpenter & Bürkner, 2020
  - Diagnostics of MCMC convergence
- HMC/NUTS in Pyro
- Diagnostics from Arviz

Monte Carlo & Bayes

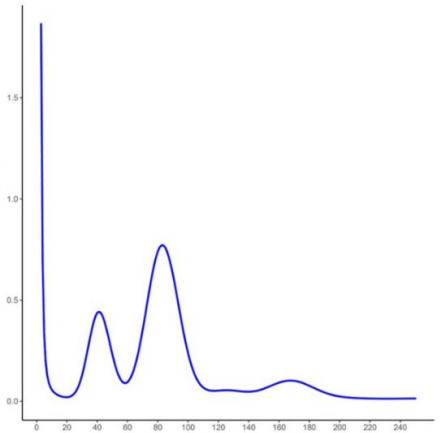
- The Bayesian posterior is often unavailable as a closed-form expression.
- Monte Carlo methods approximate the posterior using samples.
  - Fast computers made this approach mainstream.
- The core idea is simple: approximate an expectation using samples.

$$\mathbb{E}\left[f(x)\right] = \int f(x)p(x)\mathrm{d}x \approx \frac{1}{S}\sum_{s=1}^{S} f(x_s)$$

Parsimonious expectation computation

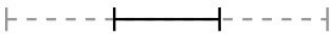
- Exploring regions of parameter space that have negligible contributions to the expectation is inefficient.
- Intuition: concentrate on regions where f(.) and the density peak (i.e. >>0)
  - To keep things general, focus is typically on the density
- Naive, flawed approach: focus on neighborhood of mode

$$\mathbb{E}\left[f(x)\right] = \int f(x)p(x)\mathrm{d}x \approx \frac{1}{S}\sum_{s=1}^{S} f(x_s)$$

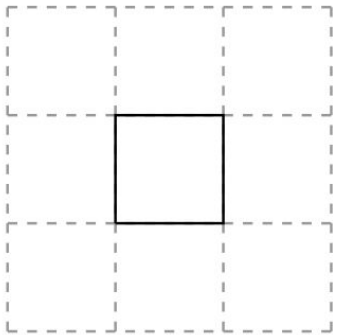


Geometry of high dimensional spaces

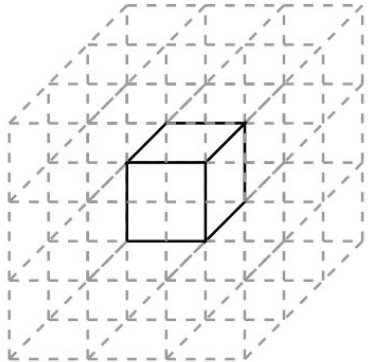
- It gets worse with increasing dimensionality!
- For increasing D, the volume of neighboring volume elements dominates the volume of the element containing the mode
  - 3^D-1 neighboring volume elements



1/3



1/9

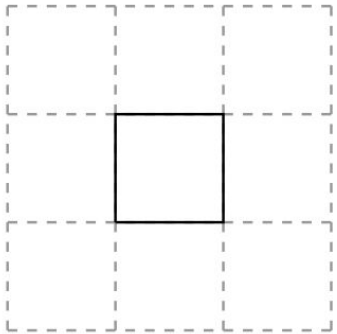


1/27

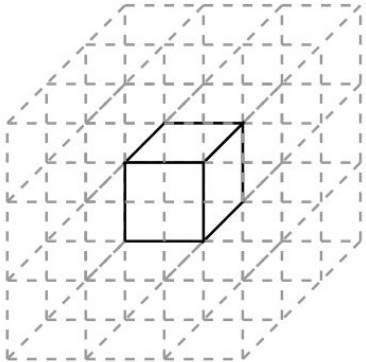
Geometry of high dimensional spaces

- We need to consider both density AND volume with increasing D
  - Intuition: Massive volume can compensate for low density

1/3



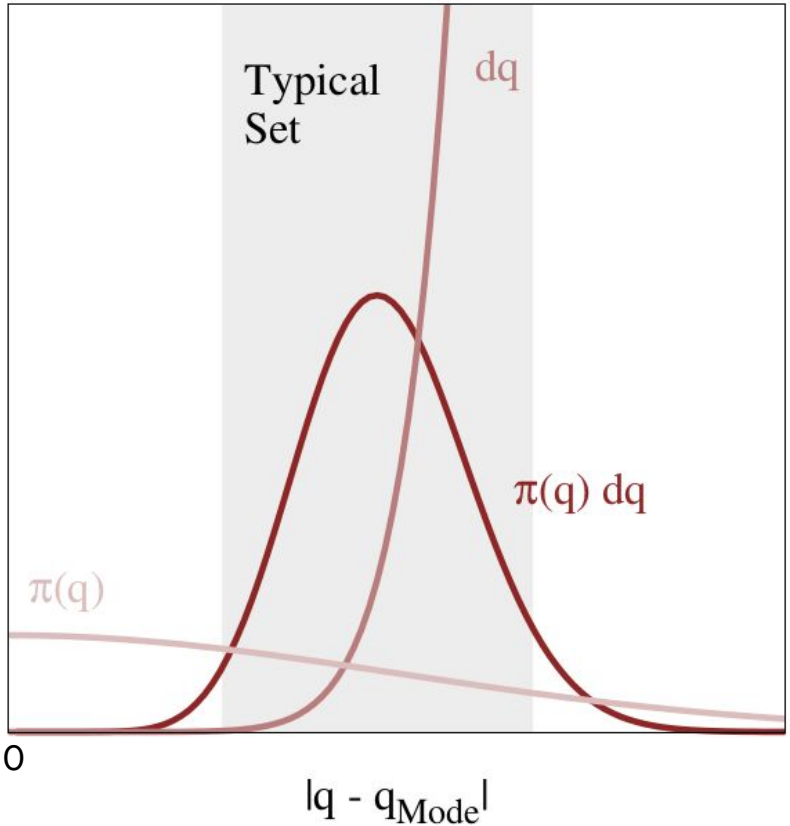
1/9



1/27

Typical set

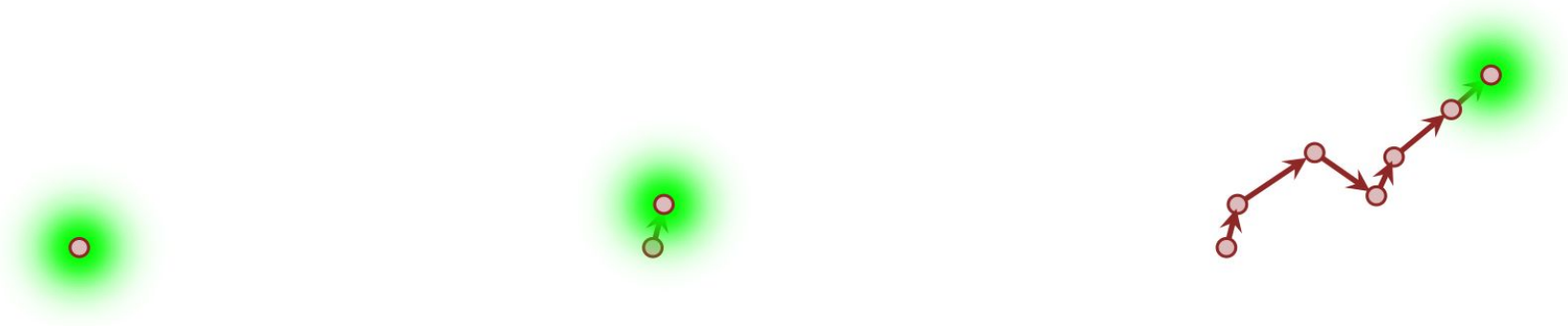
- Insignificant contributions
  - High density but no volume
  - High volume but no density
- Significant contributions only come from the **typical set**
  - In between the above extremes
  - Invariant under transformation
  - Becomes more narrow with D
- This is why brute-force methods involving **grids** do so poorly



Markov chain Monte Carlo

- Generate samples by jumping from point q to point q' using a **Markov transition T(q|q')** that preserves the target distribution (ie. the posterior).
  - The chain will move towards the typical set

$$\pi(q) = \int_{\mathcal{Q}} \mathrm{d}q' \, \pi(q') \, \mathbb{T}(q \mid q')$$



MCMC estimators

- **MCMC estimators** will eventually explore the typical set and converge to the true expectation.

$$\hat{f}_N = \frac{1}{N} \sum_{n=0}^{N} f(q_n)$$

$$\lim_{N \to \infty} \hat{f}_N = \mathbb{E}_\pi[f]$$

MCMC estimators

- MCMC Central Limit Theorem

$$\hat{f}_N^{\text{MCMC}} \sim \mathcal{N}(\mathbb{E}_\pi[f], \text{MCMC-SE})$$

- The MCMC estimates will be normally distributed, with mean equal to the **true expectation** and standard deviation equal to the **MCMC standard error (MCMC-SE)**
  - For the calculation of the MCMC-SE, we need to take into account that our **N** samples might be **highly correlated**.

MCMC estimators

- MCMC Central Limit Theorem

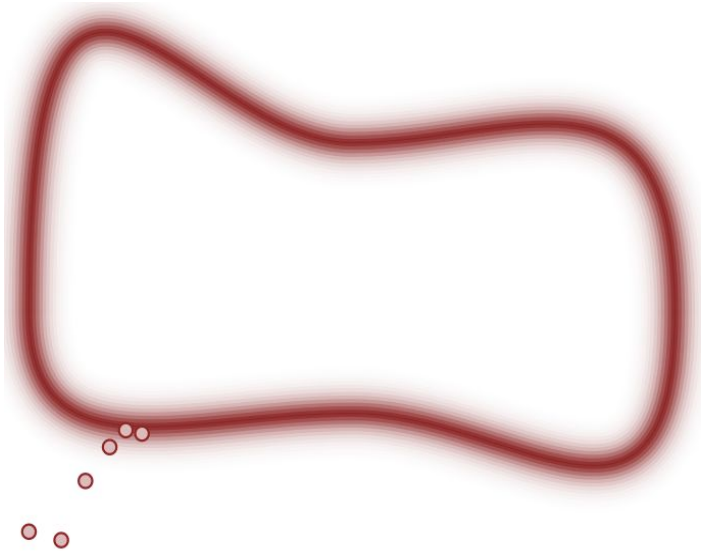
$$\hat{f}_N^{\text{MCMC}} \sim \mathcal{N}(\mathbb{E}_\pi[f], \text{MCMC-SE})$$

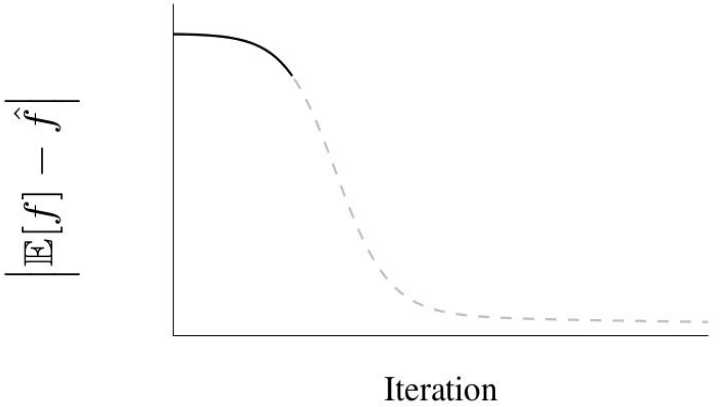
- **MCMC standard error (SE)** and **Effective Sample Size (ESS)**
  - **ESS < N** ≃ effective number of samples / number of sojourns over typical set
  - ρ_l=Lag-l autocorrelation ≃ how correlated are our samples?

$$\text{MCMC-SE} \equiv \sqrt{\frac{\text{Var}_\pi[f]}{\text{ESS}}} \qquad \text{ESS} = \frac{N}{1 + 2\sum_{l=1}^{\infty} \rho_l}$$

Three MCMC phases

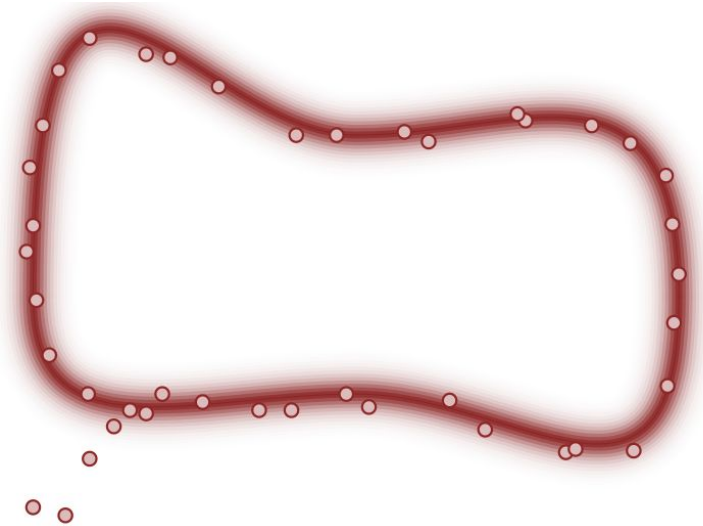
- Phase 1: convergence towards typical set
  - Strong bias in MCMC estimators (warm up)

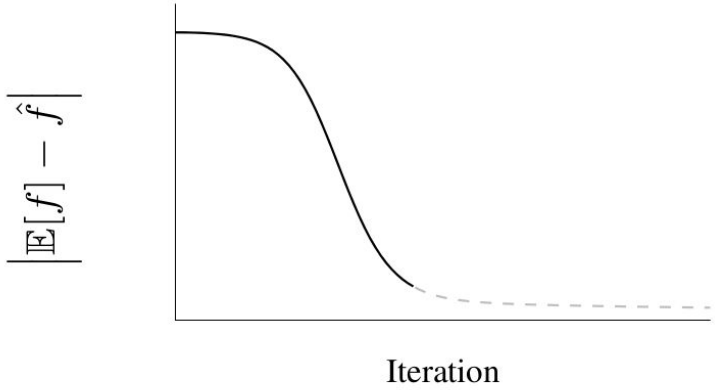




Three MCMC phases

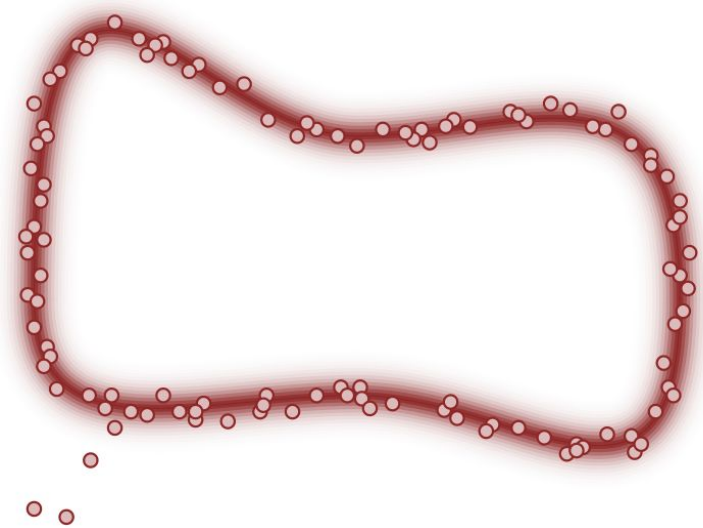
- Phase 2: first sojourn across typical set
  - Accuracy of MCMC estimators rapidly increases

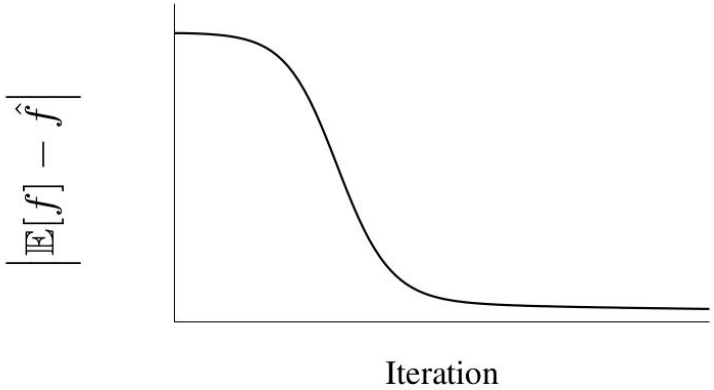




Three MCMC phases

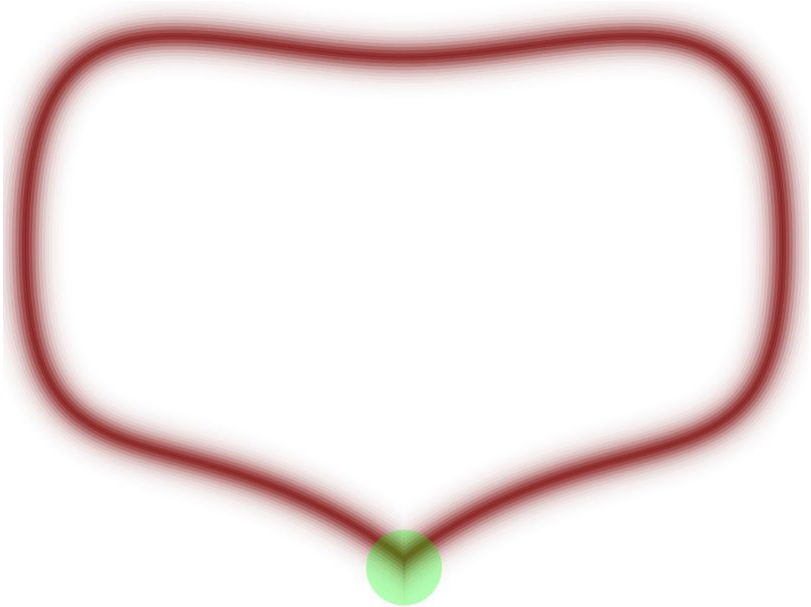
- Phase 3: subsequent exploration of the typical set
  - Accuracy of MCMC estimators increases at slower rate





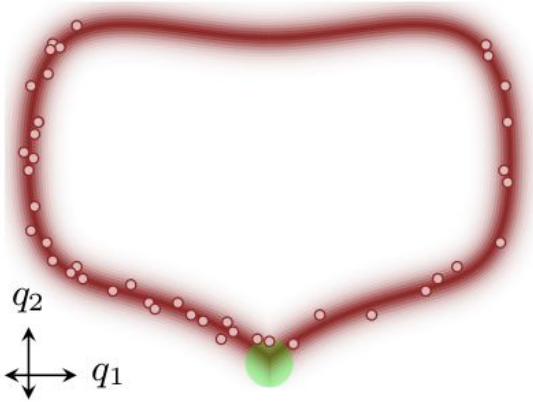
Pathological behaviour

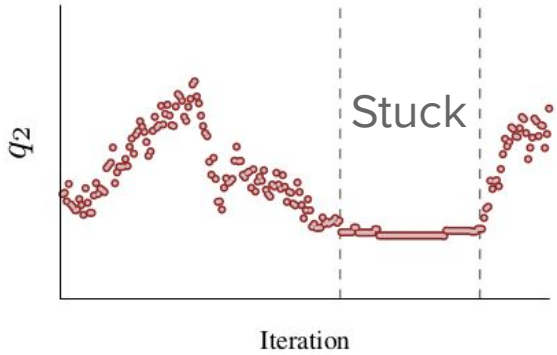
- Typical set pinches into a region of high curvature.

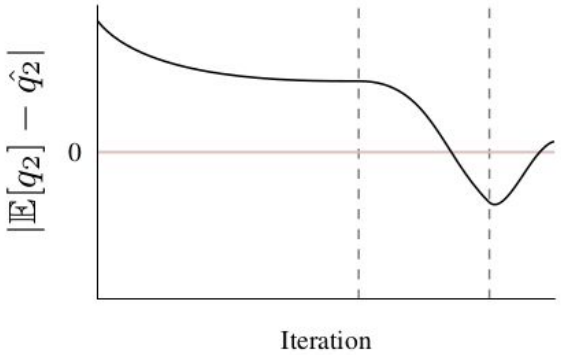


Pathological behaviour

- MCMC will get stuck at the pinch (green) and escape now and then
  - MCMC estimators will oscillate around true value

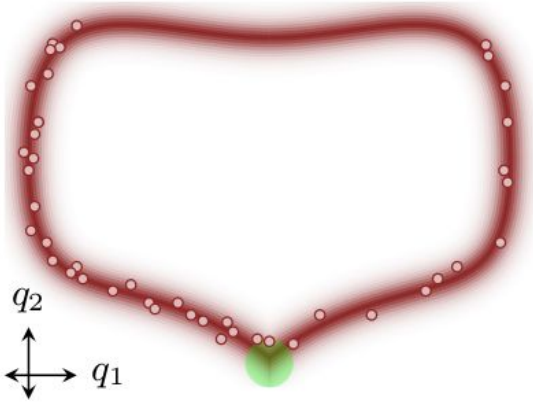


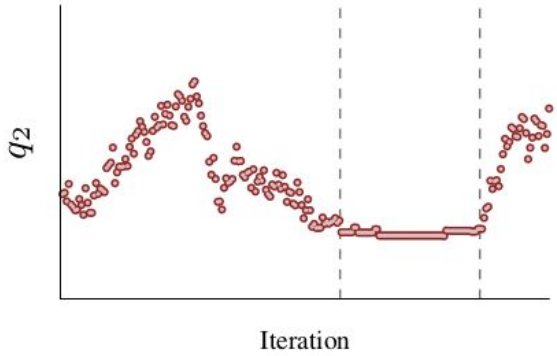


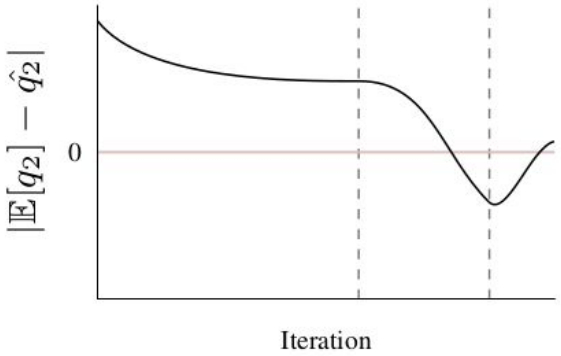


Pathological behaviour

- We need **geometric ergodicity**, but this is difficult to diagnose
  - Diagnostics such as **split-R-hat**, **tail-ESS**, **rank plots**,...
  - Diagnostics can always be fooled!







Metropolis-Hastings

- **Proposal step**: stochastic perturbation of previous point
  - Often uses a Gaussian distribution, which is symmetric
- **Correction step**: rejection of proposals far away from typical set
  - *a*=probability of accepting new proposal *q'*

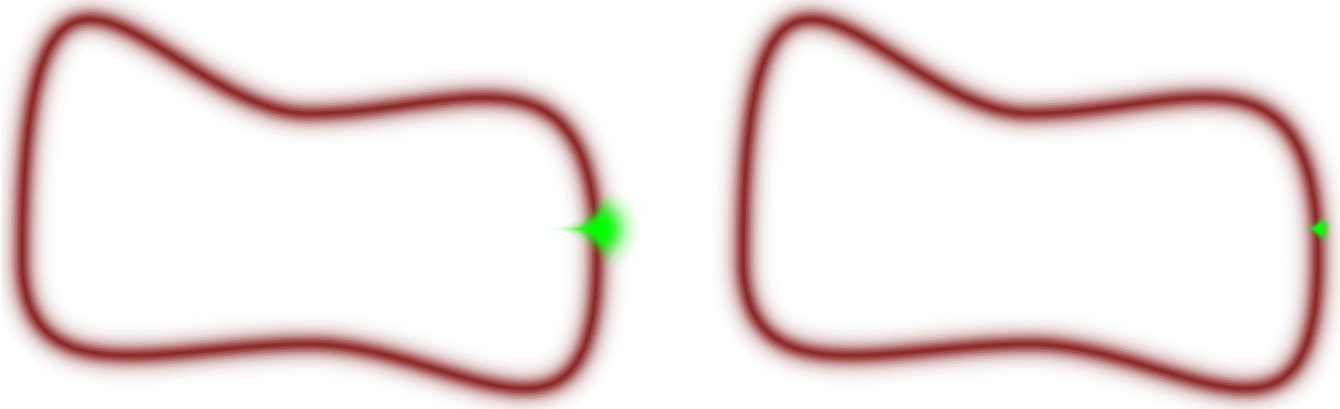
$$a(q' \mid q) = \min\left(1, \frac{\mathbb{Q}(q \mid q')\,\pi(q')}{\mathbb{Q}(q' \mid q)\,\pi(q)}\right)$$

- Gaussian case (Metropolis):

$$\mathbb{Q}(q' \mid q) = \mathcal{N}(q' \mid q, \Sigma) \longrightarrow a(q' \mid q) = \min\left(1, \frac{\pi(q')}{\pi(q)}\right)$$

Problems with MH-MCMC

- Scales poorly with D and complexity of distribution
  - Almost every proposal will be outside the typical set (left)
  - Shrinking the proposal range lead to slow convergence (right)



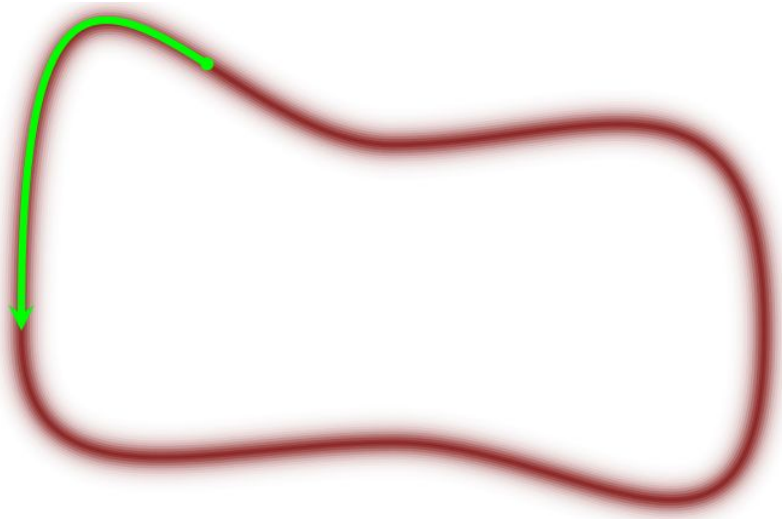
Hamiltonian Monte Carlo

History

- Hybrid Monte carlo
  - Lattice quantum chromodynamics (Duane et al, 1987)
- Bayesian neural networks (Radford Neal, 1995)
- Hamiltonian Monte Carlo (MacKay, 2003)
- Textbooks: MacKay (2003) and Bishop (2006)
- Review by Radford Neal (2011)
- NUTS and Stan PPL (2017)

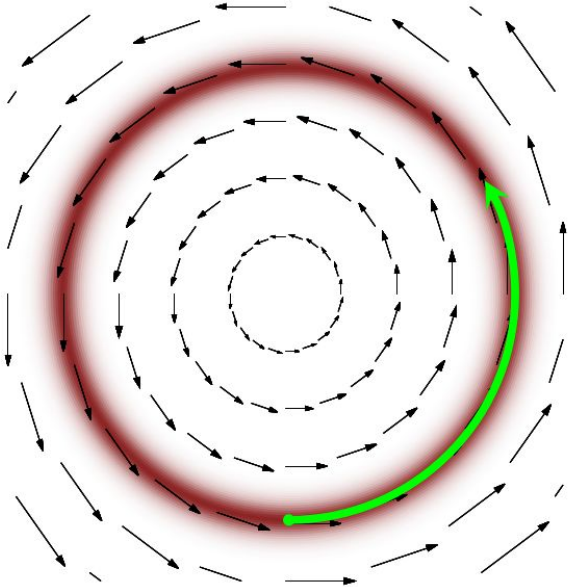
Effective Markov transitions

- Exploit the geometry of the typical set



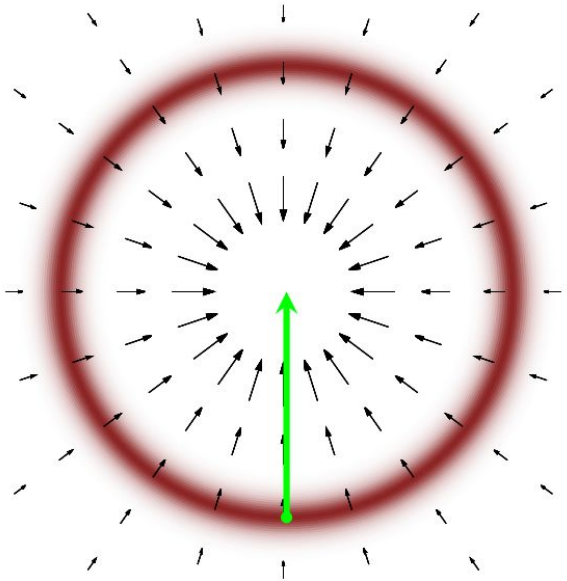
Effective Markov transitions

- Construct a **vector field that is** aligned with the typical set
  - Use the differential structure of the target distribution



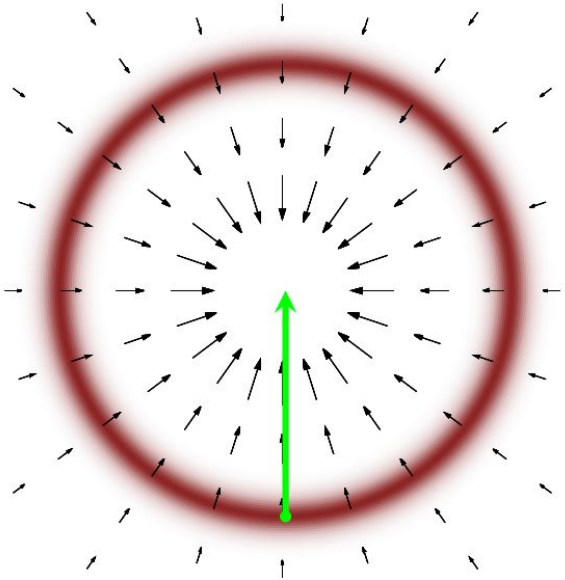
Gradient information

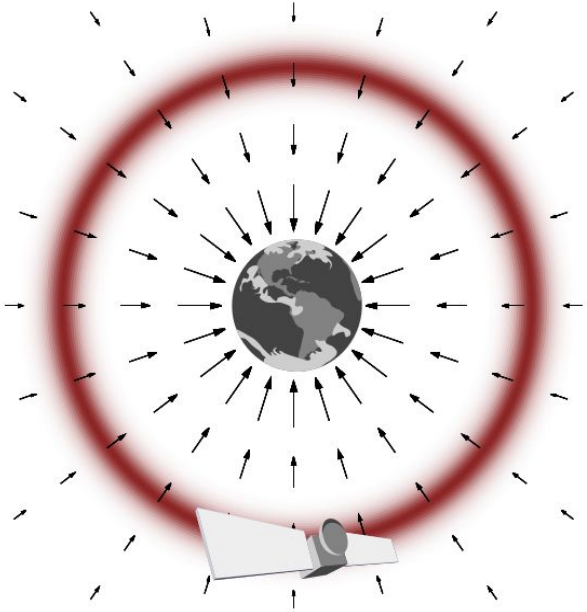
- The gradient will not work
  - The gradient will point to the mode and is sensitive to re-parametrization



Differential geometry

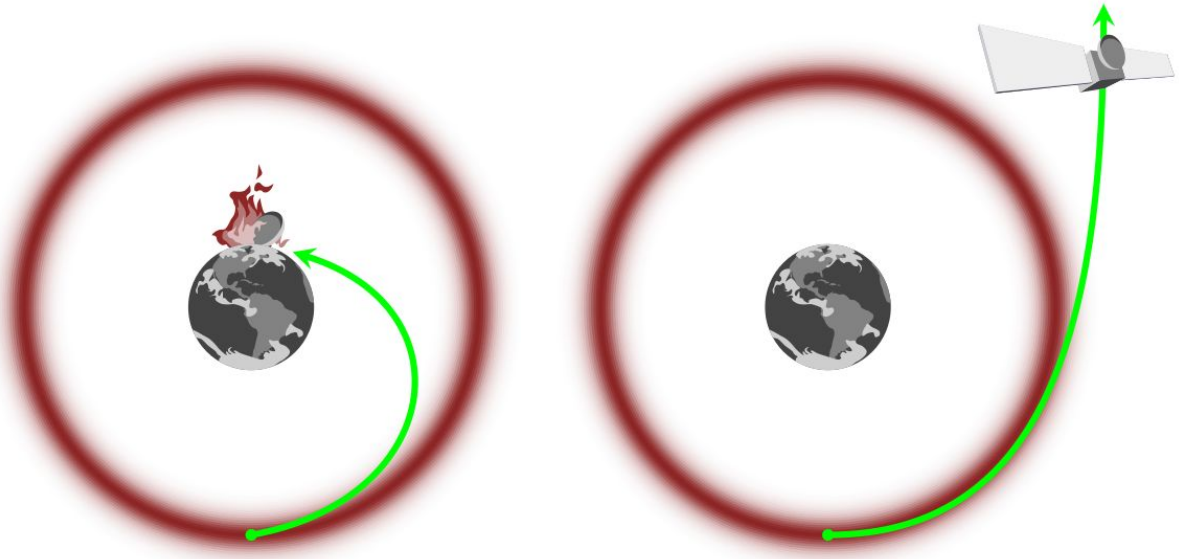
- Gradient-based force field + momentum
  - Physical system: Satellite-in-orbit analogy





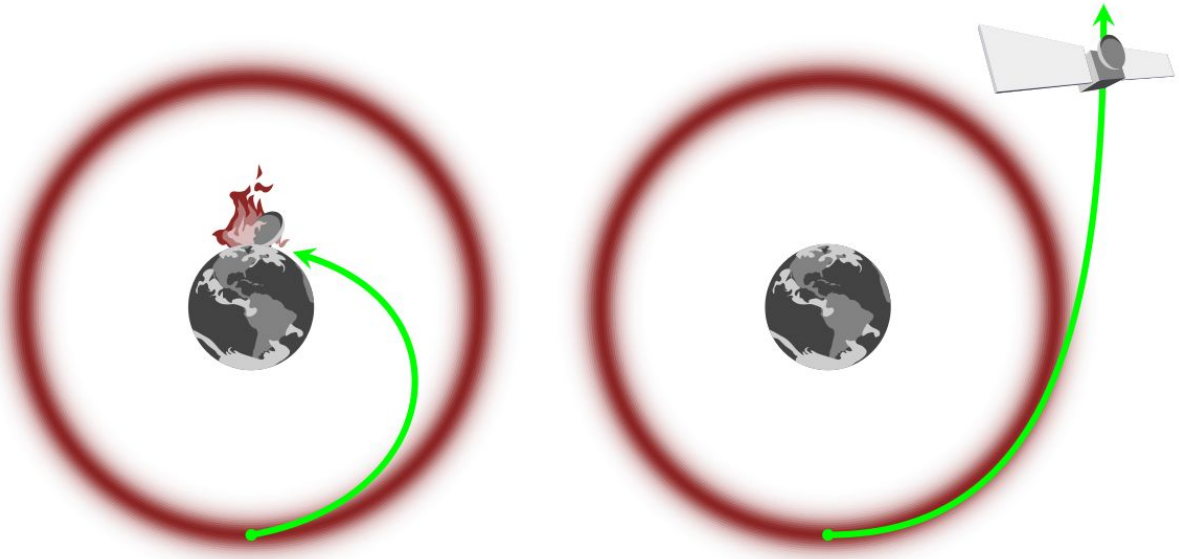
Conservative dynamics

- The momentum needs to be just right
  - We get such conservative dynamics from HMC



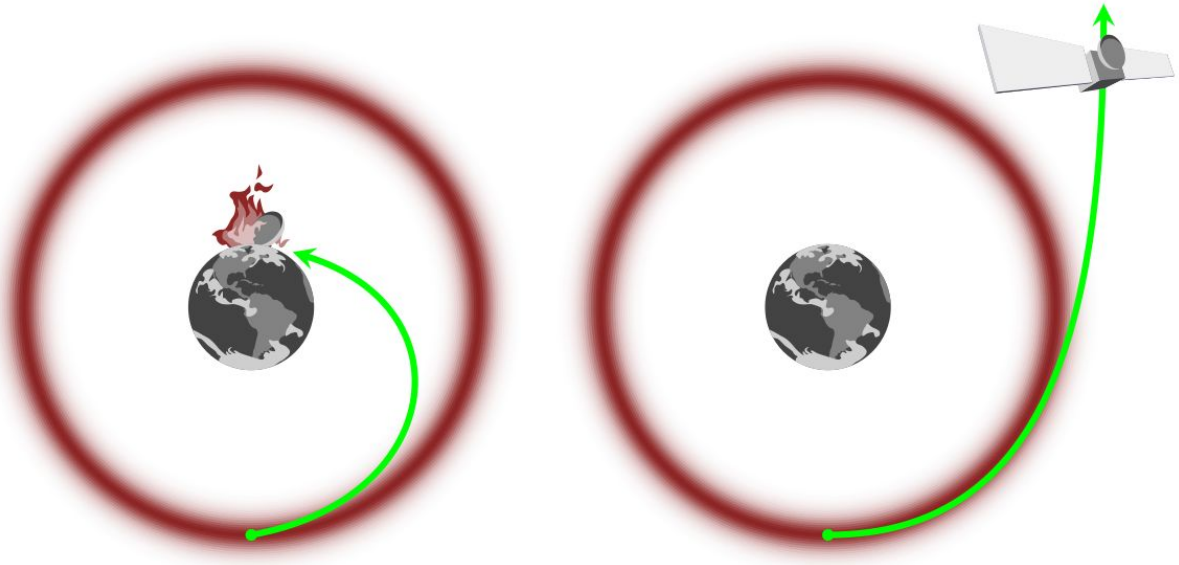
Conservative dynamics

- As the satellite falls towards the planet the momentum grows until it is large enough to propel the satellite away from the planet.



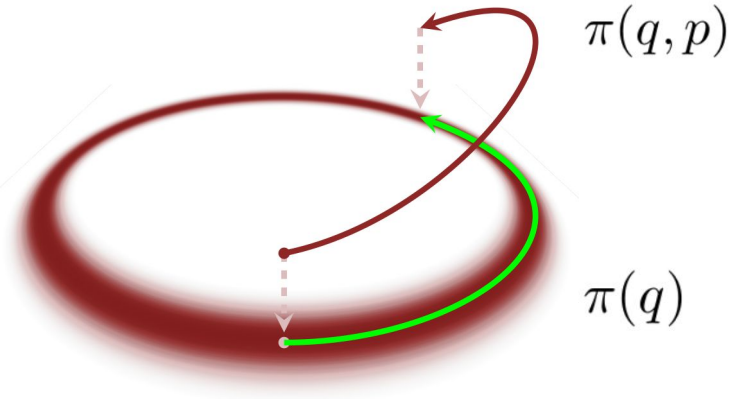
Conservative dynamics

- Similarly, if the satellite drifts away from the planet then the momentum shrinks and the satellite slows, allowing gravity to pull it back towards the planet.



Central idea

- We introduce an auxiliary parameter: the **momentum *p***
  - This parameter will be integrated away to obtain the posterior.
- Thus, we expand the D-dimensional **parameter space** to a 2D-dimensional **phase space**.



Hamiltonian dynamics

- We work in **position-momentum phase-space**

$$q_n \rightarrow (q_n, p_n)$$

- Each of the n parameters q_n is augmented with a momentum p_n
- **Canonical distribution**; joint distribution of p and q
  - H=**Hamiltonian**

$$\pi(q,p) = \pi(p \mid q)\,\pi(q)$$

$$\pi(q,p) = e^{-H(q,p)}$$

Kinetic and potential energy

- The Hamiltonian can be interpreted using a **physical analogy**
  - K(p,q)=**kinetic energy**
  - V(q)=**potential energy**

$$
\begin{aligned}
H(q,p) &= -\log \pi(p \mid q) - \log \pi(q) \\
&\equiv \quad K(p,q) \quad + \; V(q)
\end{aligned}
$$

Kinetic and potential energy

- The potential energy is fully defined by the posterior.
- The kinetic energy must be specified by the HMC implementation.

$$
\begin{aligned}
H(q, p) &= -\log \pi(p \mid q) - \log \pi(q) \\
&\equiv \quad K(p, q) \quad + \; V(q)
\end{aligned}
$$

Hamilton's equations

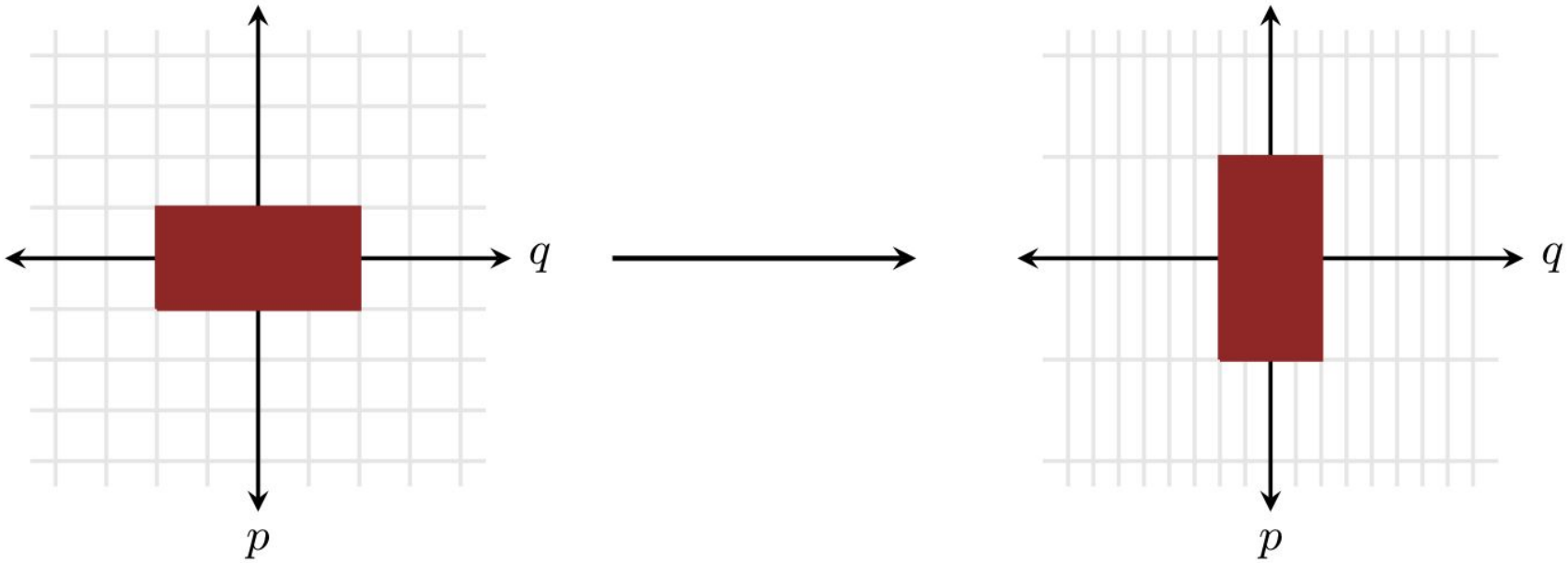
- The desired vector field is obtained from integrating **Hamilton's equations**
  - No collapse to the mode as with the gradient
  - Will sojourn on the typical set

$$\frac{\mathrm{d}q}{\mathrm{d}t} = +\frac{\partial H}{\partial p} = \frac{\partial K}{\partial p}$$
$$\frac{\mathrm{d}p}{\mathrm{d}t} = -\frac{\partial H}{\partial q} = -\frac{\partial K}{\partial q} - \frac{\partial V}{\partial q}$$

Sir William Rowan Hamilton
(1805 - 1865)

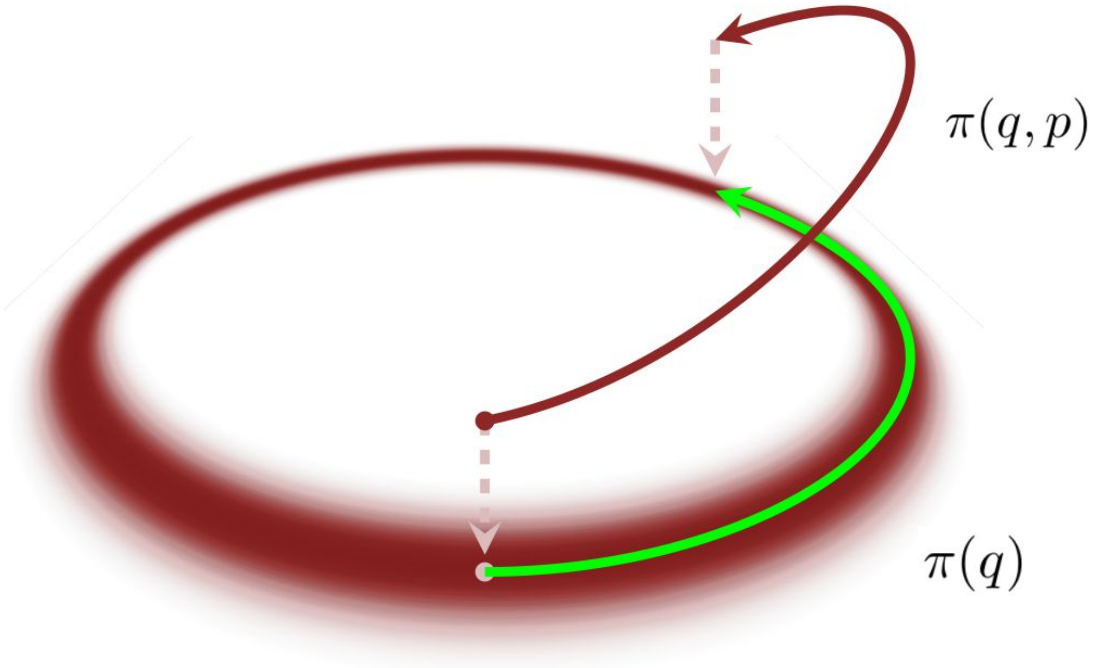
Conservative dynamics

- Volume is preserved in position-momentum phase space



Marginalization to target

- Marginalizing the momentum gives us the target distribution again



Idealized Hamiltonian Markov transition

- **Lift up** an initial point q (random)

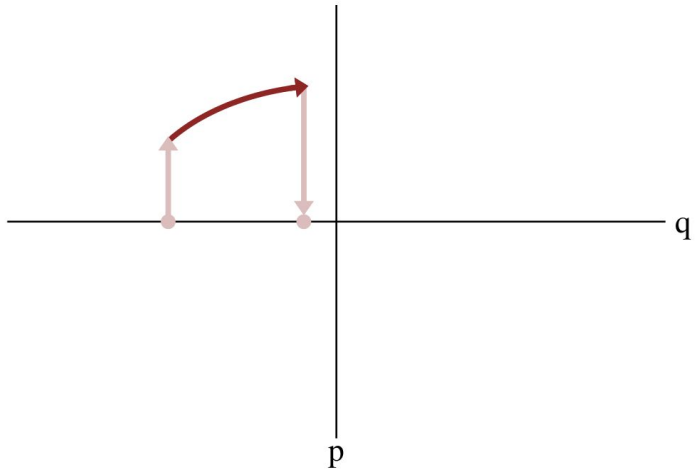
$$p \sim \pi(p \mid q)$$

- **Integrate** Hamilton's equations for some time t (deterministic)

$$(q, p) \to \phi_t(q, p)$$

- Return to target distribution by **projecting away** the momentum p (deterministic)

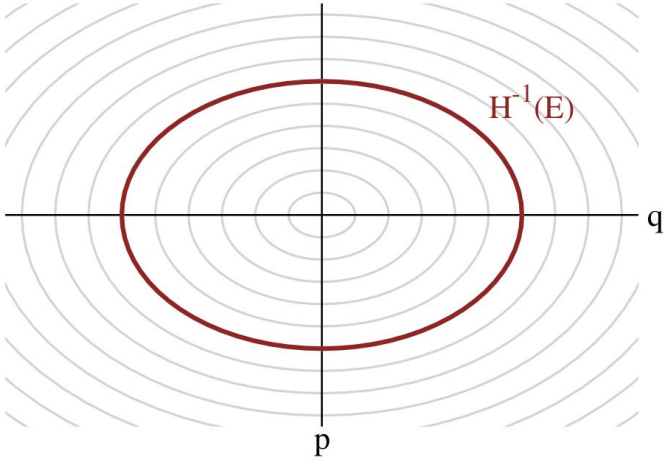
$$(q, p) \to q$$



Geometry of phase space

- Every orbit is confined to an **energy level set**
  - Value of the Hamiltonian is preserved

$$H^{-1}(E) = \{q, p \mid H(q, p) = E\}$$

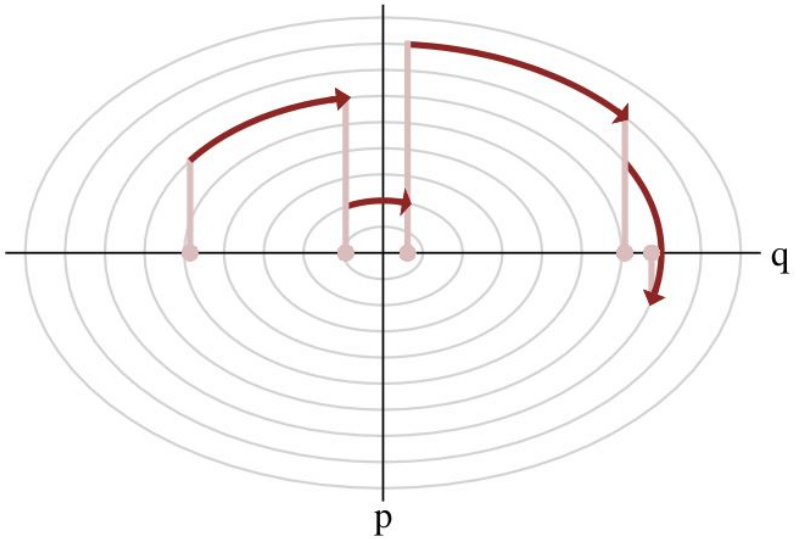


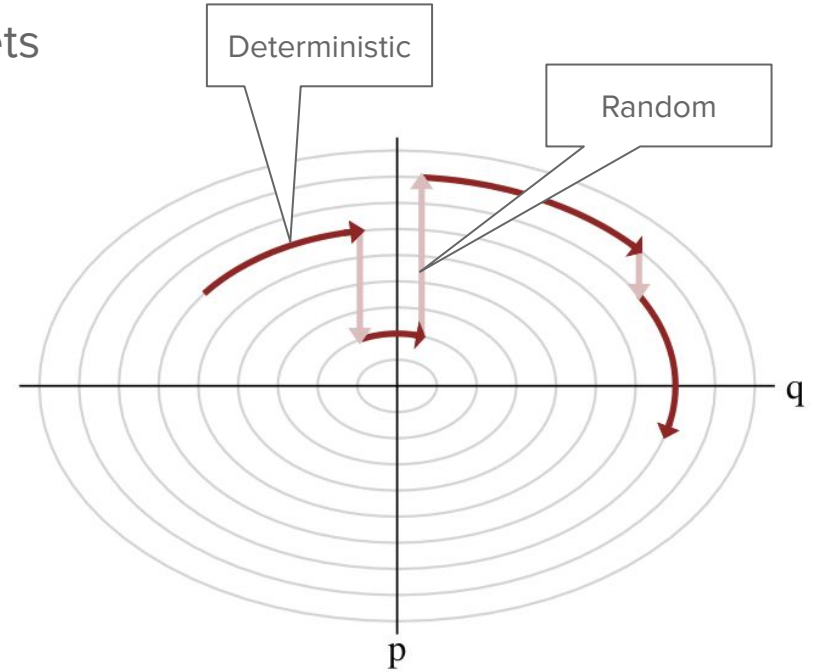
- The concentric level sets decompose or **foliate** the phase space
- **Microcanonical decomposition**
  - θ_E=position within the level set
  - Microcanonical distribution
  - Marginal energy distribution

$$\pi(q, p) = \pi\left(\theta_E \mid E\right) \pi(E)$$

Hamiltonian transitions

- **Deterministic** exploration within a level set
- **Stochastic** exploration between level sets

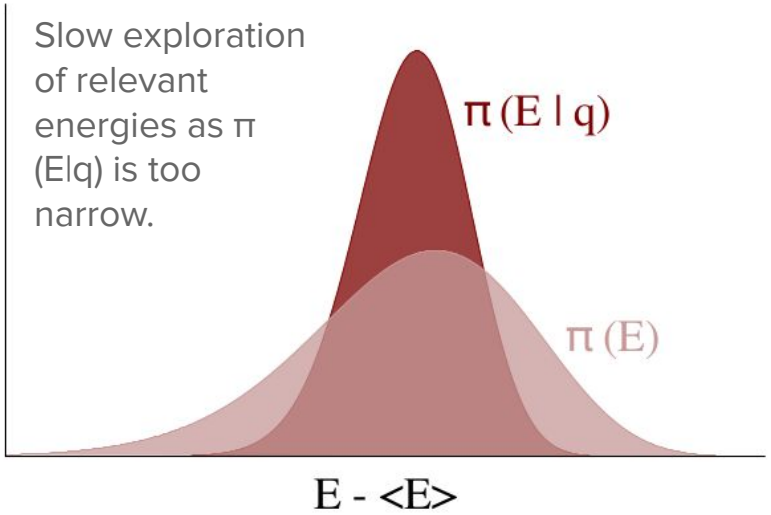


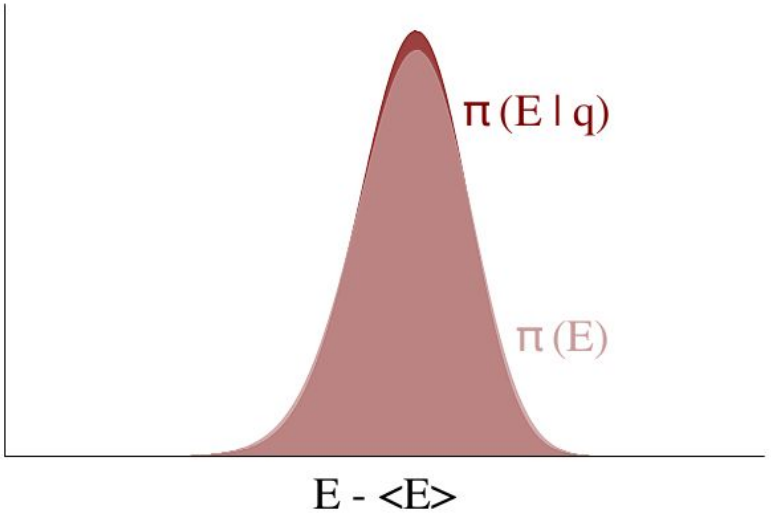


Projection + random lift=**Momentum resampling step**

Momentum resampling

- We have efficient **momentum resampling** when **marginal π(E)** and **transition π(E|q)** are well matched.





Tuning HMC: choice of kinetic energy

- Infinite range of possibilities
- **Euclidean-Gaussian kinetic energy**
  - Euclidean metric *g* for *q*
    - *g* = DxD **mass matrix**
  - M=Rotated (R) and scaled (S) natural metric
  - Also defines a metric for the momenta *p*
  - Construct π(p|q)
  - This defines K

$$\Delta(q, q') = (q - q')^T \cdot g \cdot (q - q')$$

$$M = R \cdot S \cdot g \cdot S^T \cdot R^T$$

$$\Delta(p, p') = (p - p')^T \cdot M^{-1} \cdot (p - p')$$

$$\pi(p \mid q) = \mathcal{N}(p \mid 0, M)$$

$$K(q, p) = \frac{1}{2} p^T \cdot M^{-1} \cdot p + \log |M| + \text{const.}$$

Tuning HMC: Kinetic energy, warm up

- The **choice of M^{-1}** rotates and scales parameter space
  - If close to the covariance of the target distribution, it de-correlates the target distribution.
  - This makes the energy level sets more uniform and easier to explore
- Warm up iterations
  - Get to the typical set
  - Start with a default Euclidean metric
  - Sample from HMC
  - Estimate target covariance
  - Update metric
  - Repeat

$$M^{-1} = \mathbb{E}_{\pi}[(q-\mu)(q-\mu)^{T}]$$

Tuning HMC: Riemannian HMC

- Unless the target is Gaussian, no global rotation and rescaling will yield uniform level sets. Strong local curvature can slow down exploration.
- Solution: Use a **Riemannian metric**, which unlike to Euclidean metric, varies throughout parameter space
- Gaussian distribution whose covariance is **now a function of q**

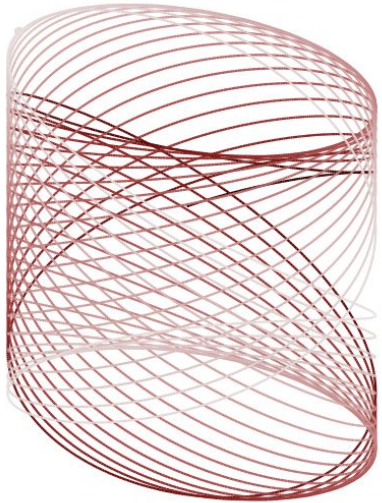
$$\pi(p \mid q) = \mathcal{N}(p \mid 0, \Sigma(q))$$

- **Riemannian-Gaussian kinetic energy**

$$K(q,p) = \frac{1}{2} p^T \cdot \Sigma^{-1}(q) \cdot p + \frac{1}{2} \log |\Sigma(q)| + \text{const}$$

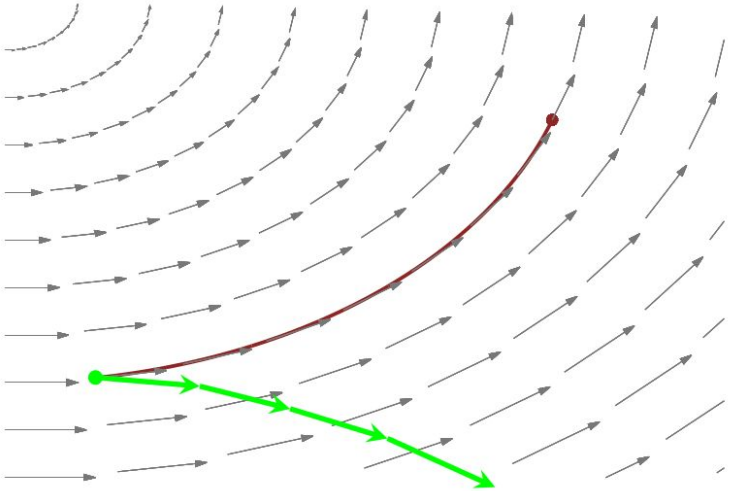
Tuning HMC: Integration time

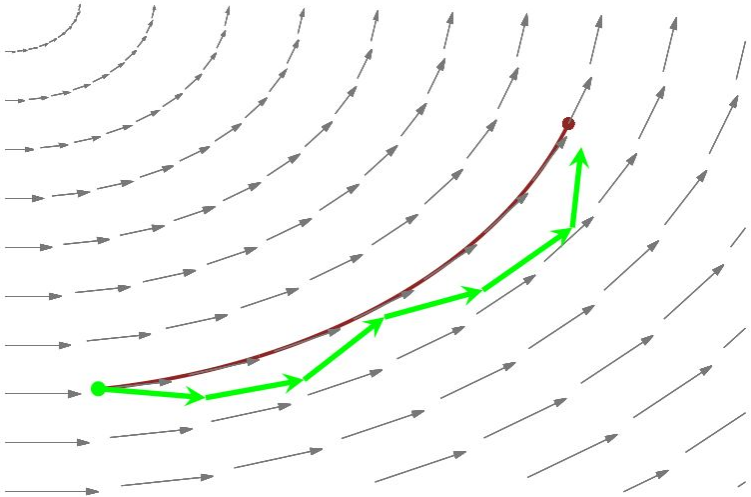
- How long should we follow an orbit?
  - Too long is wasteful, as we might return to visited neighborhoods
  - NUTS: heuristic stopping rule based on avoiding U-turns



Numerical integration

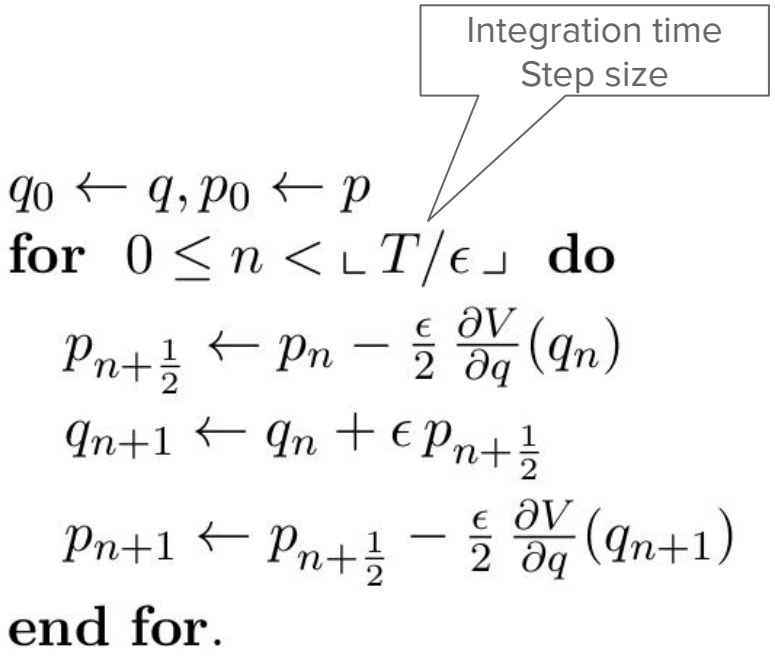
- Solving Hamilton's equations is done numerically
  - Numerical inaccuracies lead to drift





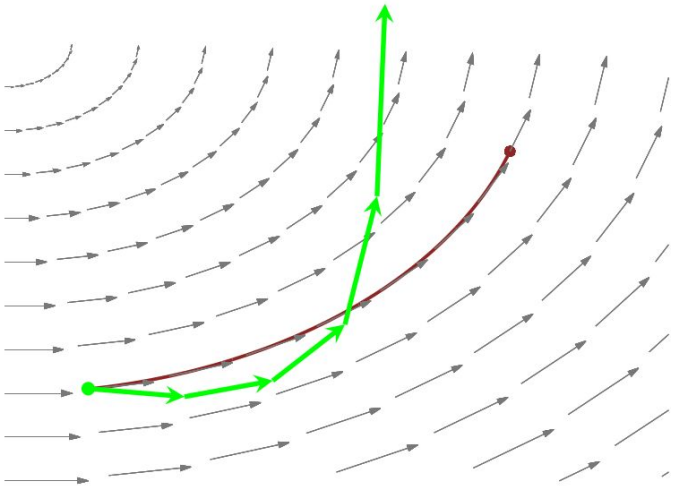
Symplectic integrators

- Robust to drift
- **Exactly preserve phase space volume**
  - Interleave p and q updates
- Oscillate around exact energy level set
- Example
  - **Leapfrog estimator**



Divergence

- Even symplectic integrators can diverge around regions of **high curvature**
  - Easy to identify (infinite energy) and thus used for **diagnostics**



As MH-MCMC proposal

- Idea: Use Hamiltonian transition as a MH-MCMC proposal
  - Easy way to get rid of symplectic integrator error
- But conventional Hamiltonian transitions are **deterministic and not reversible**

$$\mathbb{Q}(q_L, p_L \mid q_0, p_0) = 1$$

$$\mathbb{Q}(q_0, p_0 \mid q_L, p_L) = 0$$

Reversible Hamiltonian transitions

- Flip the sign of the momentum

$$\mathbb{Q}(q_L, -p_L \mid q_0, p_0) = 1$$

$$\mathbb{Q}(q_0, p_0 \mid q_L, -p_L) = 1$$

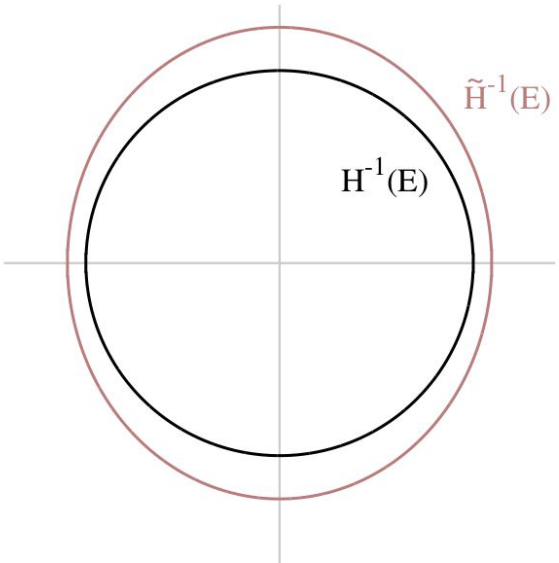
Reversible proposal

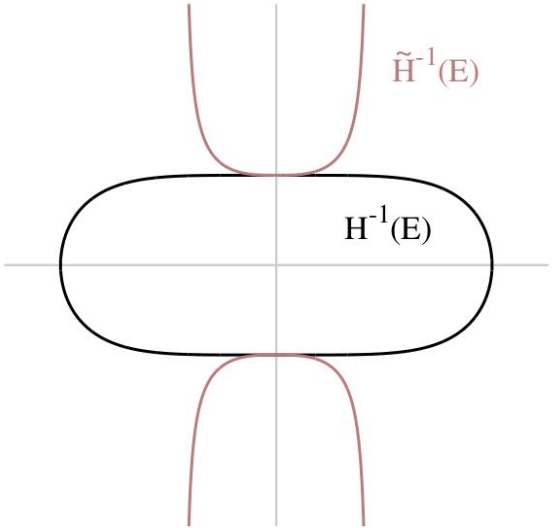
- Metropolis-Hasting with the reversible proposal

$$
\begin{aligned}
a(q_L, -p_L \mid q_0, p_0) &= \min\left(1, \frac{\mathbb{Q}(q_0, p_0 \mid q_L, -p_L)\,\pi(q_L, -p_L)}{\mathbb{Q}(q_L, -p_L \mid q_0, p_0)\,\pi(q_0, p_0)}\right) \\
&= \min\left(1, \frac{\delta(q_L - q_L)\,\delta(-p_L + p_L)\,\pi(q_L, -p_L)}{\delta(q_0 - q_0)\,\delta(p_0 - p_0)\,\pi(q_0, p_0)}\right) \\
&= \min\left(1, \frac{\pi(q_L, -p_L)}{\pi(q_0, p_0)}\right) \\
&= \min\left(1, \frac{\exp\left(-H(q_L, -p_L)\right)}{\exp\left(-H(q_0, p_0)\right)}\right) \\
&= \min(1, \exp\left(-H(q_L, -p_L) + H(q_0, p_0)\right)),
\end{aligned}
$$

Tuning HMC: Symplectic integrator

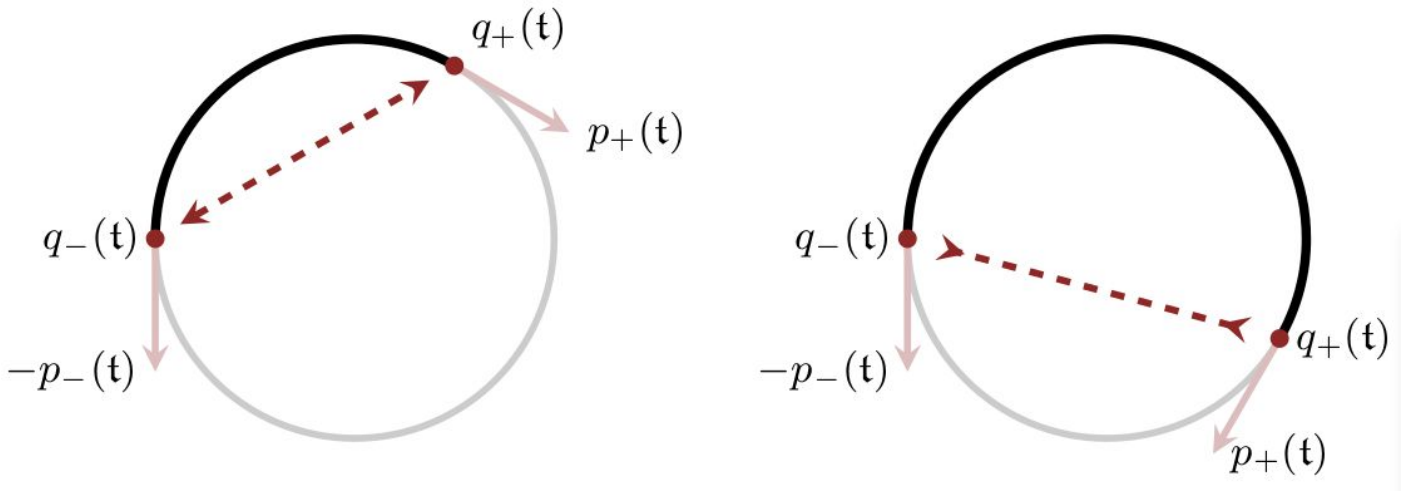
- Two hyperparameters: **step size ε** and **#gradient evaluations K**
- Compare with a “shadow Hamiltonian” H for which the integrator is exact
  - We should get the same topology





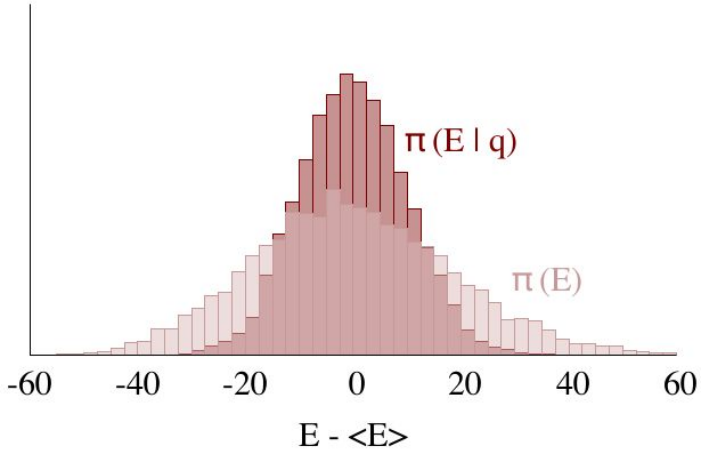
No-U-Turn (NUTS) termination criterion

- Stop when trajectories, expanded in both directions, turn towards each other
  - Hofman & Gelman, 2014; Stan PPL



Robustness and problems

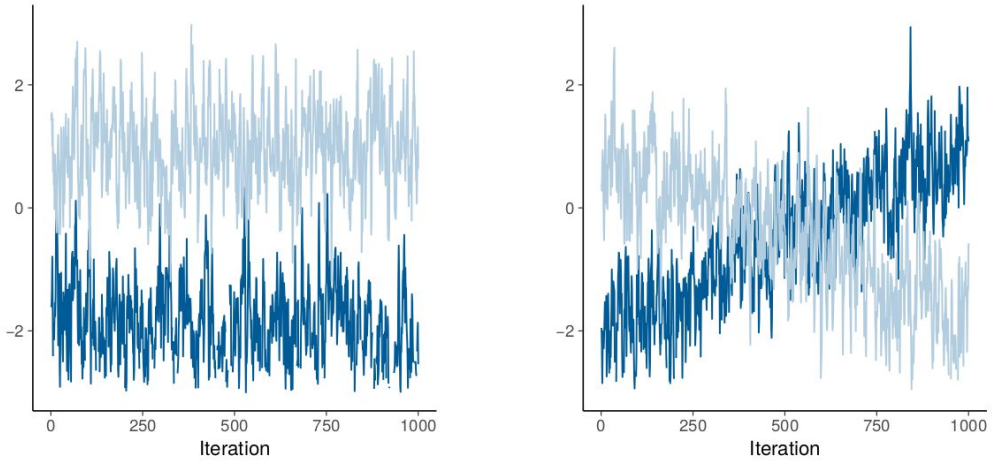
- **Heavy tails** due to bad model or bad kinetic energies
  - Visualize marginal and transition energy density.
  - Is π(E|q) narrower than π(E)?
  - Can be done in [Arviz]
- **Large curvature** (pinching)
  - Hierarchical models are problematic
  - Divergent trajectories
- General diagnostics
  - R-split, ESS, rank plots...



Diagnostics

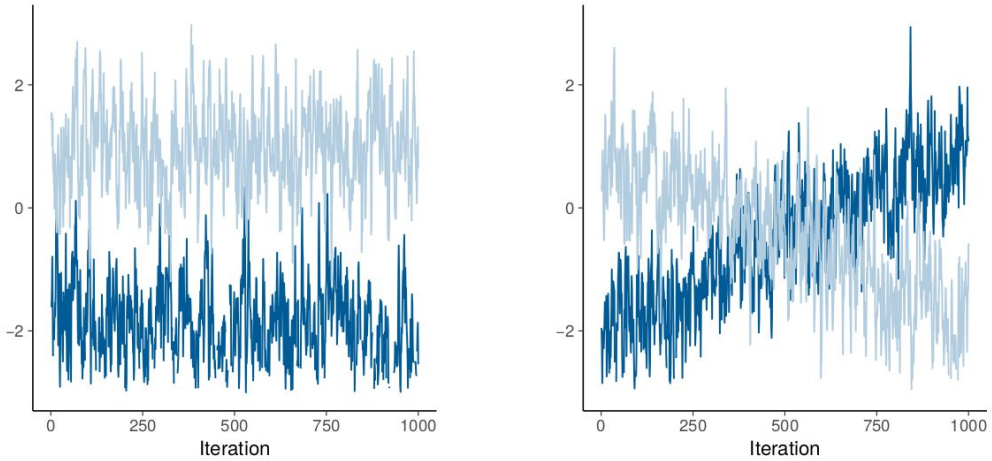
Convergence diagnostics

- MCMC is guaranteed to converge to the posterior for infinite samples.
  - But there are rarely any strong guarantees for finite samples.
- Diagnostics are needed, for example from running multiple chains.
  - **Trace plots**



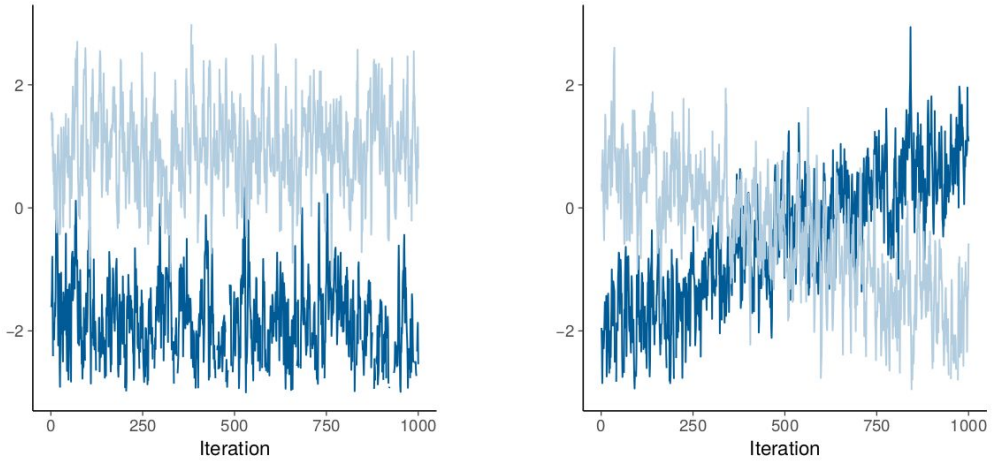
Trace plots

- **Left:** trace plots look stable, but did not converge to the same distribution.
- **Right:** trace plots are not stationary, though they seem to cover similar distributions.



Trace plots

- We need both **between-sequence** and **within-sequence** diagnostics.
- We can't visually inspects trace plots of 1000s of variables.
  - We need **numerical summaries**.



Split-R-hat

- Question: “Did the chains mix well?”
  - The **mixing time** is the time until the chain is "close" to its steady state distribution.
  - We have M chains with we have N samples per chain
- B=**between-chain variance**

$$B = \frac{N}{M-1} \sum_{m=1}^{M} (\overline{\theta}^{(.m)} - \overline{\theta}^{(..)})^2$$

Average for chain m

Average for all chains

$$\overline{\theta}^{(.m)} = \frac{1}{N} \sum_{n=1}^{N} \theta^{(nm)}, \quad \overline{\theta}^{(..)} = \frac{1}{M} \sum_{m=1}^{M} \overline{\theta}^{(.m)}$$

Split-R-hat

- Question: “Did the chains mix well?”
  - We have M chains
  - We have N samples per chain
- W=**within-chain variance**

$$W = \frac{1}{M} \sum_{m=1}^{M} s_m^2$$

$$s_m^2 = \frac{1}{N-1} \sum_{n=1}^{N} (\theta^{(nm)} - \overline{\theta}^{(.m)})^2$$

Split-R-hat

- Weighted average of W and B

$$\widehat{\text{var}}^{+}(\theta|y) = \frac{N-1}{N}W + \frac{1}{N}B.$$

- Split-R-hat, which **declines to 1 for N → inf**

$$\widehat{R} = \sqrt{\frac{\widehat{\text{var}}^{+}(\theta|y)}{W}}$$

Monte Carlo standard error (MCSE)

- If we have S **independent** samples, the accuracy of the sample average estimator $\bar{\theta}$ for the posterior mean $\mathbb{E}(\theta \mid y)$ is

$$\mathrm{Var}(\bar{\theta}) = \frac{\mathrm{Var}(\theta|y)}{S}$$

- The square root is called the **MCSE**
- But MCMC samples are **correlated**!
  - We need to replace S with the effective sample size (ESS)

Effective sample size (ESS)

- The **effective sample size (ESS)** measures the worth of the MCMC estimator.
- It is defined as the number of samples simulated from the target pdf $\mathbf{S}_{\mathbf{eff}}$ that would provide an estimator with a variance equal to the variance of the MCMC estimator based on **S** samples.

$$\mathrm{Var}(\bar{\theta}) = \frac{\mathrm{Var}(\theta|y)}{S_{\mathrm{EFF}}} = \frac{\mathrm{Var}(\theta_S)}{S} \Rightarrow S_{\mathrm{EFF}} = S\frac{\mathrm{Var}(\theta|y)}{\mathrm{Var}(\theta_S)}$$

Can be approximated

Here we assume independence

Effective sample size (ESS)

- MCMC samples are not independent, but correlated.
- The ESS heuristic estimates the effective number of independent samples
- **Single chain case**
  - Autocorrelation ρ_t is calculated using FFT

$$N_{\text{eff}} = \frac{N}{\sum_{t=-\infty}^{\infty} \rho_t} = \frac{N}{1 + 2\sum_{t=1}^{\infty} \rho_t}$$

$$\rho_t = \frac{1}{\sigma^2} \int_{\Theta} (\theta^{(n)} - \mu)(\theta^{(n+t)} - \mu)\, p(\theta)\, d\theta$$

Effective sample size (ESS)

- MCMC samples are not independent, but correlated.
- The ESS heuristic estimates the effective number of independent samples
- **Multiple chain case**
  - Uses quantities calculated for split-R-hat
  - Uses truncated lag (T)

$$S_{\text{eff}} = \frac{NM}{1 + 2\sum_{t=1}^{T} \rho_t}$$

$$\hat{\rho}_t = 1 - \frac{W - \frac{1}{M}\sum_{m=1}^{M} s_m^2 \hat{\rho}_{t,m}}{\widehat{\text{var}}^+}$$

ESS rules-of-thumb

- **Antithetic Markov chains**
  - Chains that have negative autocorrelations
  - Super-efficient chains with $S_{eff} > S$
  - This can happen for NUTS, for example, but of limited practical use
- For four chains, in order to use R-hat diagnostics
  - **Multichain ESS>400**
  - **Individual chain ESS>50**
  - **R-hat is close to 1**

HMC in Pyro

NUTS in Pyro

pyro.sample("x", ...)

```
# Run HMC / NUTS
nuts_kernel=pyro.infer.NUTS(model, jit_compile=True)
mcmc=pyro.infer.MCMC(nuts_kernel, num_samples=2000, num_chains=2,
    warmup_steps=100)
mcmc.run()
```

pyro.sample("x", ...)

```
# Get the samples
samples = mcmc.get_samples()["x"]
```

Diagnostics with Arviz

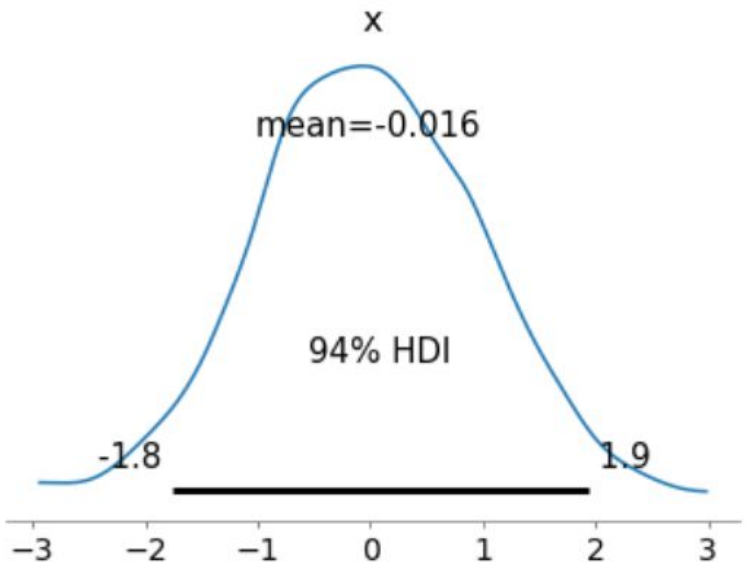
- https://arviz-devs.github.io/arviz/
- “ArviZ is a Python package for **exploratory analysis of Bayesian models**.
- Includes functions for **posterior analysis**, data storage, **sample diagnostics**, **model checking**, and comparison.
- The goal is to provide **backend-agnostic tools** for diagnostics and visualizations of Bayesian inference in Python, by first converting inference data into xarray objects.”

Diagnostics with arviz

```
data = arviz.from_pyro(mcmc)
# ESS, r-hat
summary = arviz.summary(data)
print(summary)
# Density plot
arviz.plot_posterior(data)
plt.show()
```

Diagnostics with arviz

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
x	-0.016	0.985	-1.757	1.937	0.029	0.021	1154.0	1453.0	1.0



Conclusions

- Monte Carlo + Hamiltonian dynamics
  - Parameters + momentum
- HMC solves many classic MCMC issues
- NUTS fully automates HMC
  - Ideal for PPLs (Stan, Numpyro)
  - Diagnosis with arviz
- Scaling to tall data is a timely problem
  - Subsampling HMC
  - Subsampling HMC in Numpyro by PhD student Ola Rønning