Bayesian Computation and Intro to Markov chains

May 6, 2025

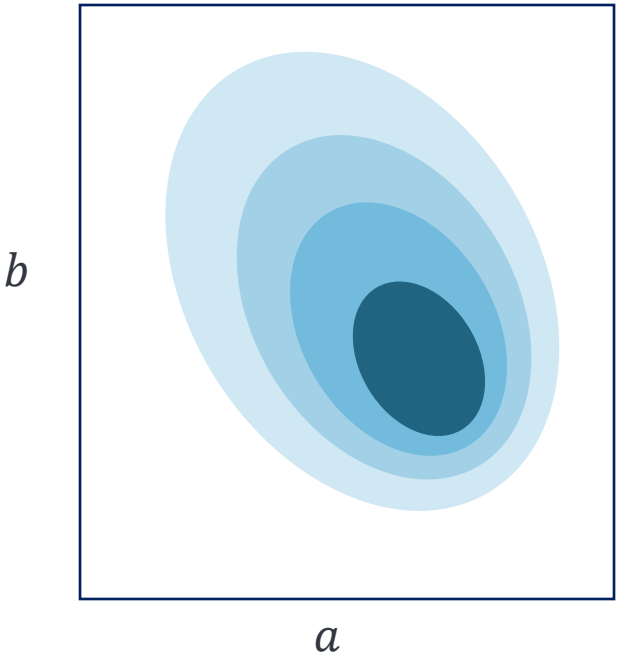
Prof. Gwendolyn Eadie

Bayesian Inference

- Parameters are not fixed
- The quantity of interest is the **distribution of the parameters**
- Hence, we want to calculate exactly or, if we can't calculate exactly then estimate the ***posterior distribution***

$$p(\theta|y) \propto p(y|\theta)p(\theta)$$

- Estimating the posterior may be done using computational methods
  - Setting up a grid of θ's and evaluating the posterior at every point given the data ($$$)
  - Drawing samples from the posterior using Markov Chain Monte Carlo, Hamilton Monte Carlo, Gibbs Sampling, or some other sampling method
  - Variational Bayes
  - Other methods



Sampling from the posterior distribution (Ch 3, McElreath)

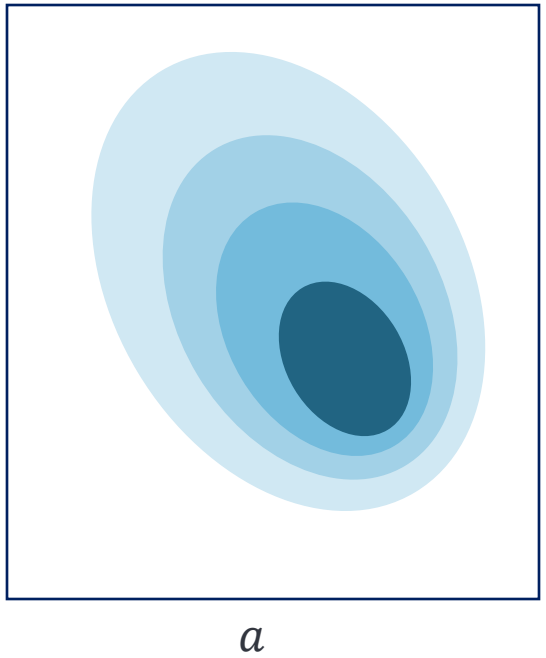
- Our example with m&m's was nice and analytic
  - We got p(theta|y) no problem
- In many contexts, the posterior distribution p(theta|y) is intractable, or computationally expensive to calculate directly
- One alternative approach is grid-approximation
- **Better yet, *draw samples* from a distribution proportional to the posterior**
- Once you have samples, it's easy to calculate many quantities of interest
  - Summary statistics
  - Estimating the probability within some interval (e.g., credible intervals)
  - Quantities of scientific interest that depend on the parameters

Bayesian Computation: sampling from a distribution

- Grid approximation
  - Computationally expensive as number of parameters increases
- Markov Chain Monte Carlo (MCMC)
  - **Metropolis algorithm**
    - Invented by physicists
    - Assumes some symmetry in the sampling mechanism
  - Metropolis-Hastings
    - Generalizes to include asymmetry
  - Gibbs sampling
    - Uses conditional probabilities
- Hamilton Monte Carlo
    - Uses physics ideas
    - Will cover this more after reading break

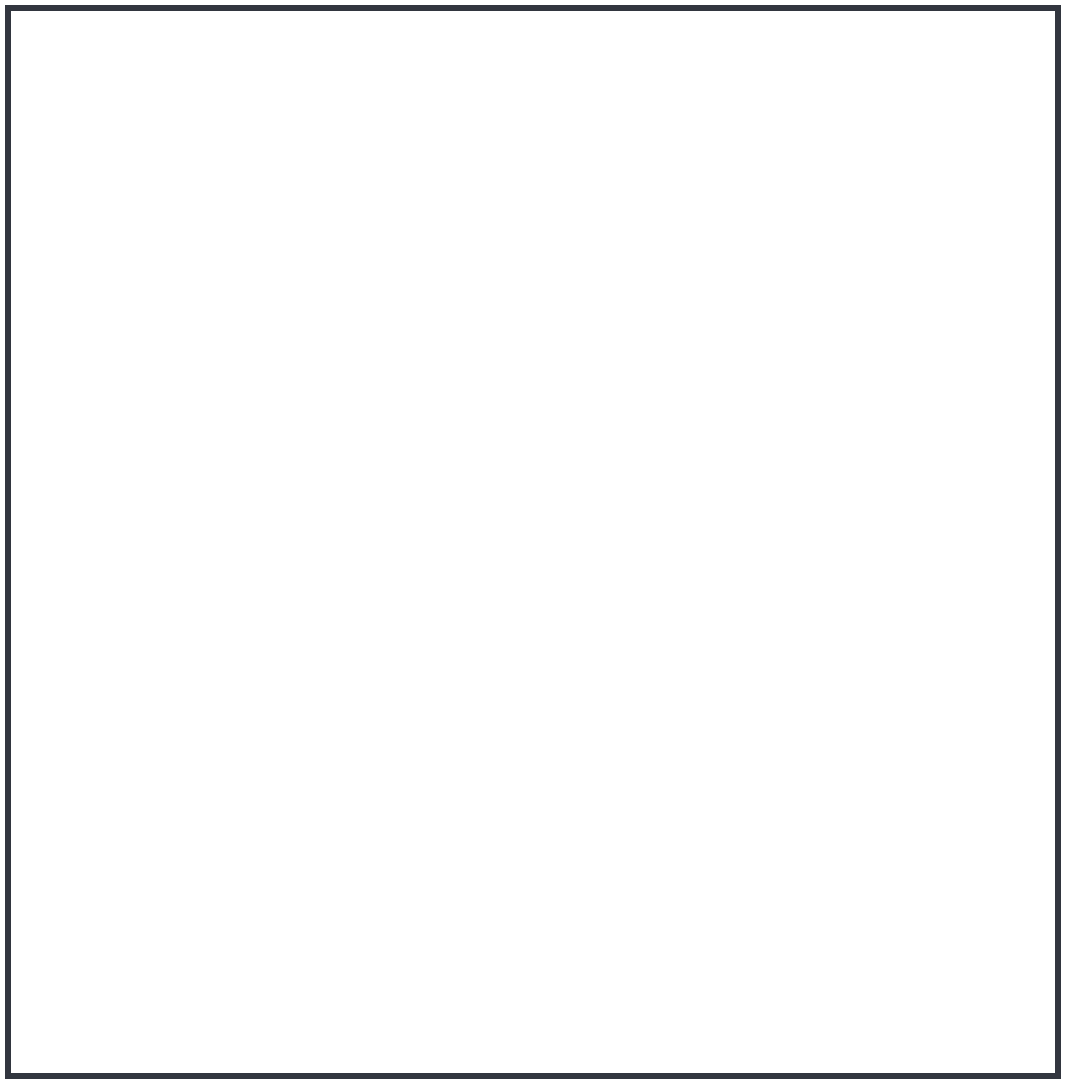
Bayesian Linear Regression

- Obtain samples that are drawn from the posterior distribution for β_0, β_1, etc given the data, model, and priors.



Helpful animation:
https://chi-feng.github.io/mcmc-demo/app.html?algorithm=RandomWalkMH&target=standard

What can you calculate with samples from a posterior?



- Probability of parameter being above or below some value
- Probability of parameter being within some range
- Credible intervals (e.g., 90% credible interval gives the inner 90% of probability)
- What parameter value has highest posterior probability
- Other scientific quantities of interest derived from parameter values
  - Credible intervals for these too!

Markov Chain Monte Carlo (MCMC)

- Markov chain simulation (aka MCMC) is a general method that aims to draw values from some target distribution
- Sampling is done sequentially
  - Ideally each sample depends only on the sample drawn immediately before it
  - that is, a *Markov chain* is created in the sampling process
  - For a chain with t samples, i.e., $\theta^1, \theta^2, \ldots,$ each sample θ^t depends only on sample θ^{t-1}
- Cannot just run the chains and assume everything worked
  - Assessing convergence is very important
  - Need good "mixing" of the chains

Metropolis algorithm

Invented by physicists!

Cited over 47,000 times

THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

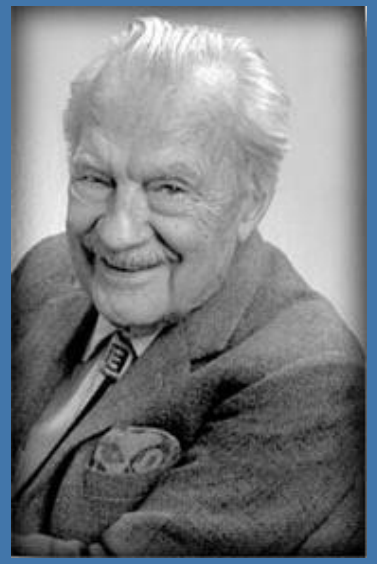
NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

AND

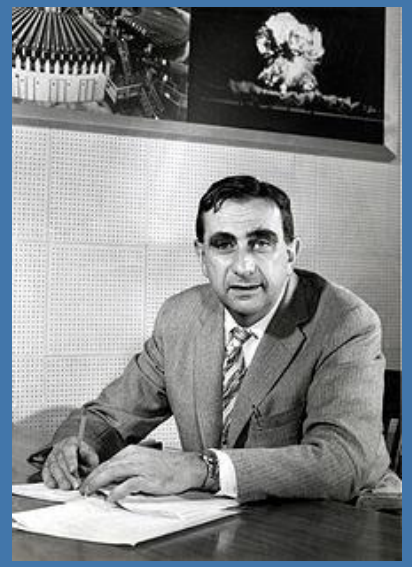
EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.







THE JOURNAL OF CHEMICAL PHYSICS VOLUME 21, NUMBER 6 JUNE, 1953

Equation of State Calculations by Fast Computing Machines

NICHOLAS METROPOLIS, ARIANNA W. ROSENBLUTH, MARSHALL N. ROSENBLUTH, AND AUGUSTA H. TELLER,
Los Alamos Scientific Laboratory, Los Alamos, New Mexico

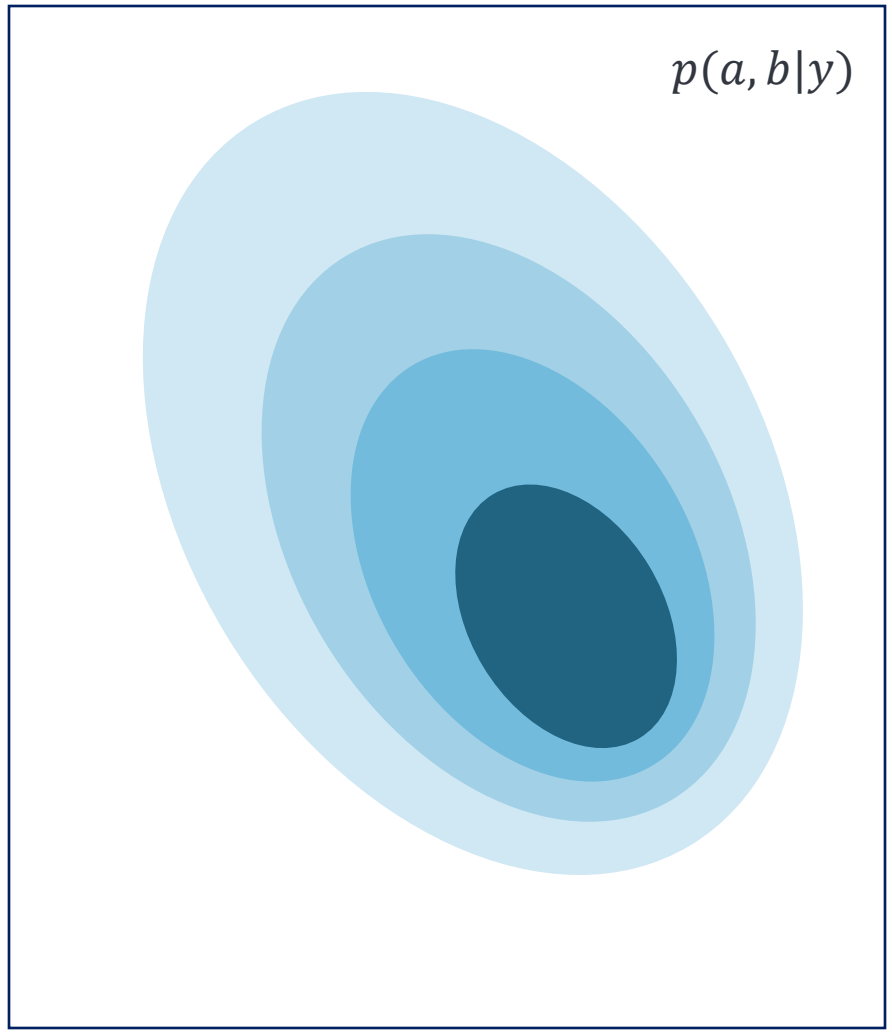
AND

EDWARD TELLER,* *Department of Physics, University of Chicago, Chicago, Illinois*

(Received March 6, 1953)

A general method, suitable for fast computing machines, for investigating such properties as equations of state for substances consisting of interacting individual molecules is described. The method consists of a modified Monte Carlo integration over configuration space. Results for the two-dimensional rigid-sphere system have been obtained on the Los Alamos MANIAC and are presented here. These results are compared to the free volume equation of state and to a four-term virial coefficient expansion.

Sketch of how the Metropolis (Rosenbluth) algorithm works for a two-parameter posterior distribution p(a,b|y)



1. Current state is somewhere in parameter space (a_{t-1}, b_{t-1})
2. Suggest a new place in parameter space (a^*, b^*) by making a "jump" according to the proposal distribution
   a. Compare $p(a^*, b^*|y)$ to $p(a_{t-1}, b_{t-1}|y)$
      - Accept a* and b^* if $p(a^*, b^*|y) > p(a_{t-1}, b_{t-1}|y)$
      - If $p(a^*, b^*|y) < p(a_{t-1}, b_{t-1}|y)$ then accept only with some probability
3. If (a^*, b^*) are accepted, then save these in the Markov chain in spot t. If not accepted, then save again (a_{t-1}, b_{t-1}) in spot t
4. Repeat steps 2-3 many times, saving the (a, b) values as you go, to get your Markov chain

Markov Chain Monte Carlo (MCMC)

- Notation
  - θ^* → suggested parameter value(s) (e.g., (a^*, b^*) in our example)
  - θ_{old} → value of parameters saved in previous step of Markov chain

- **Accept/reject step**
  - If $p(\theta^*|y) > p(\theta_{old}|y)$, then accept θ^* as next value in Markov chain
    - i.e., if $\mathrm{r} = \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$ **is greater than 1**, then go to that place in parameter space
  - If $p(\theta^*|y) < p(\theta_{old}|y)$, i.e. $\mathrm{r} = \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$ **is less than 1**, then …
    - draw a random number x from $\mathbf{Unif(0,1)}$
    - If $x < \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$, then accept θ^*. Otherwise reject θ^* and stay at θ_{old}

Metropolis (Rosenbluth) algorithm → the accept/reject step

- Notation
  - θ^* → suggested parameter value(s) (e.g., (a^*, b^*) in our example)
  - θ_{old} → value of parameters saved in previous step of Markov chain

- Accept/reject step
  - If $p(\theta^*|y) > p(\theta_{old}|y)$, then accept θ^* as next value in Markov chain
    - i.e., if $\mathrm{r} = \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$ **is greater than 1**, then go to that place in parameter space
  - If $p(\theta^*|y) < p(\theta_{old}|y)$, i.e. $\mathrm{r} = \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$ **is less than 1**, then …
    - draw a random number x from $\mathrm{Unif}(0,1)$
    - If $\boldsymbol{x < \frac{p(\theta^*|y)}{p(\theta_{old}|y)}}$ then accept $\boldsymbol{\theta^*}$, otherwise reject θ^* and stay at θ_{old}

Metropolis (Rosenbluth) algorithm → the accept/reject step

- If $x < \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$ then accept θ^*, otherwise reject θ^* and stay at θ_{old}

- Notice that you don't need the normalization constant for the ratio above:

$$r = \frac{p(\theta^*|y)}{p(\theta_{old}|y)} = \frac{\frac{p(y|\theta^*)p(\theta^*)}{p(y)}}{\frac{p(y|\theta_{old})p(\theta_{old})}{p(y)}}$$

Metropolis-Hastings Algorithm

Generalizes Metropolis algorithm to an asymmetric jumping (proposal) distribution

- Can be more efficient if you know the target distribution is skewed in some way

Instead of the following used in the accept/reject step

$$r = \frac{p(\theta^*|y)}{p(\theta_{old}|y)}$$

the ratio r is replaced with

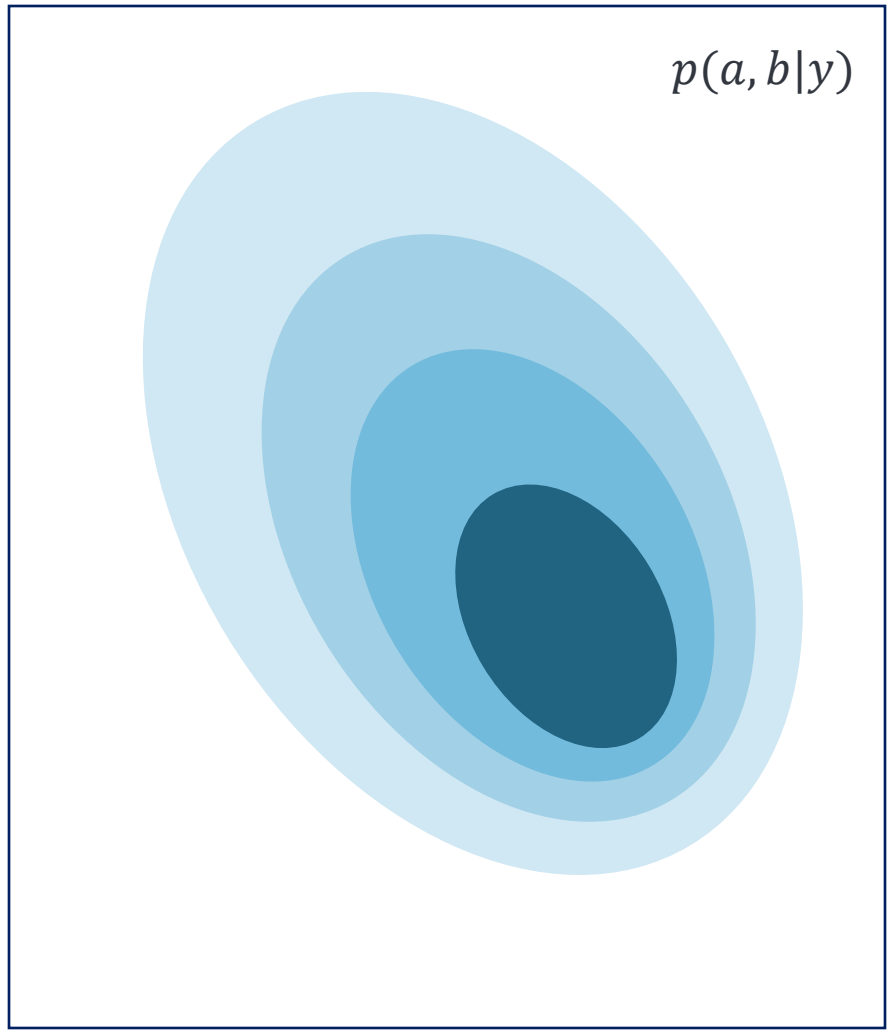
$$\boldsymbol{r = \frac{\frac{p(\theta^*|y)}{J_t(\theta^*|\theta_{old})}}{\frac{p(\theta_{old}|y)}{J_t(\theta_{old}|\theta^*)}}}$$

→ This accounts for the asymmetric jumping distribution $\boldsymbol{J_t}$

A helpful visualization tool for different sampling algorithms

- https://chi-feng.github.io/mcmc-demo/

Sketch of how the Metropolis (Rosenbluth) algorithm works for a two-parameter posterior distribution p(a,b|y)



1. Start somewhere in parameter space
   a. this is the first value in the Markov chain, a_1, b_1
2. "Jump" to a new place in parameter space (a^*, b^*)
   a. Call (a^*, b^*) *suggested* values; don't save them yet!
   b. Compare $p(a^*, b^*|y)$ to $p(a_1, b_1|y)$
      - Accept a^* and b^* if $p(a^*, b^*|y) > p(a_1, b_1|y)$
      - If $p(a^*, b^*|y) < p(a_1, b_1|y)$ then accept only with some probability (we'll get to this)
3. If $(a^* b^*)$ are accepted, then save these in the Markov chain in spot #2. If not accepted, then save again (a_1, b_1) in spot #2
4. Repeat steps 2-3 many times, saving the (a, b) values as you go, to get your Markov chain

a	b

Proposal distribution

- What makes a good proposal distribution?
  - Easy to sample from $J(\theta^*|\theta)$
  - Easy to compute the ratio r
  - Jumps are not too large (this will cause too many reject steps, and will be inefficient)
  - Jumps are not too small (this will cause too many accept steps, and won't explore parameter space)
- "Tuning" the proposal
  - E.g., if using a multivariate normal distribution centered on 0, then adjust the Σ matrix until a good acceptance rate is achieved
  - Some methods do the above automatically (e.g., adaptive-Metropolis algorithms)

Other sampling methods we haven't talked about

- Gibbs Sampling
- Trans-dimensional Markov chain simulation
  - When you have a number of candidate models with different numbers of parameters
- Nested sampling (invented in 2004 by physicist John Skilling)
- Slice sampling
- Many others!

- Hamiltonian Monte Carlo

Hamiltonian Monte Carlo

- Actually more computationally costly than Metropolis or Gibbs sampling, **but *it's more efficient***
- HMC doesn't need as many samples as Metropolis (Rosenbluth) or Gibbs sampling
- When models are highly complex, HMC can easily outshine other methods
  - "the Earth would be swallowed by the Sun before your chain produces a reliable approximation to the posterior" – McElreath

Exercise: Linear regression in a Bayesian context

Linear Regression in Bayesian context

- The model for y is
- The likelihood under this model is
- We must set priors on the parameters