

## Lecture 5: modern MCMC algorithms

Ben Lambert<sup>1</sup>

`ben.lambert@some.ox.ac.uk`

<sup>1</sup>Somerville College  
University of Oxford

Wednesday 23<sup>rd</sup> November, 2016

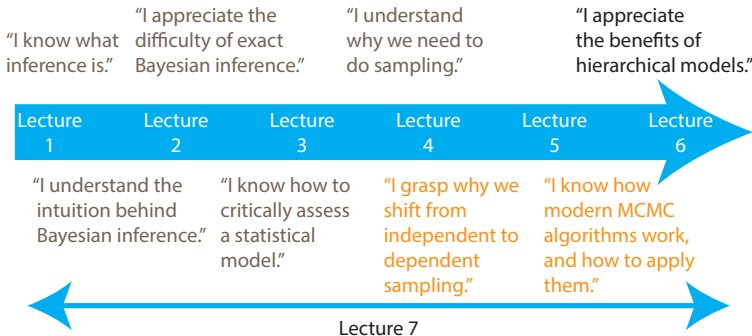
# Lecture outcomes

By the end of this lecture you should:

- ① Understand how dependent sampling via MCMC can be used to sample from posterior distributions.
- ② Grasp how the concept of “effective sample size” quantifies the information cost of dependent sampling.
- ③ Understand the basic mechanics and intuition behind Random Walk Metropolis.
- ④ Know how the Gibbs sampler works and how it compares to Random Walk Metropolis.
- ⑤ Recognise the underlying problem with Random Walk Metropolis and Gibbs.
- ⑥ Recognise that Hamiltonian Monte Carlo overcomes some of the problems of Random Walk Metropolis and Gibbs.



# Overall course outline



- 1 Recap from last lecture
- 2 Start to finish Bayesian inference
- 3 Metropolis-Hastings
- 4 Gibbs sampling
- 5 Hamiltonian Monte Carlo

# What is independent sampling?

**Definition:**

“A sample drawn from a distribution that does not depend on any previous samples drawn.”

# A common misconception about independent sampling

Suppose I can write down the pdf for a distribution:

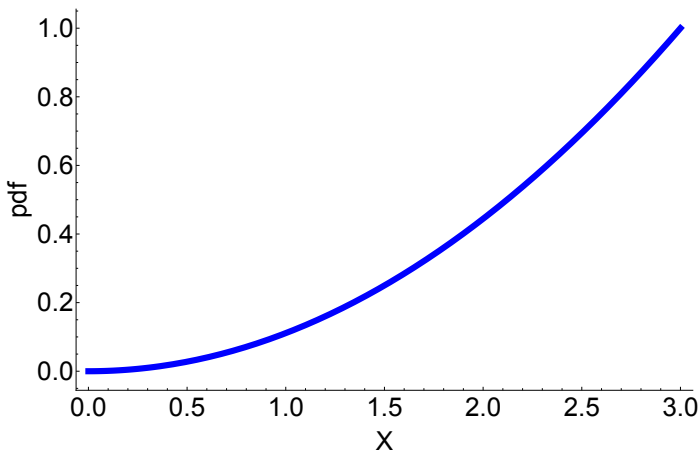
$$f(x) = \frac{1}{9}x^2 \quad (1)$$

where  $0 \leq x \leq 3 \implies$  a valid probability distribution!

# A common misconception about independent sampling

And we can draw this function...

**Question:** doesn't this mean we can automatically sample from it?



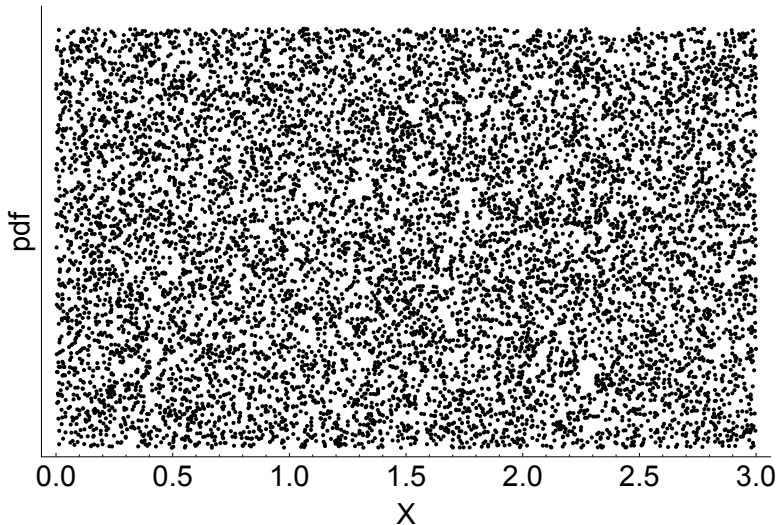
# A common misconception about independent sampling

**Answer:** no!

- No inbuilt command in statistical software to sample from our function.
- $\implies$  use Rejection sampling.

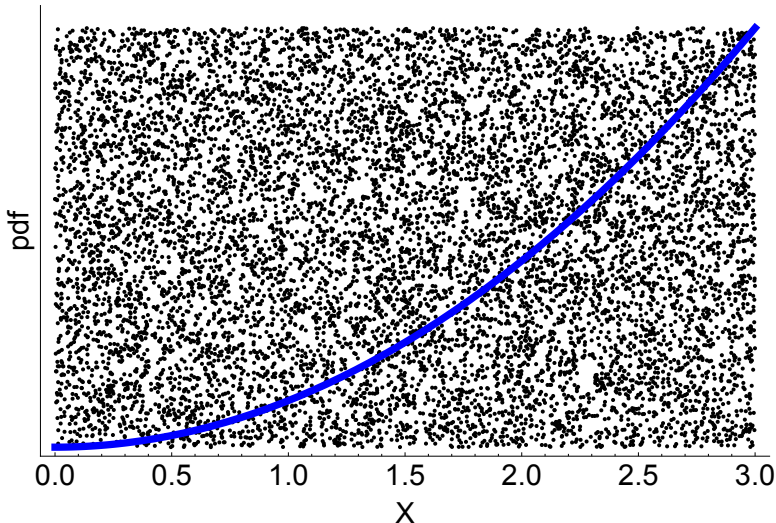
# A common misconception about independent sampling

Generate  $(x, y)$  pairs at random from continuous uniform.



# A common misconception about independent sampling

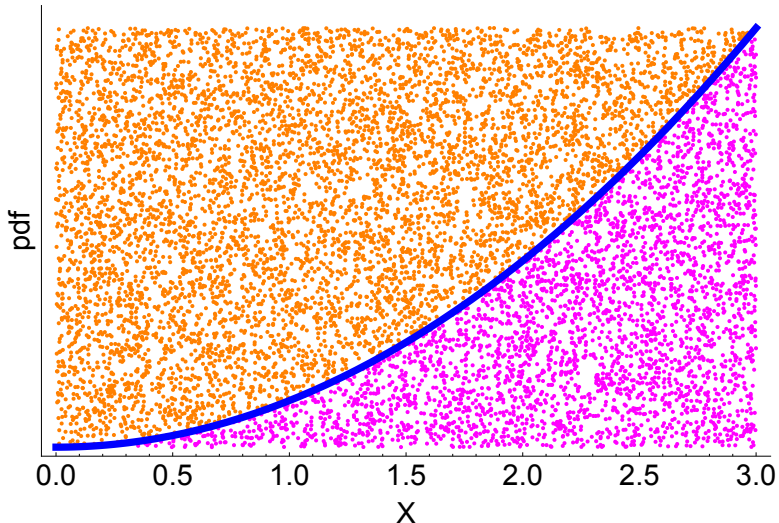
Overlay our distribution's pdf.





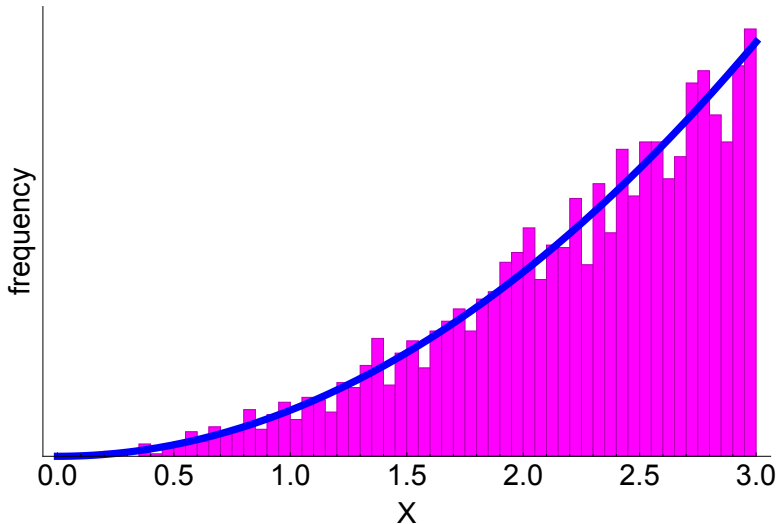
# A common misconception about independent sampling

Accept those  $x$  samples with a  $y$  value below pdf.



# A common misconception about independent sampling

Histogram of  $x$  samples.



## What is dependent sampling?

“A sampling algorithm where the next sample **depends** on the current value.”

# The war of independence

- Think of independent samplers as paratroopers.
- Dependent samplers (MCMC!) as infantry.



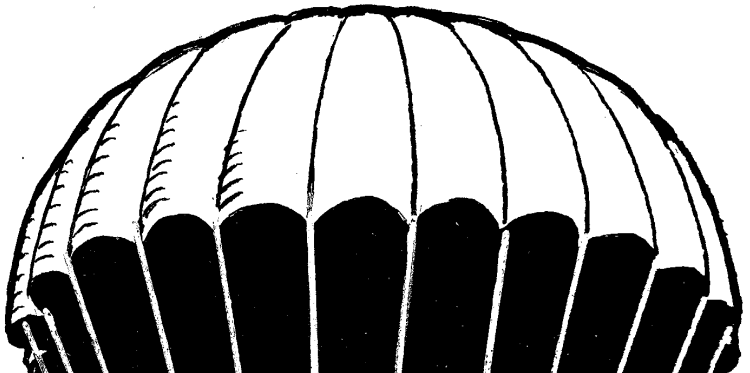
## Independent samples as paratroops

## Dependent samples as infantry

# The winner

Independent troops/sampler! Because:

- Their aerial overview gives them a better ability to plan samples.
- They can traverse a terrain more rapidly than ground troops (dependent sampler).



# The Bayesian battle winner

- In Bayesian inference  $\implies$  the posterior distribution is too complex (aerial overview impossible) to do independent sampling.
  - However we can still do dependent sampling!
- $\implies$  Infantry wins!





# Random Walk Metropolis algorithm

**Question:** how should we step across the terrain of the posterior to ensure we generate samples from the posterior?

**Answer:** use **Random Walk Metropolis** algorithm.

# Random Walk Metropolis algorithm: definition

- ① Start in a random location  $\theta_0 \in \Theta$ .
- ② For times  $t = 1 \dots T$  do:
  - Propose a new location using **symmetric** jumping kernel,  
 $\theta_{t+1} \sim J(\theta_{t+1}|\theta_t)$ .
  - Calculate:

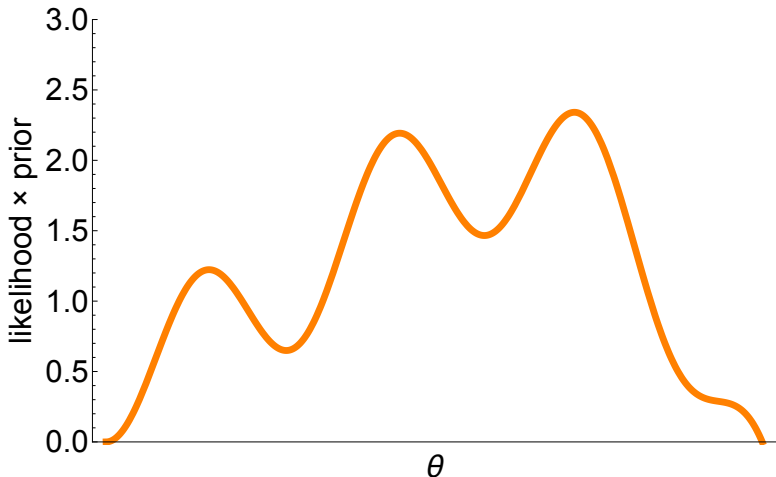
$$r = \frac{\text{likelihood}(\theta_{t+1}) \times \text{prior}(\theta_{t+1})}{\text{likelihood}(\theta_t) \times \text{prior}(\theta_t)} \quad (2)$$

$\implies$  independent of denominator!

- Generate  $u \sim \text{uniform}(0, 1)$ .
- If  $r > u$  we move from  $\theta_t \rightarrow \theta_{t+1}$ ; otherwise we stay at  $\theta_t$ .

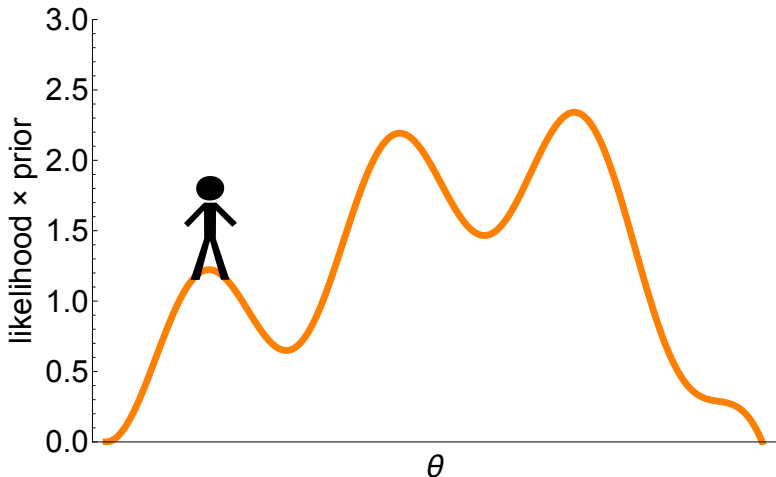
# Defining Random Walk Metropolis

Start with the un-normalised density.



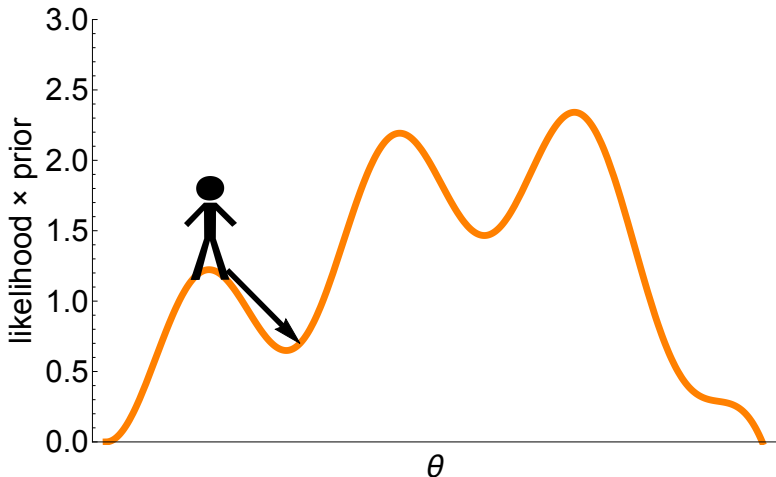
# Defining Random Walk Metropolis

Select a random starting location.



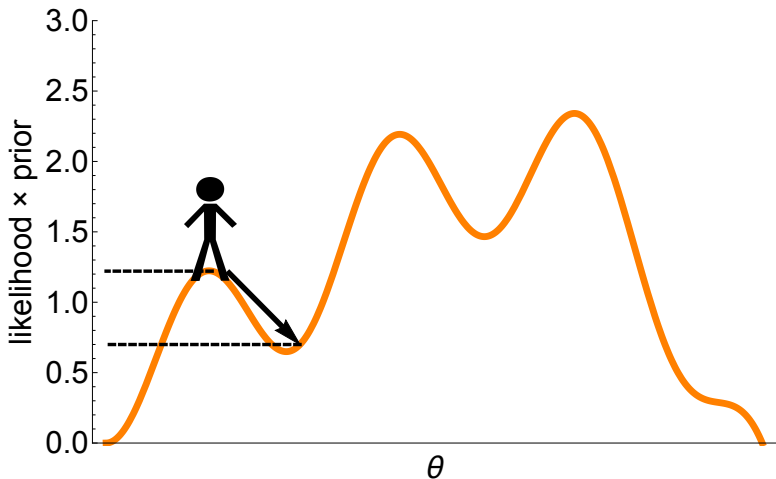
# Defining Random Walk Metropolis

Propose a new location using jumping distribution.



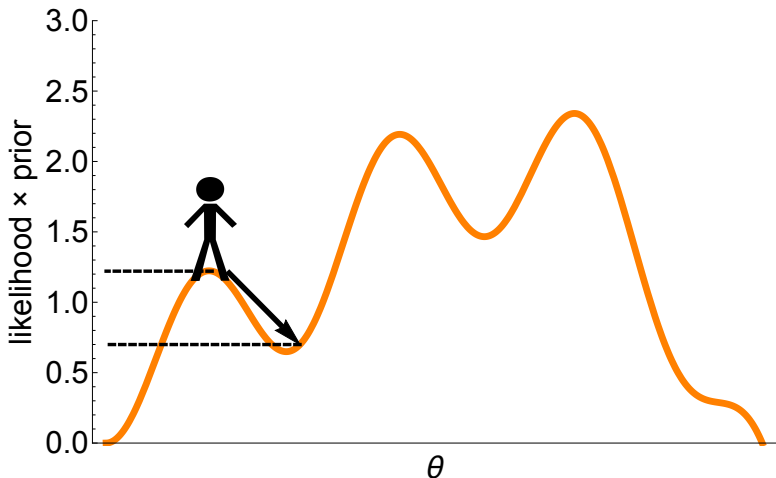
# Defining Random Walk Metropolis

Calculate ratio of likelihood  $\times$  prior at proposed to current location, and find  $r \approx 0.58$ .



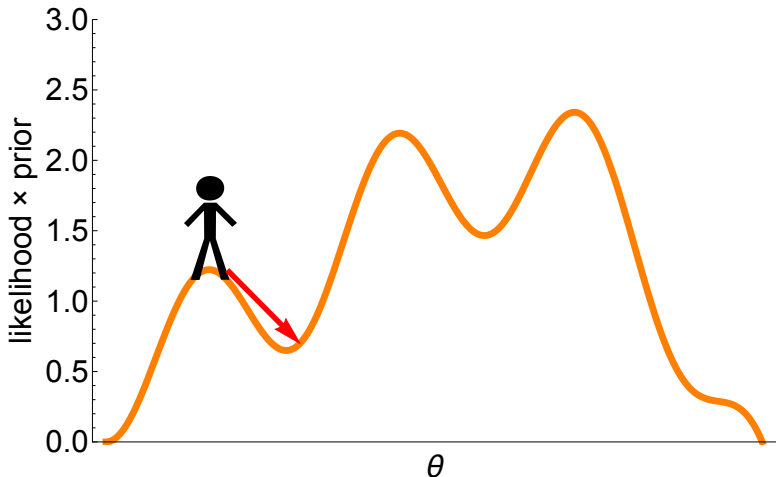
# Defining Random Walk Metropolis

Compare  $r \approx 0.58$  with random real between 0 and 1. For example suppose we obtain  $u = 0.823$ .



# Defining Random Walk Metropolis

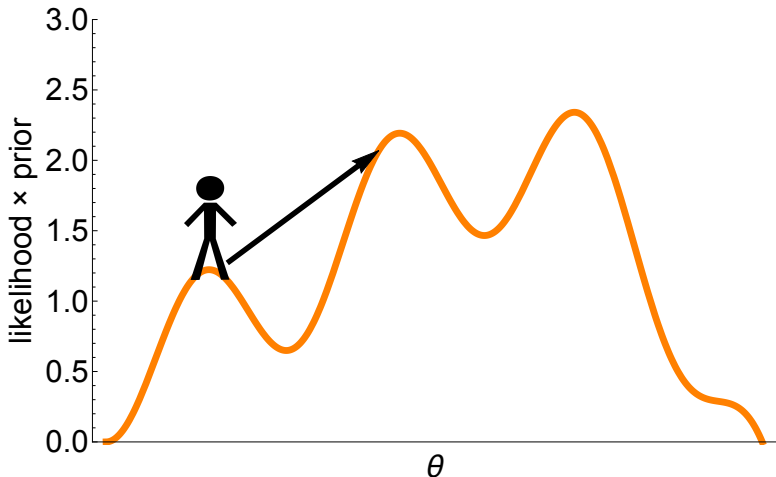
Since  $r < u$  we remain at our original location.





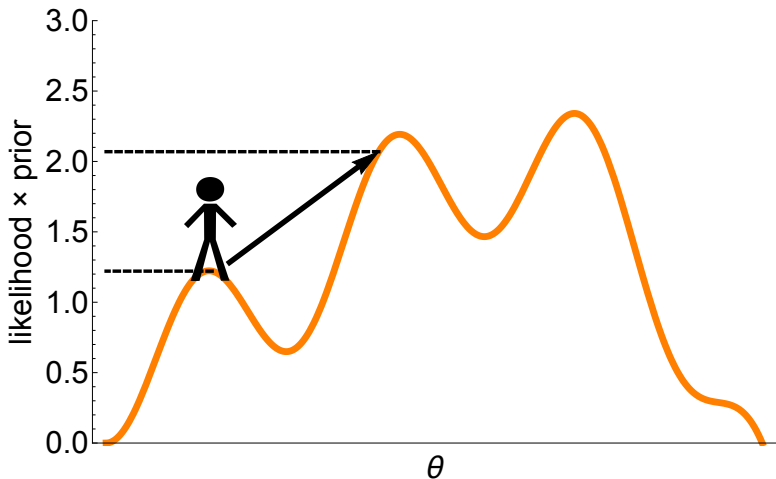
# Defining Random Walk Metropolis

Generate a new proposed step using jumping distribution.



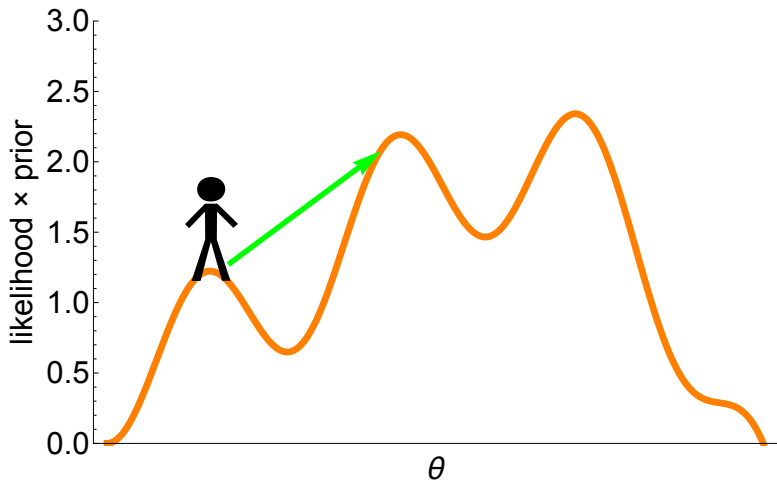
## Defining Random Walk Metropolis

Calculate ratio of likelihood  $\times$  prior at proposed to current location, and find  $r \approx 1.75$ .



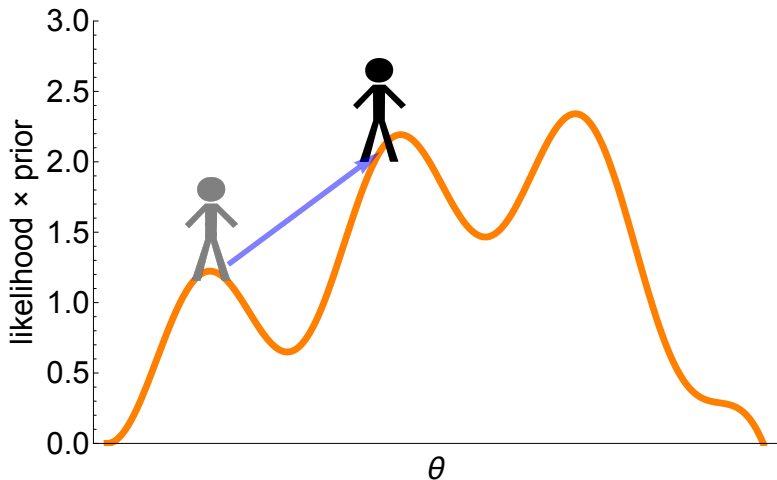
# Defining Random Walk Metropolis

Since  $r > 1$  (maximum possible  $u$ )  $\implies$  we move to new location.



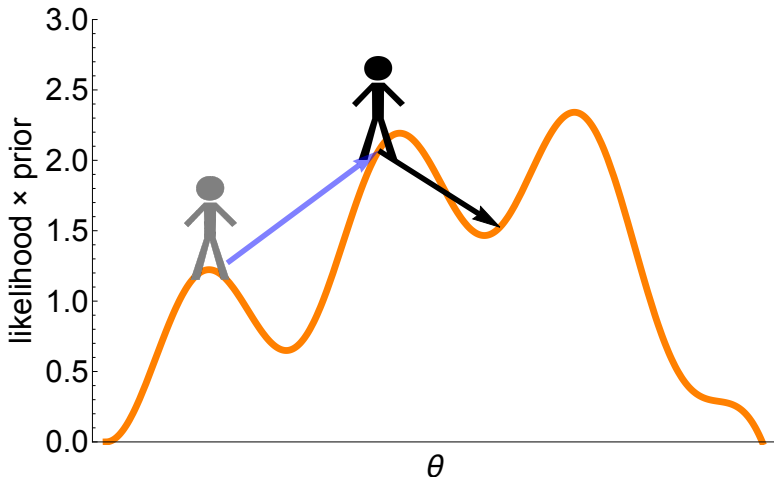
# Defining Random Walk Metropolis

Since  $r > 1$  (maximum possible  $u$ )  $\implies$  we move to new location.



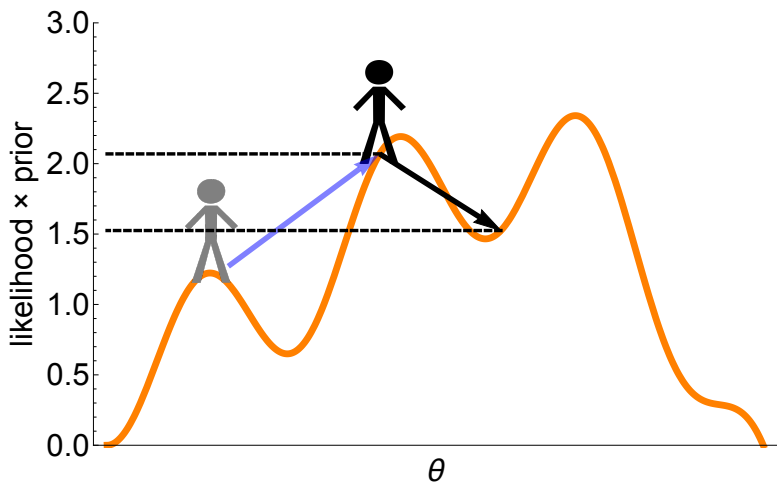
# Defining Random Walk Metropolis

Propose a new step using jumping distribution.



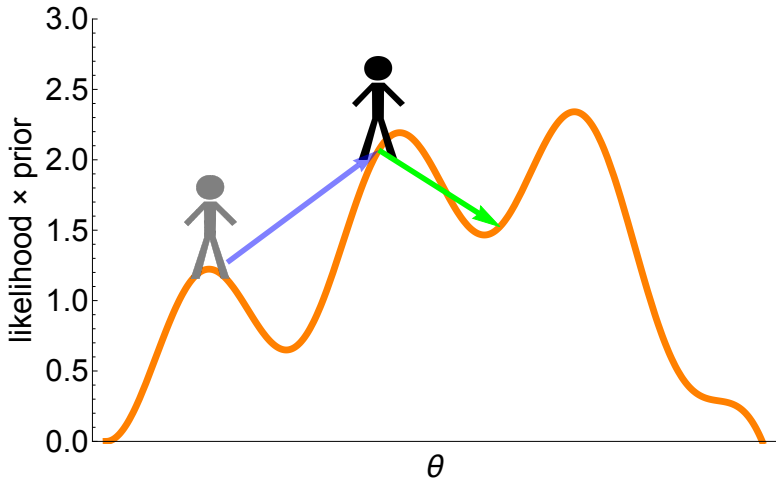
# Defining Random Walk Metropolis

Calculate  $r \approx 0.75$ .



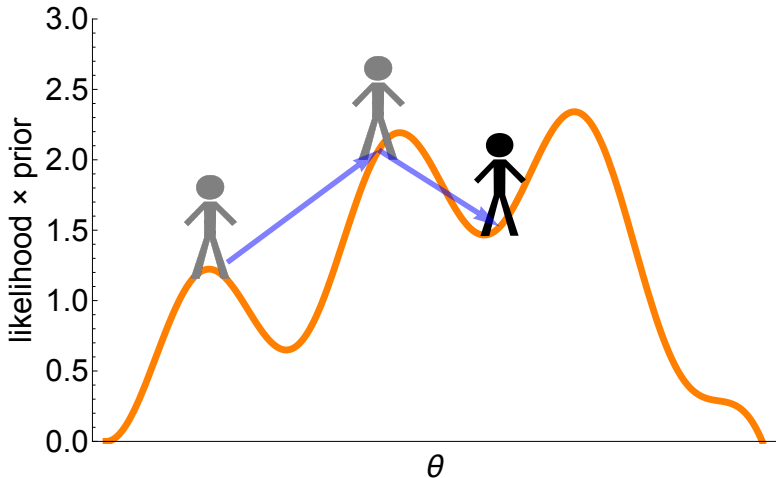
# Defining Random Walk Metropolis

Generate  $u = 0.278 < r \implies$  we move!



# Defining Random Walk Metropolis

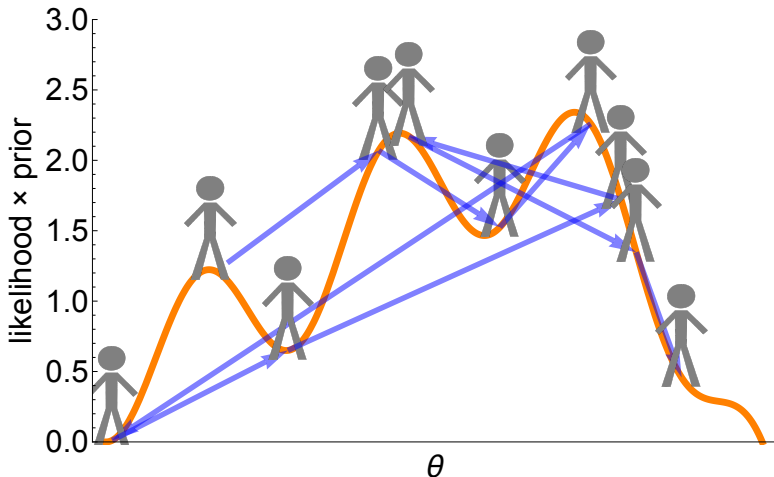
Generate  $u = 0.278 < r \implies$  we move!





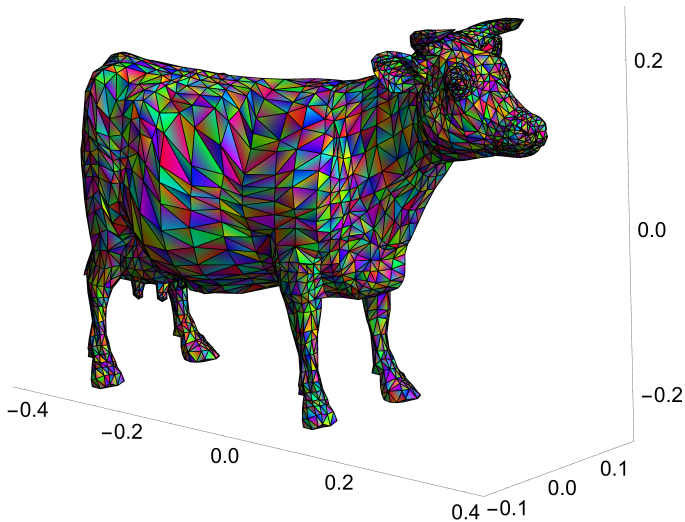
# Defining Random Walk Metropolis

Repeat a large number of times.



# Example Random Walk Metropolis: cow revisited

**Question:** remember the cow?



## Example Random Walk Metropolis: cow revisited

Define a distribution:

$$p(r) \propto \exp(-100r) \quad (3)$$

where  $r$  is the shortest euclidean distance from an  $(x,y,z)$  point to the cow's surface.

**Question:** can we use Random Walk Metropolis to sample from this density?

## Example Random Walk Metropolis: cow revisited

# The problem of tuning the step size in Random Walk Metropolis

The width of the jumping kernel  $\theta_{t+1} \sim J(\theta_{t+1}|\theta_t)$  is a free parameter that needs to be specified.

Choosing an optimal value for this tuning parameter is essential for efficient sampling:

- Too small  $\implies$  the sampler takes a long time to find the typical set (area where most of probability mass lies).
- Too large  $\implies$  the sampler finds the typical set quickly but takes a long time to explore it.

Step size: too small

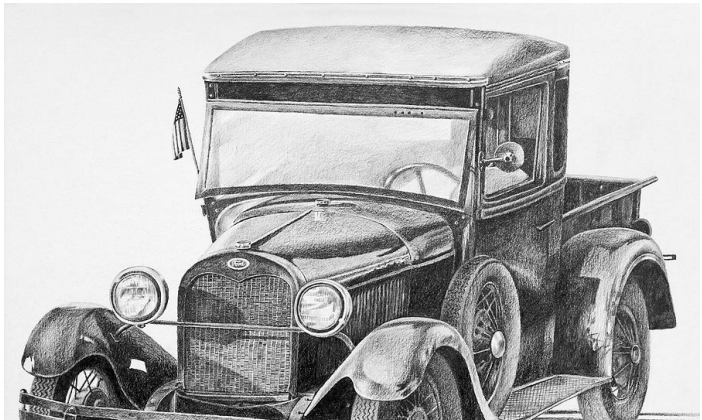
Step size: too big

Step size: just right

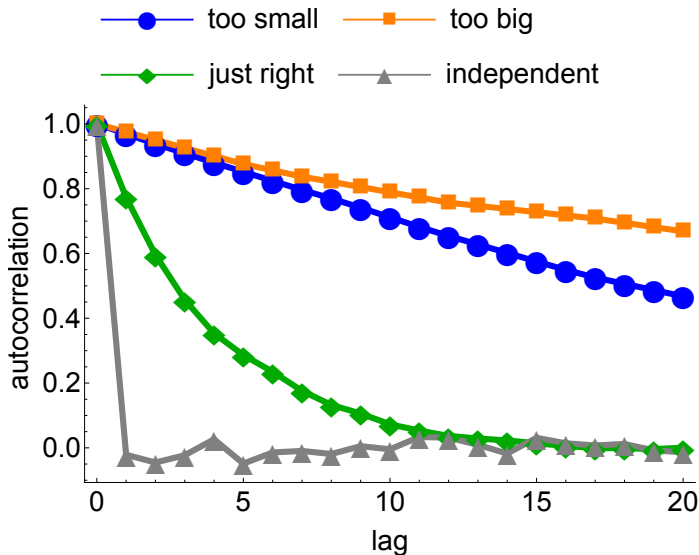


# Autocorrelation across samplers

- Calculate autocorrelation for one of the dimensions of the previous simulations.
- Compare with autocorrelation from an independent sampler.



## Autocorrelation of different step sizes



# Shifting from independent to dependent posterior sampling

Ideally we want to use the powerful WLLN:

For  $X_i \overset{i.i.d}{\sim} f(X)$ .

$$\lim_{n \rightarrow \infty} \frac{X_1 + X_2 + \dots + X_n}{n} = E[X] \quad (4)$$

To evaluate posterior integrals like:

$$\bar{X}_n \approx E[X] = \int_{\mathcal{X}} xf(x)dx \quad (5)$$

# Using dependent sampling to evaluate integrals

- However *independent* sampling from the posterior is *not* generally possible.
- $\implies$  switch to *dependent* sampling.
- And use a less powerful convergence property:  
For  $X_t = \rho X_{t-1} + \epsilon_t$ , and  $|\rho| < 1$  where  $\rho$  measures autocorrelation in sampler.

$$\lim_{t \rightarrow \infty} \frac{X_1 + X_2 + \dots + X_n}{t} = E[X] \quad (6)$$

- The rate of convergence for a dependent sequence like this is **slower** than for the weak law of large numbers  $\implies$  rate of convergence slows as  $\rho \uparrow$  but is always slower than an independent sampler.

# Effective sample size: quantifying the cost of dependent sampling

Intuitively each incremental dependent sample conveys **less information** than an independent sampler.

⇒ quantify this “cost” with the concept of an “effective sample size”.

Defined as:

“The equivalent number of samples for an independent sampler”.

**Question:** how should we design such a metric?

## Effective sample size: designing a metric

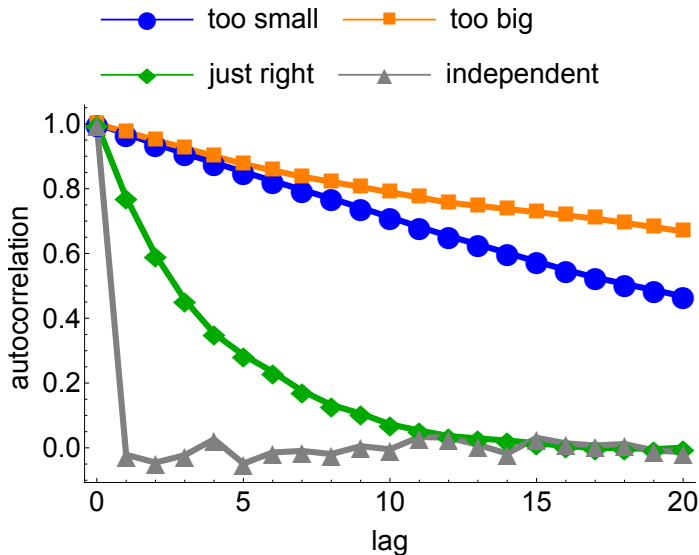
As the dependence  $\rho \uparrow$  the incremental information conveyed by each sample  $\downarrow$

$\implies$  design a measure of effective sample size that reflects this:

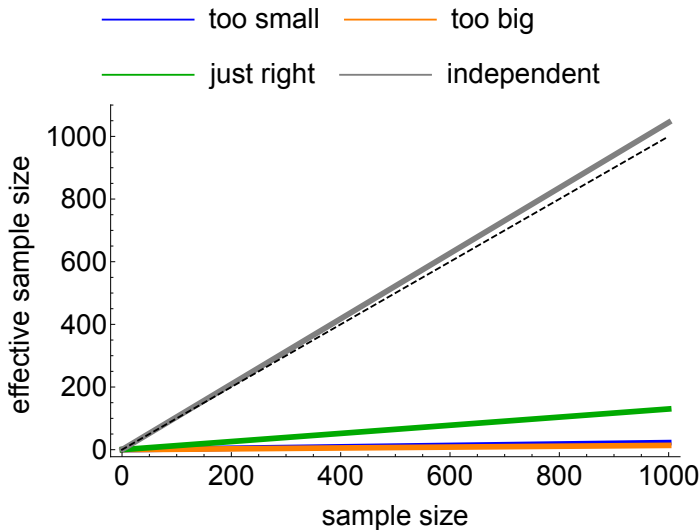
$$ESS(\theta_i) = \frac{mT}{1 + 2 \sum_{\tau=1}^{T_{max}} \rho_{\tau}(\theta_i)} \quad (7)$$

Where  $m$  is the number of chains, and  $T$  is the number of samples *per* chain, and  $\rho_{\tau}$  is the  $\tau$ th order autocorrelation for  $\theta_i$ .

## Autocorrelation of different step sizes

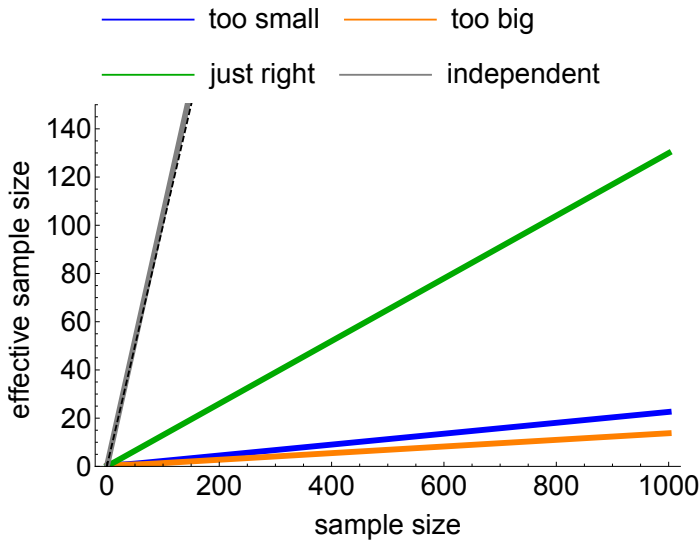


## Effective sample size of different step sizes





## Effective sample size of different step sizes: zoomed



## Effective sample size: summary

- There is a cost to dependent sampling  $\implies$  each incremental sample is less informative than for independent sampling.
- We quantify the cost through the concept of “effective sample size”; the equivalent number of samples for an independent sampler.
- The cost increases along with the dependence of the sampler.
- A good measure of dependence is autocorrelation of a sampler's value.
- Accordingly we create a measure of effective sample size that increases as autocorrelation decreases.

# Why do we need to monitor convergence?

**The problem:** we know the initial proposal distribution (i.e. the distribution governing each chain's start value) is **not** the posterior. However:

- We know that chains will converge **asymptotically** to the posterior; i.e.  $\pi(\theta_t) \rightarrow p(\theta|X)$ .
- However when is  $\pi(\theta_t) \approx p(\theta|X)$ ?

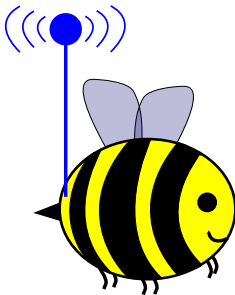
**The solution:**

Use multiple chains starting at random over-dispersed locations in parameter space!

# Judging convergence: Bob's bees

Thought experiment:

- Imagine a house of unknown shape.
- We have an unlimited supply of bees, each equipped with a GPS tracker allowing us to accurately monitor their position.
- **Question:** How can we use these to estimate the shape of the house?



# Judging convergence

Single bee in a house.

# Judging convergence

Multiple bees in a house released in a single room.

# Judging convergence

**Question:** have we converged?

## Judging convergence

Multiple bees in new house released in highly dispersed rooms.



## Judging convergence

Multiple bees in new house released in highly dispersed rooms...much later.

## Judging convergence: summary

- Determining convergence via a single chain is very dangerous, and fraught with the “curse of hindsight” problem.
- Multiple chains reduces the risk of faux-convergence.
- However if we start all chains in same location (for example a mode) then there is a risk of faux-convergence because chains are unable to widely explore parameter space.
- Therefore it is important to use over-dispersed start locations across all chains.
- No convergence monitoring technique is foolproof.
- More chains the better!

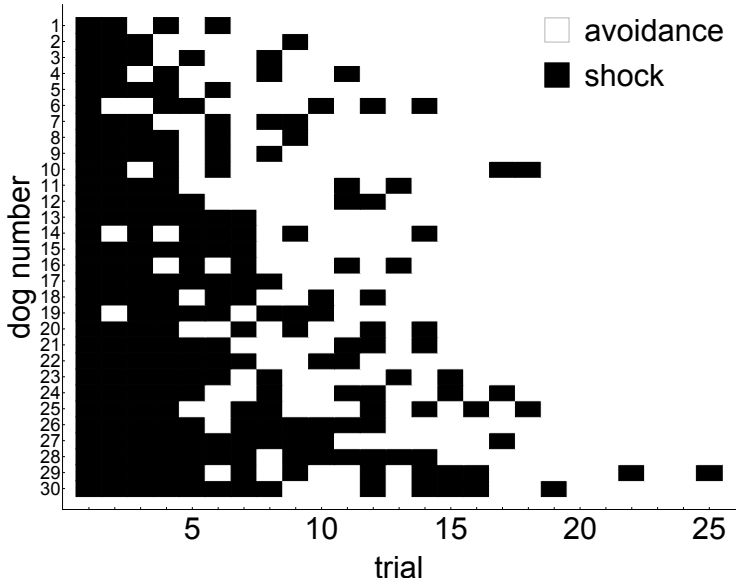
- 1 Recap from last lecture
- 2 Start to finish Bayesian inference
- 3 Metropolis-Hastings
- 4 Gibbs sampling
- 5 Hamiltonian Monte Carlo

# Science gone to the dogs

- Data from a “Solomon-Wynne” experiment on dogs (described in Bush and Mosteller, 1955).
- Dogs were initially confined to a cage which could be **electrified**.
- Before each shock a light was switched on for 10 seconds.
- To avoid the shock the dogs could jump over a low-lying net that separated the electrified cage from another (less-painful) cage.
- Here we analyse the results of 25 trials across 30 dogs; where  $Y_t = 1$  if dog is shocked, and  $Y_t = 0$  if shock is avoided in trial  $t$ .



# Science gone to the dogs: data



## Science gone to the dogs: questions

- Did dogs learn more from successful avoidances or from shocks?
- Can a single stochastic learning model fit data from all the dogs?

## Science gone to the dogs: model

Suppose the probability of shock on trial  $t$ :

$$Pr(Y_t = 1|A, B) = (1 - A)^{X_t}(1 - B)^{t-1-X_t} \quad (8)$$

where:

- $X_t = \sum_{t'=1}^{t-1} Y_{t'}$  is the cumulative number of shocks received before trial  $t$ .
- $0 \leq A \leq 1$  measures incremental learning associated with each additional **shock**.
- $0 \leq B \leq 1$  measures incremental learning associated with each additional **avoidance**.

# Science gone to the dogs: model

Can reformulate the model as a **logistic regression**:

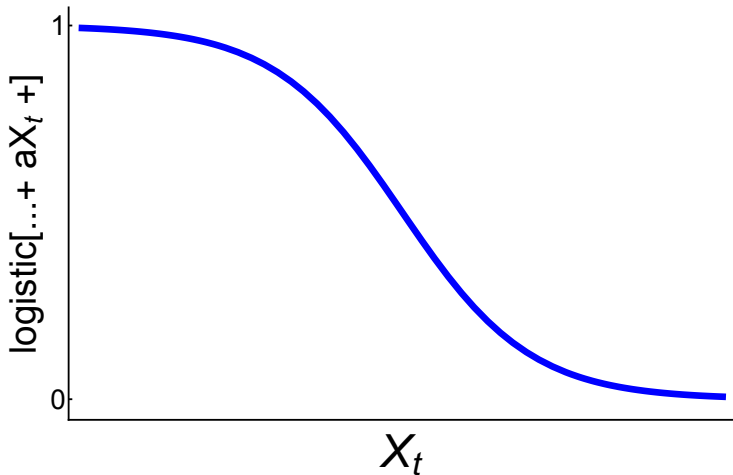
$$Pr(Y_t = 1|a, b, c) = \text{logistic} [c + aX_t + b(t - 1 - X_t)]$$

where:

- As number of shocks **received** increases,  $X_t \uparrow$ , the probability of receiving another shock falls; i.e.  $a < 0$ .
- As number of shocks **avoided** increases,  $(t - 1 - X_t) \uparrow$ , the probability of receiving another shock falls; i.e.  $b < 0$ .



## Science gone to the dogs: model



# Science gone to the dogs: model

Assumptions: conditional on  $(a, b, c, X_t, t)$  the outcome of the next trial is **independent** and **identically-distributed** across dogs and time  $\implies$

- Likelihood is set as a bernoulli-logit distribution:

$$L(a, b, c | Y_t) = [Pr(Y_t = 1 | a, b, c)]^{Y_t} \\ \times [1 - Pr(Y_t = 1 | a, b, c)]^{1 - Y_t}$$

- Where if:

- $Y_t = 1 \implies L(a, b, c | Y_t = 1) = Pr(Y_t = 1 | a, b, c).$
- $Y_t = 0 \implies L(a, b, c | Y_t = 0) = 1 - Pr(Y_t = 1 | a, b, c).$

# Science gone to the dogs: model

**Priors** are final ingredient of the model.

Unfortunately, no conjugate priors for this likelihood!  $\implies$   
choose same diffuse prior for all parameters:

- $a \sim N(0, 10)$ .
- $b \sim N(0, 10)$ .
- $c \sim N(0, 10)$ .

# Science gone to the dogs: model

The numerator of Bayes' rule is given by:

$$p(a, b, c | \mathbf{Y}) \propto \overbrace{\prod_{dogs} \prod_t \text{bernoulli-logit}(a, b, c | Y_{dog,t})}^{\text{likelihood b.c. independence}} \\ \times \underbrace{N(|a)N(|b)N(|c)}_{\text{prior}}$$

# Science gone to the dogs: estimating the posterior

Unfortunately:

- The denominator of Bayes' rule is hard to calculate (Mathematica broke when I tried.)
- Further posterior summaries are as difficult.
- The un-normalised posterior is too complex to generate **independent** samples via Rejection sampling (other methods are also problematic.)

⇒ use **dependent** sampling; i.e. MCMC!

# Science gone to the dogs: coding Random Walk Metropolis

- Parameters are unconstrained because they can be negative or positive.
- $\implies$  can use “vanilla” Metropolis (not Metropolis-Hastings).
- Start 12 chains in over-dispersed locations in  $(a,b,c)$  space; i.e. select an initial location using a multivariate normal with mean 0.
- Select a new location to which to step also using a multivariate normal:

$$\begin{pmatrix} a' \\ b' \\ c' \end{pmatrix} \sim N \left[ \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \Sigma \right]$$

# Science gone to the dogs: coding Random Walk Metropolis

- After proposed  $(a', b', c')$  calculate:

$$r = \frac{\text{likelihood}(a', b', c') \times \text{prior}(a', b', c')}{\text{likelihood}(a, b, c) \times \text{prior}(a, b, c)} \quad (9)$$

- Generate  $u \sim \text{uniform}(0, 1)$ .
- If  $r > u \implies$  move to  $(a', b', c')$ .
- Otherwise stay at  $(a, b, c)$ .

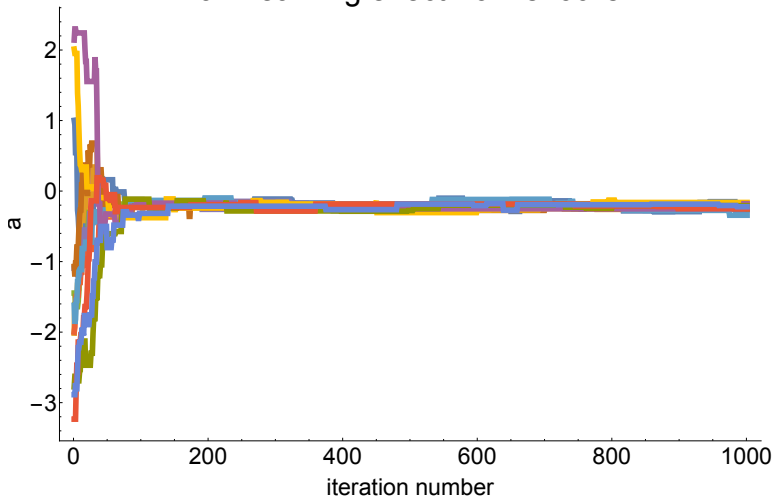
# Science gone to the dogs: how long to run the chains?

- We know that chains will converge **asymptotically** to the posterior; i.e.  $\pi(\theta_t) \rightarrow p(\theta|X)$ .
- However when is  $\pi(\theta_t) \approx p(\theta|X)$ ?
- $\implies$  calculate  $\hat{R}$ !



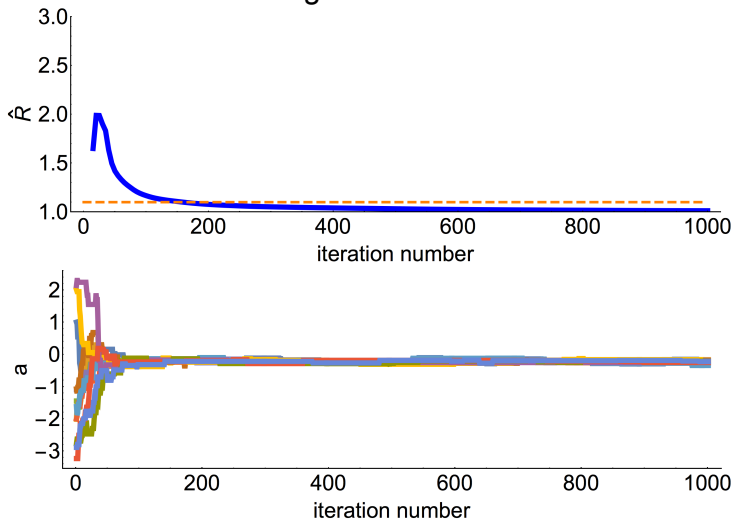
## Science gone to the dogs: chain convergence

a – learning effect from shocks



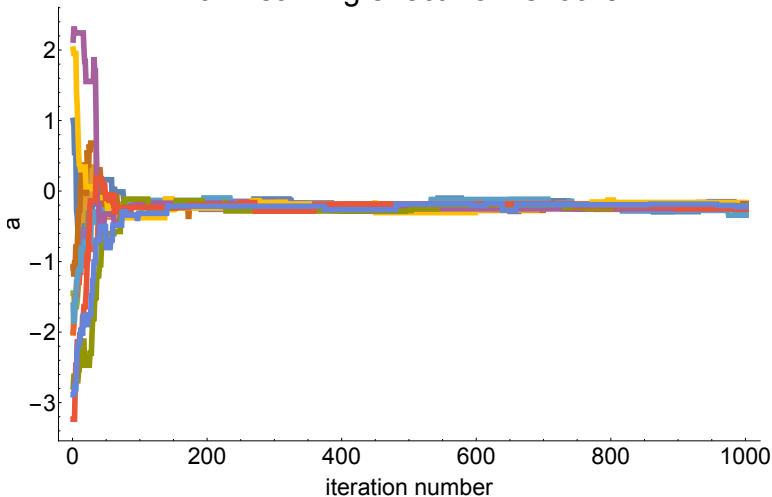
# Science gone to the dogs: $\hat{R}$

a – learning effect from shocks

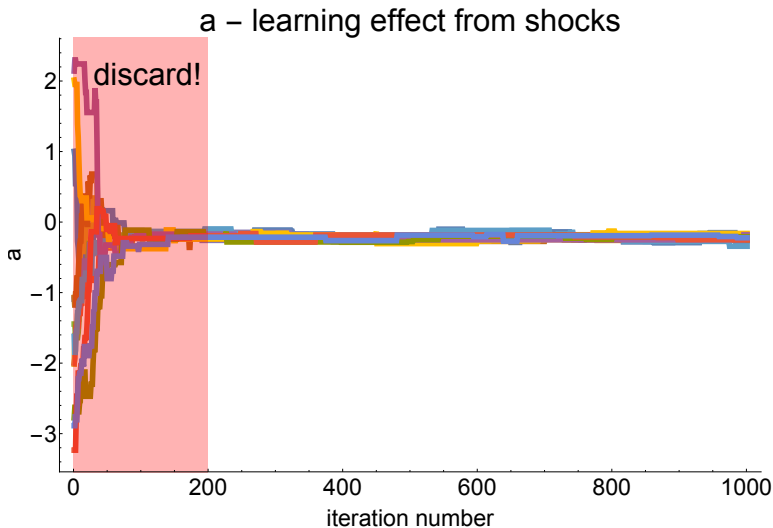


## Science gone to the dogs: MCMC warm-up

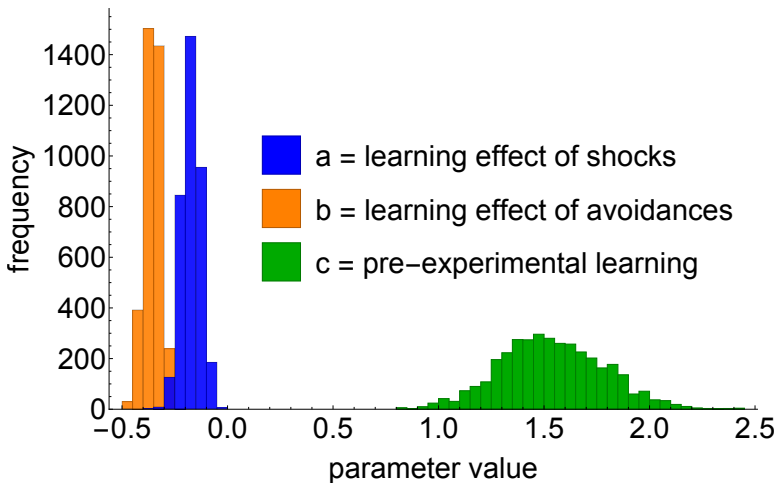
a – learning effect from shocks



## Science gone to the dogs: MCMC warm-up



## Science gone to the dogs: posterior summaries



## Science gone to the dogs: tentative conclusions

- The magnitude of the “avoidance” effect is lower than the “shock” effect  $\implies$  dogs learn more from successful avoidances than they do from shocks.
- However, are we right in assuming homogeneous coefficients across all dogs?

$\implies$  posterior predictive checks!

# Science gone to the dogs: posterior predictive distribution

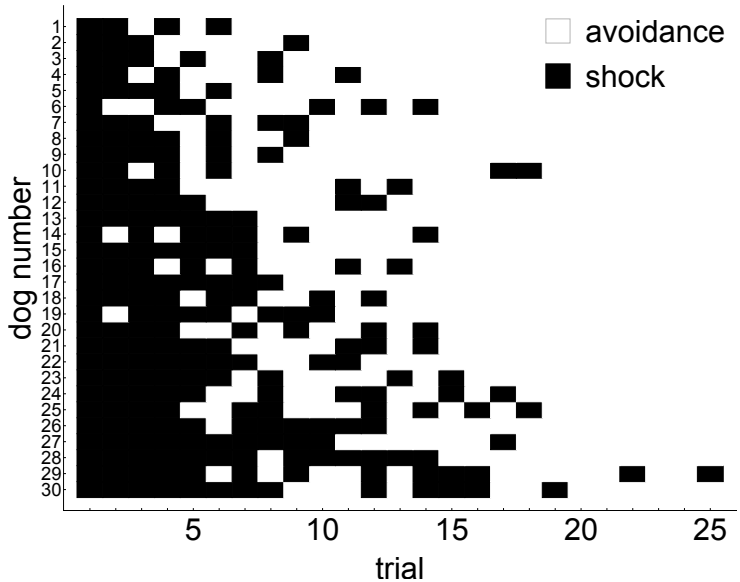
- To do posterior predictive checks we need to sample from the posterior predictive distribution.
- In our example this is not trivial:
  - First sample  $(a, b, c)$  from the posterior distribution (here itself a list of samples).
  - Then sample  $Y_{dog,t}$  - whether “dog” receives a shock on trial  $t$ .

$$Y_{dog,t} \sim \text{bernoulli-logit}[c + aX_{dog,t} + b(t - 1 - X_{dog,t})] \quad (10)$$

where  $X_{dog,t}$  is the cumulative number of shocks received **before** time  $t$ .

- Update  $X_{dog,t+1} = X_{dog,t} + Y_{dog,t}$ .
- Repeat for  $Y_{dog,t+1}$ .

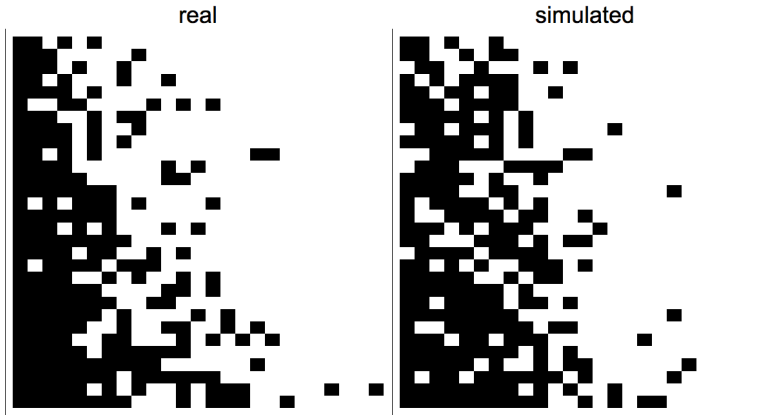
# Science gone to the dogs: real data





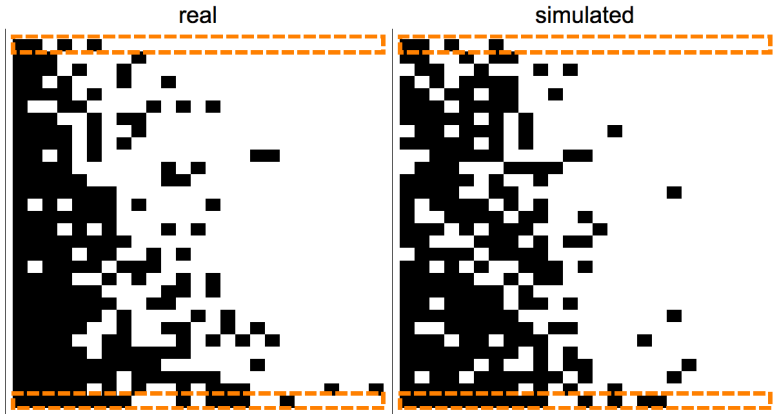
# Science gone to the dogs: posterior predictive checks

A posterior predictive simulation.



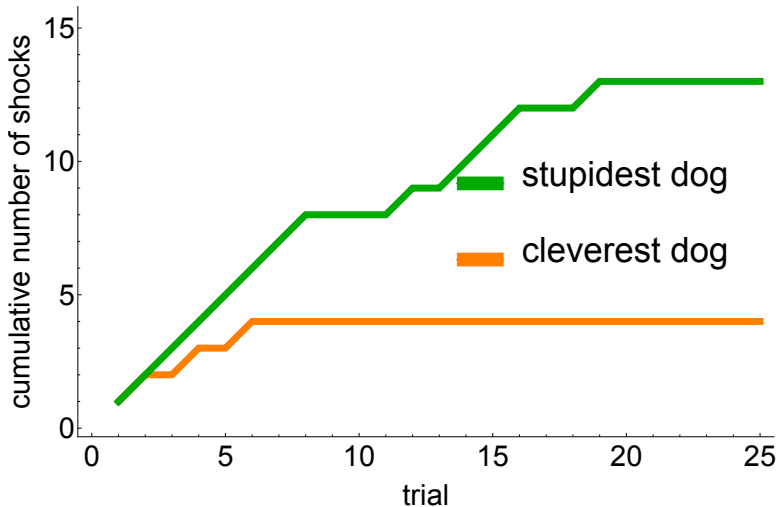
# Science gone to the dogs: posterior predictive checks

Select best and worst dogs.



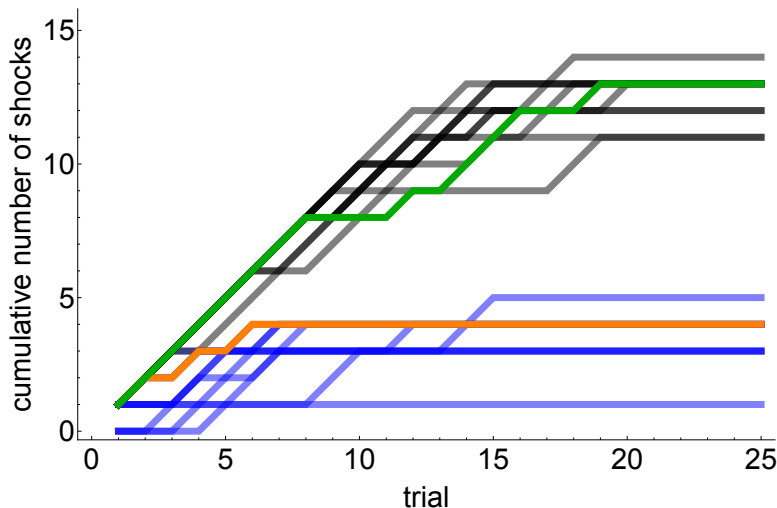
# Science gone to the dogs: posterior predictive checks

Cumulate shocks for the best and worst dogs.



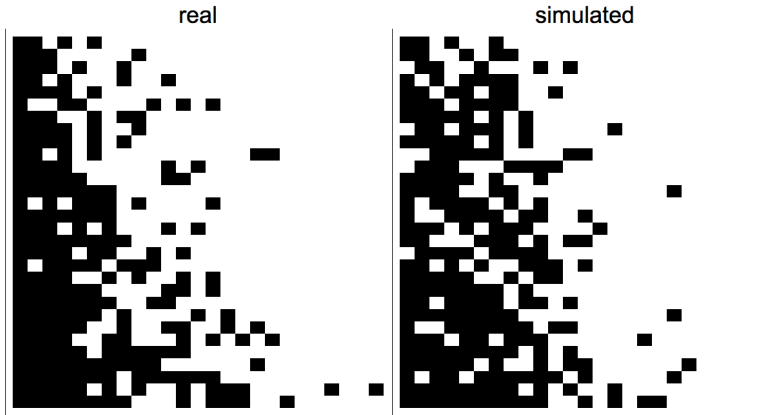
# Science gone to the dogs: posterior predictive checks

Versus 10 simulated datasets.



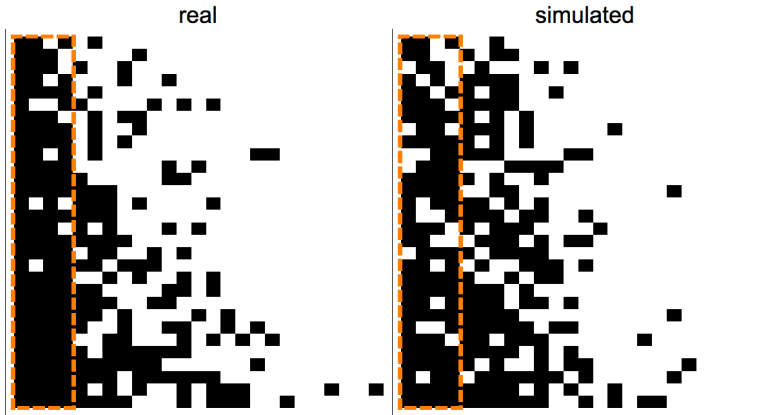
# Science gone to the dogs: posterior predictive checks

A posterior predictive simulation.



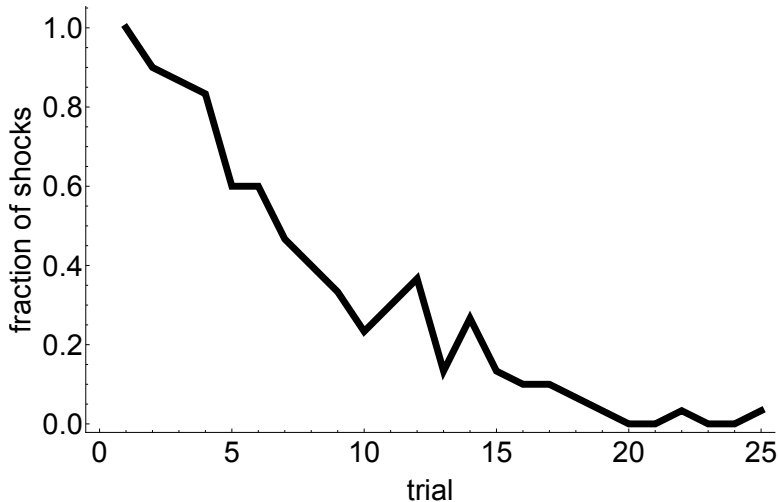
# Science gone to the dogs: posterior predictive checks

Simulated data overstates rate of learning for early trials.



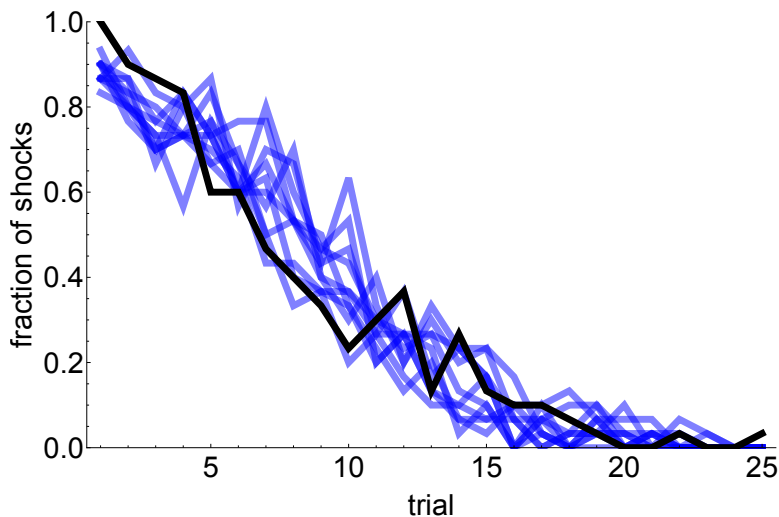
# Science gone to the dogs: posterior predictive checks

Fraction of dogs shocked by trial number for real data.



# Science gone to the dogs: posterior predictive checks

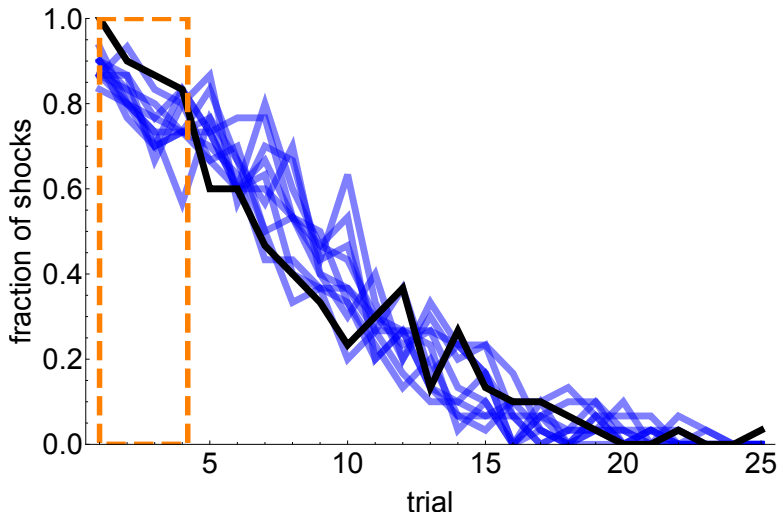
Versus 10 simulated datasets.





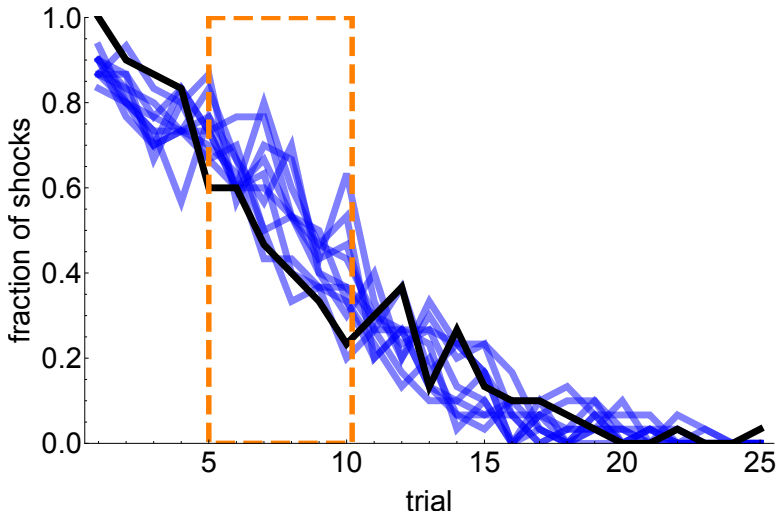
# Science gone to the dogs: posterior predictive checks

Under-prediction for early trials.



# Science gone to the dogs: posterior predictive checks

Some over-prediction for trials 5-10.



# Science gone to the dogs: logistic regression appraisal

Posterior predictive checks show:

- The between-dog variation is replicated in simulated datasets  $\implies$  homogeneous  $(a, b, c)$  look fine.
- Simulated data indicates a fraction  $> 0$  of avoided shocks for the first trials  $\implies$  not seen in real data.
- There are persistent runs of under-prediction and over-prediction in the simulated data  $\implies$  important because we want model to represent the learning process.

In conclusion: model not terrible but can we do better?

# Science gone to the dogs: exponential model introduction

Existing model:

$$Pr(Y_t = 1|a, b, c) = \text{logistic}[c + aX_t + b(t - 1 - X_t)]$$

New model (same Bernoulli likelihood, just different “link”):

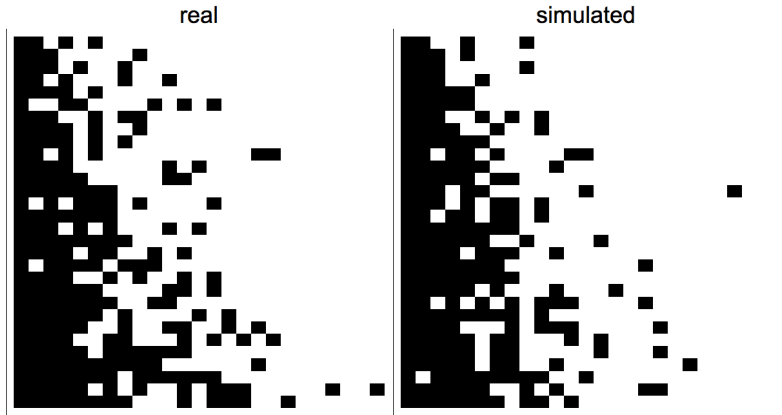
$$Pr(Y_t = 1|a, b) = \exp[aX_t + b(t - 1 - X_t)]$$

$\implies$  naturally forces  $Pr(Y_t = 1|a, b) = 1$ .

**Question: how does the new model fare?**

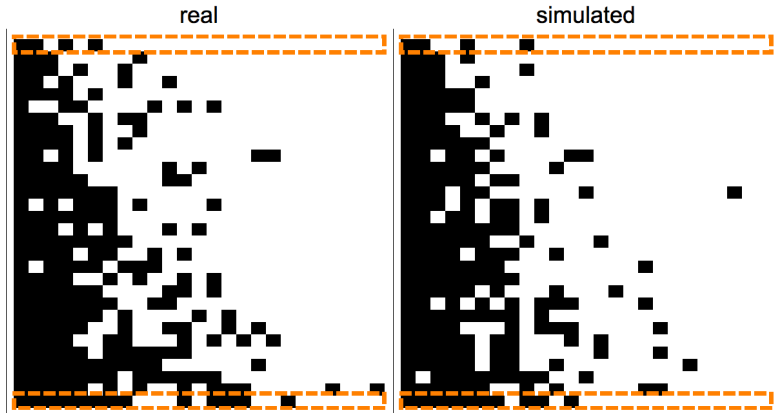
# Science gone to the dogs: posterior predictive checks for exponential model

A posterior predictive simulation from new model.



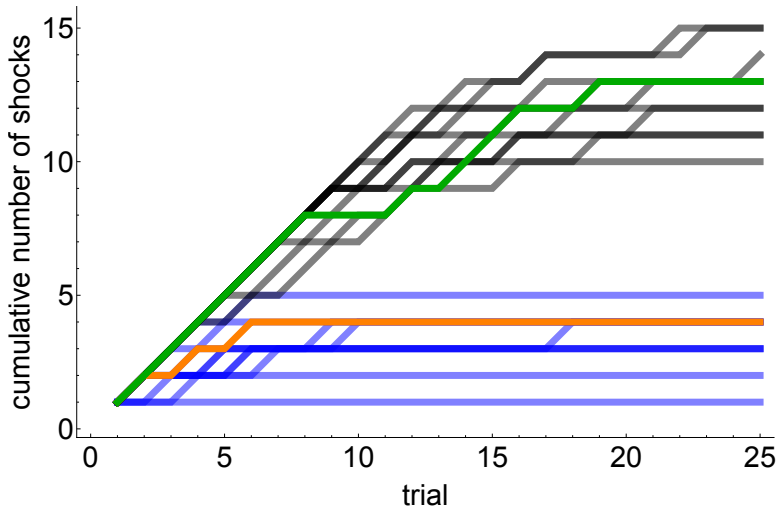
# Science gone to the dogs: posterior predictive checks for exponential model

Again select best and worst dogs.



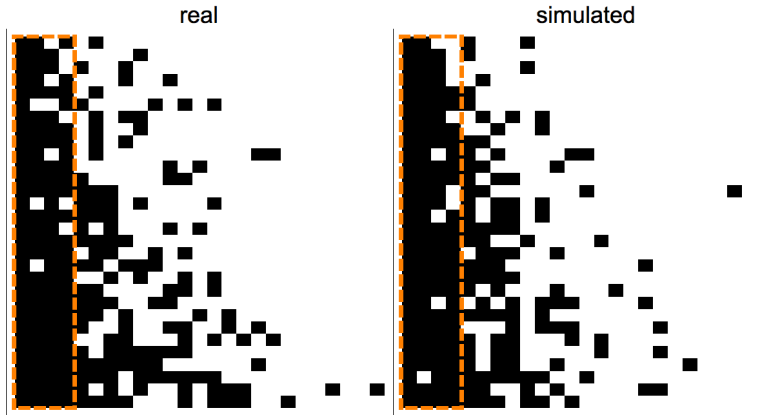
# Science gone to the dogs: posterior predictive checks for exponential model

Still good.



# Science gone to the dogs: posterior predictive checks for exponential model

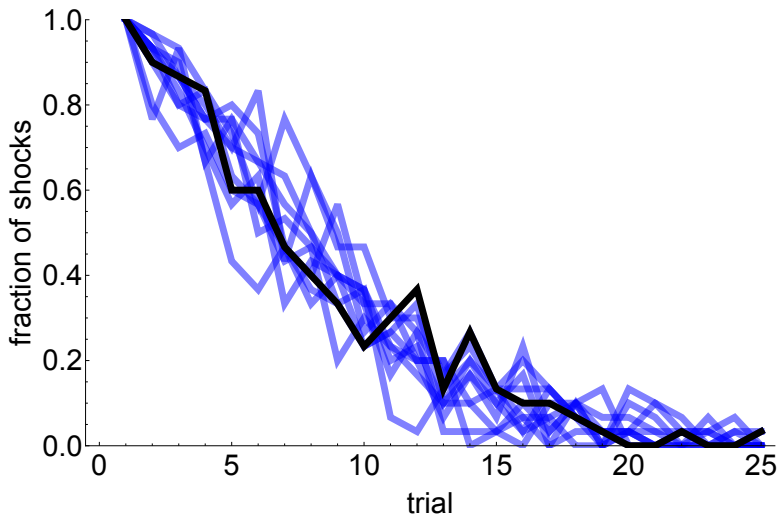
Look at early trial performance.





# Science gone to the dogs: posterior predictive checks for exponential model

Much better.

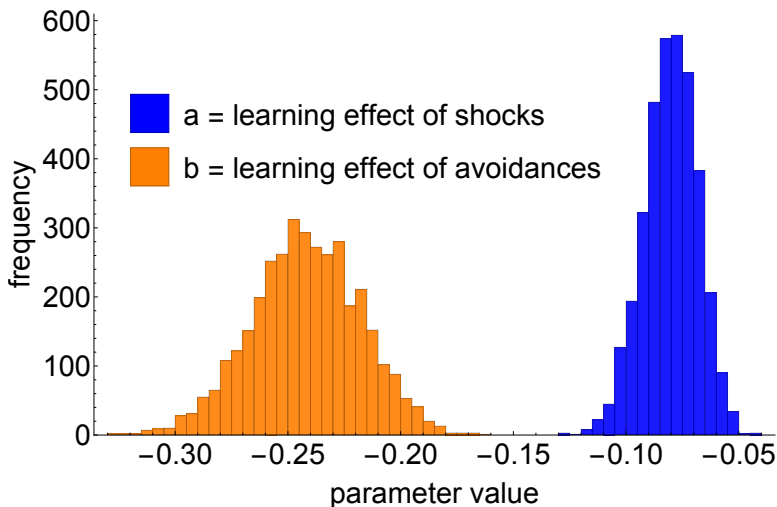


# Science gone to the dogs: exponential regression model appraisal

In summary:

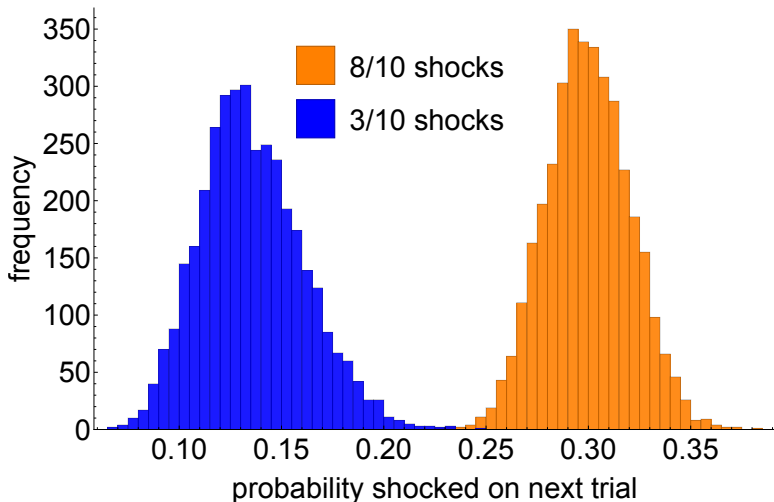
- Simpler model with homogeneous parameters ( $a, b$ ) also able to account for between-dog variation.
- Automatically means all dogs are shocked on first trial.
- New model does not give runs of over-prediction or under-prediction.
- Interrogate all model's assumptions; a shift from “logistic” to “log” link is subtle but important.

# Science gone to the dogs: exponential model results



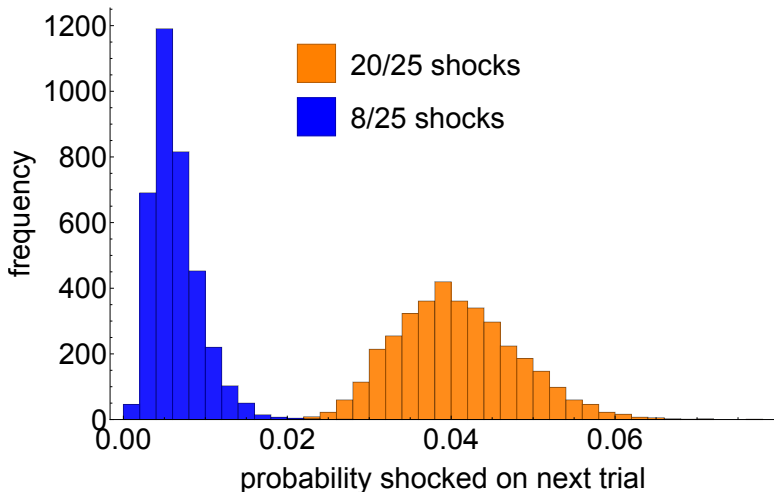
# Science gone to the dogs: exponential model results

Instead show for  $t = 10$  the probability of being shocked on next trial for two different histories.



# Science gone to the dogs: exponential model results

And for  $t = 25$  the probability of being shocked on next trial for two different histories.



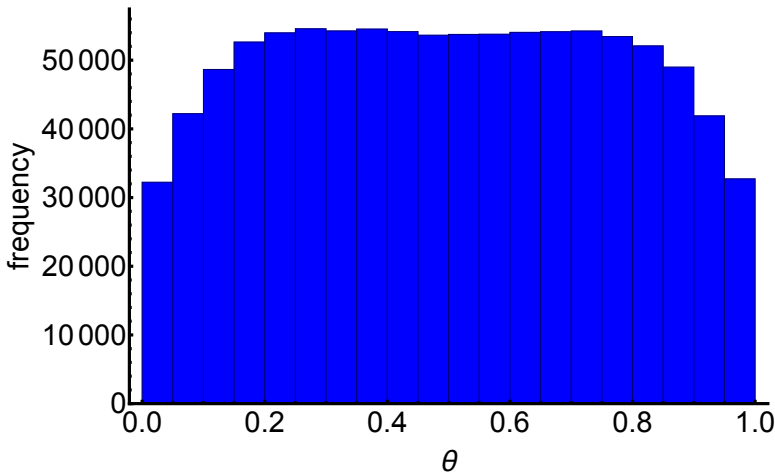
- 1 Recap from last lecture
- 2 Start to finish Bayesian inference
- 3 Metropolis-Hastings**
- 4 Gibbs sampling
- 5 Hamiltonian Monte Carlo

# Constrained parameters

- Suppose one of your parameters is constrained.
- As an example consider a likelihood  $X \sim \text{Binomial}(n, \theta)$ , where  $0 \leq \theta \leq 1$ .
- Consider the following jumping (rejection) routine:
  - 1 Propose  $\theta_{t+1} \sim N(\theta_t, 0.1)$ ; i.e. centred on current position.
  - 2 If  $\theta_{t+1} < 0$  or  $\theta_{t+1} > 1$  reject  $\theta_{t+1}$ , and propose new  $\theta_{t+1} \sim N(\theta_t, 0.1)$ .
  - 3 Otherwise accept  $\theta_{t+1}$ .
- **Question:** does this stepping routine propose  $\theta_{t+1}$  evenly across  $(0,1)$ ?

## Constrained parameters

Do 1 million steps (always accepting) of this routine. **Answer:**  
no!  $\Rightarrow$  lower sampling weight nearer 0 or 1!





# Constrained parameters: problem and solution

## The problem:

- If we use symmetric jumping distribution we get bias away from boundaries.
- For a two-sided boundary we can rectify things by using modular arithmetic; i.e. if we fall off one side we enter the other side.
- For a single boundary this workaround doesn't work.
- An example of a single boundary parameter is  $\sigma > 0$  for  $X \sim N(\mu, \sigma)$ .

**The solution:** use asymmetric proposal distribution!

# Asymmetric jumping distribution

Log-normal example.

# Metropolis-Hastings

When we use an asymmetric jumping distribution the ratio from the “vanilla” Metropolis rule:

$$r = \frac{\text{likelihood}(\theta_{t+1}) \times \text{prior}(\theta_{t+1})}{\text{likelihood}(\theta_t) \times \text{prior}(\theta_t)} \quad (11)$$

Doesn't work! We don't get convergence to the posterior. We need to correct for the asymmetric jumping in  $r$ . Instead use:

$$r' = \frac{\text{likelihood}(\theta_{t+1}) \times \text{prior}(\theta_{t+1})}{\text{likelihood}(\theta_t) \times \text{prior}(\theta_t)} \times \frac{J(\theta_t|\theta_{t+1})}{J(\theta_{t+1}|\theta_t)} \quad (12)$$

Everything else remains the same.

# Metropolis-Hastings summary

- For unconstrained parameters we are free to use symmetric jumping kernels.
- However for constrained parameters we are forced to break this symmetry.
- If we use “symmetric” jumping rules (with rejection sampling)  $\implies$  we get under-sampling near boundaries.
- This under-sampling biases our sampling distribution  $\neq$  posterior.
- Better to use asymmetric jumping kernel with support over “allowed” values.
- To use an asymmetric jumping kernel we must correct the accept/reject ratio  $r$  to account for this  $\implies$  get convergence to posterior.

- 1 Recap from last lecture
- 2 Start to finish Bayesian inference
- 3 Metropolis-Hastings
- 4 Gibbs sampling**
- 5 Hamiltonian Monte Carlo

# Inefficient exploration of the typical set by Random Walk Metropolis

# Inefficient exploration of the typical set by Random Walk Metropolis

Even if the step size for Random Walk Metropolis is optimal  
 $\implies$  suboptimal exploration due to large number of rejected steps.



# Defining the Gibbs sampler

For a parameter vector:  $\theta = (\theta_1, \theta_2, \theta_3)$ :

- Select a random starting location:  $(\theta_1^0, \theta_2^0, \theta_3^0)$ , along the same lines as for Random Walk Metropolis.
- For each iteration  $t = 1, \dots, T$  do:
  - ① Select a random parameter update ordering, for example  $(\theta_3, \theta_2, \theta_1)$ .
  - ② Independently sample from the conditional posterior for each parameter in order using the most up-to-date parameters.



## Defining the Gibbs sampler

First we sample:

$$\theta_3^1 \sim p(\theta_3 | \theta_2^0, \theta_1^0) \quad (13)$$

Then conditional on freshly-sampled  $\theta_3^1$ :

$$\theta_2^1 \sim p(\theta_2 | \theta_1^0, \theta_3^1) \quad (14)$$

Then conditional on freshly-sampled  $\theta_3^1$  and  $\theta_2^1$ :

$$\theta_1^1 \sim p(\theta_1 | \theta_2^1, \theta_3^1) \quad (15)$$

# Defining the Gibbs sampler

**Important:** in Gibbs sampling there is no rejection of steps  
⇒ unlike Random Walk Metropolis!

One of the reasons Gibbs can be more efficient.

## Example application of Gibbs sampling: speed of motion of neighbouring birds in a flock

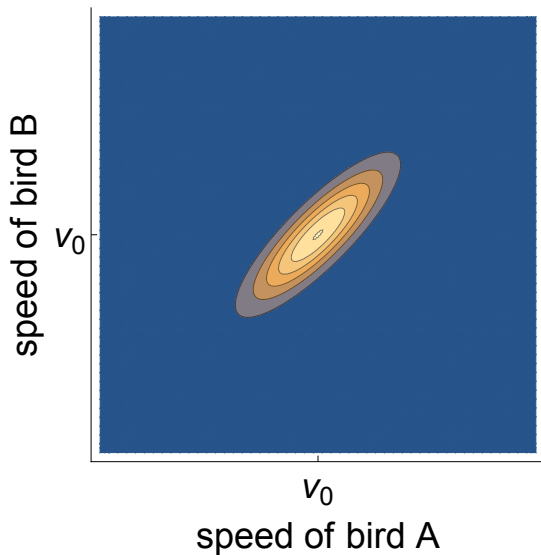
Suppose we record the speed of bird A ( $v_A$ ) and bird B ( $v_B$ ) in a flock along a particular axis.

Based on observations we find that the joint posterior distribution over speeds is a multivariate normal distribution:

$$\begin{pmatrix} v_A \\ v_B \end{pmatrix} \sim N \left[ \begin{pmatrix} v_0 \\ v_0 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \right]$$

Of course here we have an analytic expression for the posterior distribution, but this example illustrates how the method works for more general problems.

# Example application of Gibbs sampling: speed of motion of neighbouring birds in a flock



## Finding the conditional distributions

In most circumstances we cannot find the conditional distributions however here it is possible.

If we knew  $v_B$ :

$$v_A \sim N(v_0 + \rho(v_B - v_0), 1 - \rho^2) \quad (16)$$

Alternatively, if we knew  $v_A$ :

$$v_B \sim N(v_0 + \rho(v_A - v_0), 1 - \rho^2) \quad (17)$$

Use Gibbs sampling to conditionally sample:  $v_A|v_B$  then  $v_B|v_A$ .

**Remember:** in Gibbs sampling we accept **all** steps unlike Random Walk Metropolis.

# Gibbs sampling the posterior distribution over birds' speeds

# Comparing Random Walk Metropolis with Gibbs

# Highly correlated parameters: problems with Random Walk Metropolis and Gibbs

- Gibbs performs well on this simple problem.
- However if we increase the posterior correlation between parameters, how does each sampler fare?



Highly correlated parameters: both poor at finding the typical set

Highly correlated parameters: also both poor at exploring the typical set

## Other problems with Gibbs

- Requires that the conditional distributions can be derived and sampled from.
- Relies on us “knowing” a reasonable amount of the maths behind each problem.
- Maths is hard and we would like to avoid it if possible!

Often we can only sample from the conditional distributions for a few parameters  $\implies$  use Random Walk Metropolis for others (essentially the method used by BUGS and JAGS.)

## Gibbs sampling: summary

- Gibbs sampling works by cycling through each parameter dimension, and sampling from the distribution conditional on all other parameters.
- (If “joint-conditional” distributions of the form  $p(\theta_1, \theta_2 | \theta_3)$  can be sampled, then this is a more efficient form of Gibbs.)
- Depends on us knowing the conditional distribution for each parameter  $\implies$  in majority of circumstances not possible.
- Can be more efficient than Random Walk Metropolis but not a panacea.

# Gibbs sampling: summary

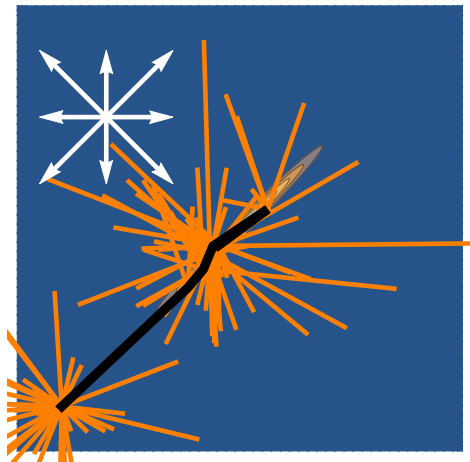
“If I had a nickel for every time someone had asked for help with slowly converging MCMC and the answer had been to stop using Gibbs, I would be rich.”

— William Shakespeare / Charles Geyer.

- 1 Recap from last lecture
- 2 Start to finish Bayesian inference
- 3 Metropolis-Hastings
- 4 Gibbs sampling
- 5 Hamiltonian Monte Carlo

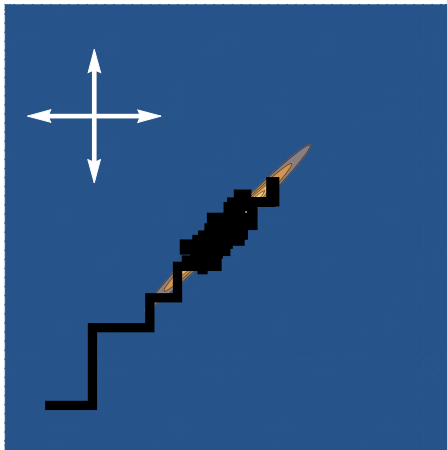
# What are the problems with Random Walk Metropolis and Gibbs?

## Random Walk Metropolis



# What are the problems with Random Walk Metropolis and Gibbs?

Gibbs





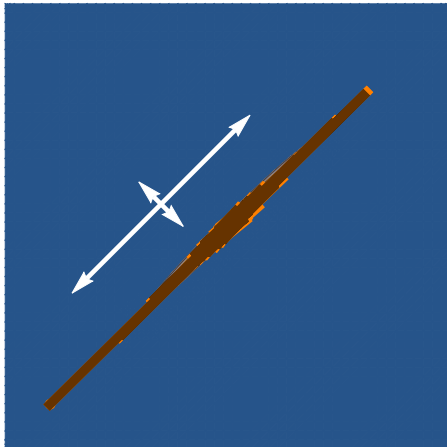
# What are the problems with Random Walk Metropolis and Gibbs?

**Answer:** both Random Walk Metropolis and Gibbs ignore the posterior geometry!

# What are the problems with Random Walk Metropolis and Gibbs?

What we would prefer is movements along diagonal.

HMC

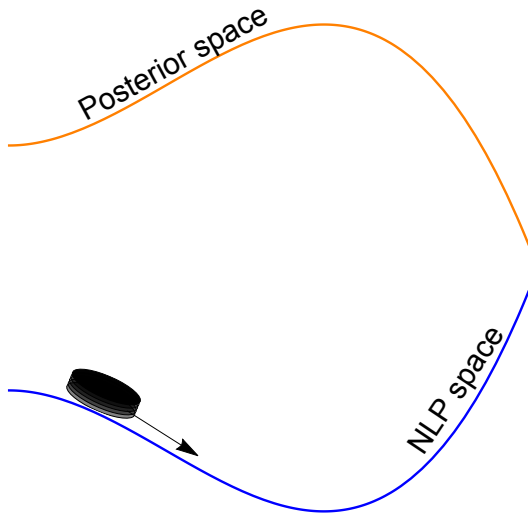


Enter the third murderer: Hamiltonian Monte Carlo

# Introduction to Hamiltonian Monte Carlo

- Assume a space related to posterior space (more on this next) can be thought of as a landscape.
- Imagine an ice puck moving over the frictionless surface of this terrain.
- At defined time points we measure the location of the puck, and instantaneously give the puck a shove in a random direction.
- The locations traced out by the puck represent proposed steps from our sampler.
- Based on the height of the posterior and momentum of the puck we accept/reject steps.

Why does this physical analogy help us?



## Why does this physical analogy help us?

- Allow the potential energy of the puck to be determined partly by the posterior density.
- $\implies$  puck will move in the “natural” directions dictated by the posterior geometry.
- And will visit areas of low NLP  $\implies$  high posterior density.

# An introduction to Hamiltonian Monte Carlo

Questions we need to answer:

- ① What is the space over which the puck slides?
- ② How do we solve for the motion of the puck?
- ③ How should we “shove” the puck?
- ④ What is our new accept/reject rule?

# The space of HMC: physical analogy made concrete

- In statistical mechanics/information theory we explore systems in thermal equilibrium whose energy “state” cannot be directly observed.
- Instead we associate a probability with each energy level,  $E$ :

$$p(E) \propto \exp\left(-\frac{E}{T}\right) \quad (18)$$

where  $T$  is the “temperature” of the system.

- Note to physicists: I have assumed units where  $k_B = 1$ .
- In HMC we convert our statistical problem into a physical one, by assuming the “puck” has energy that is determined partly by the posterior density.



# The space of HMC: physical analogy made concrete

Assume that our ice puck has a location  $\theta$  and momentum  $k$ , with an associated energy state  $E(\theta, k)$ . If we assume  $T = 1$  the probability distribution over states:

$$p(\theta, k) \propto \exp(-E(\theta, k)) \quad (19)$$

Where the energy is the sum of:

$$E(\theta, k) = \underbrace{U(\theta)}_{\text{potential energy}} + \underbrace{KE(k)}_{\text{kinetic energy}} \quad (20)$$

# The Hamiltonian

Typically use the notation  $H() = E()$  because in classical mechanics the total energy of the system is known as the *Hamiltonian*.

For the kinetic energy in  $q$  dimensions we use (mass assumed to be 1):

$$KE(k) = \sum_{i=1}^q \frac{k_i^2}{2} \quad (21)$$

For the potential energy we use the **negative** log of the un-normalised posterior:

$$U(\theta) = -\log(p(X|\theta)p(\theta)) \quad (22)$$

## What is the space over which the puck slides?

$$U(\theta) = -\log(p(X|\theta)p(\theta)) \quad (23)$$

We choose this energy so that  $p(X|\theta)p(\theta) = \exp(-U(\theta))$ ; i.e. a probability is the negative exponential of an energy.

Here we call  $U(\theta)$  *negative log posterior space* (NLP).

- Essentially the inverse of posterior space, so that lows (highs) in NLP space correspond to highs (lows) in posterior space.
- Simulate the motion of the puck under this potential.

# How do we solve for the motion of the puck?

Classical mechanics tells us that the position and momentum of the puck evolve according to:

$$\begin{aligned}\frac{d\theta_i}{dt} &= \frac{\partial H}{\partial k_i} \\ \frac{dk_i}{dt} &= -\frac{\partial H}{\partial \theta_i}\end{aligned}$$

The trouble is these are too difficult to solve exactly in most cases

⇒ use an approximate numerical method (e.g. Leap-Frog symplectic integrator.)

**Note:** requires us to be able to evaluate derivatives of posterior  
⇒ HMC tricky where cost of evaluating likelihood is high.

# How should we “shove” the puck?

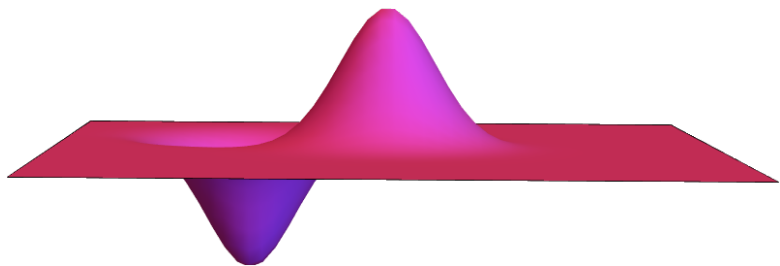
At the start of each step we generate a random initial momentum for the puck. For example:

$$k \sim N(0, \Sigma) \quad (24)$$

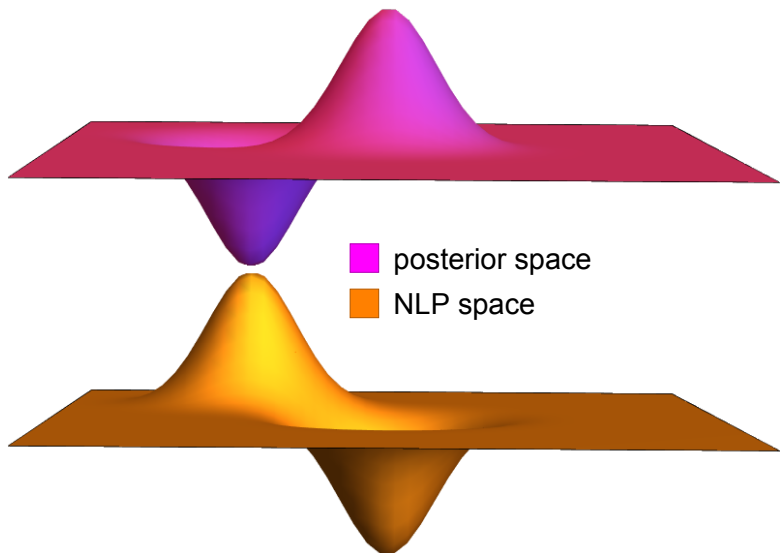
**Question:** why do we give the puck a non-zero starting momentum?

**Answer:** to allow it to climb to areas of high NLP  $\implies$  low posterior density.

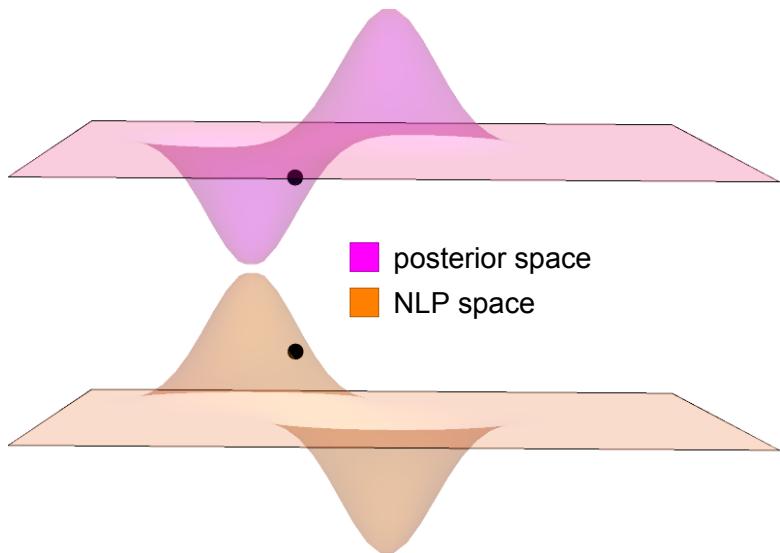
Simulating the puck's motion in NLP space: start with a posterior



# Simulating the puck's motion in NLP space: find NLP space



Simulating the puck's motion in NLP space:  
consider a point in posterior space





The path traced out for 100 different shoves from same distribution

## What is our new accept/reject rule?

After a specified length of time we stop simulating the puck and record its:

- Position; i.e. its current value of  $\theta$ .
- Momentum; i.e. its current value of  $k$ .

Both of these feed into an accept/reject rule that ensure that we get asymptotic convergence to the posterior.

## HMC: summary

- Both Random Walk Metropolis and Gibbs sampling ignore the posterior geometry when deciding where to step next  $\implies$  inefficient exploration of posterior space.
- HMC avoids inefficiency by allowing the next proposal location to be partly determined by the shape of the posterior.
- Explicitly at each step of HMC we simulate the movement of a puck over a frictionless surface that is given an initial “shove”.
- Potential energy determined by NLP  $\implies$  we tend to move to areas of low NLP/high posterior density.
- HMC is more complex in nature than Gibbs or Random Walk Metropolis  $\implies$  use Stan!

## Lecture summary

- MCMC can be used to sample from posteriors where we have no chance of finding exact answers.
- It is essential to start multiple chains in dispersed locations to judge convergence.
- Random Walk Metropolis can be inefficient to explore posterior space.
- Gibbs can be faster than RWM although requires that we can calculate exact conditionals and sample from them  $\implies$  often not possible.
- Both RWM and Gibbs struggle with correlated parameters because they ignore posterior geometry when stepping.
- Hamiltonian Monte Carlo accounts for posterior geometry when deciding on steps but is more complex to implement  $\implies$  use Stan.

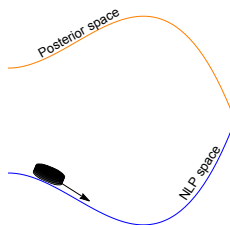
# Reading list

Only big chunks this week.

- Chapters 11 (basic MCMC) and 12 (advanced MCMC) in “Bayesian data analysis”, by Gelman et al. (2014), 3rd edition.
- Chapters 7 (MCMC) and 14 (HMC and Stan) in “Doing Bayesian data analysis”, by Kruschke (2015), 2nd edition.
- Chapter 8 (MCMC) in “Statistical Rethinking”, by McElreath (2016).
- Chapter 5 (HMC) by Neal, in “Handbook of Markov Chain Monte Carlo”, edited by Brooks et al. (2011).

# Not sure I understand?

Hamiltonian Monte Carlo.



Hamilton in Monte Carlo.



# Derivation of effective sample size measure

Estimate the mean by averaging:

$$\hat{\mu} = \frac{1}{T} \sum_{t=1}^T \theta_t \quad (25)$$

Now considering the variance of this:

$$\begin{aligned} T \text{var}(\hat{\mu}) &= \text{var}\left(\sum_{t=1}^T \theta_t\right) \\ &= \sum_{t=1}^T \text{var}(\theta_t | \text{data}) + \sum_{t=1}^T \sum_{\tau \geq 1} \text{cov}(\theta_t, \theta_{t-\tau}) \end{aligned}$$

# Derivation of effective sample size measure

Now using AR1 process definition across  $m$  chains.

$$\lim_{T \rightarrow \infty} mT \text{var}(\hat{\mu}) = \left( 1 + 2 \sum_{\tau=1}^{\infty} \rho_{\tau} \right) \text{var}(\theta | \text{data})$$

Now defining effective sample size:

$$\lim_{T \rightarrow \infty} n_{\text{eff}} \text{var}(\hat{\mu}) = \text{var}(\theta | \text{data}) \quad (26)$$

Rearranging:

$$n_{\text{eff}} = \frac{mT}{1 + 2 \sum_{\tau=1}^{\infty} \rho_{\tau}} \quad (27)$$