GENERAL ASSEMBLY

MARKOV CHAIN MONTE CARLO (MCMC) METHODS

Matt Brems

Data Science Immersive, GA DC

LEARNING OBJECTIVES

- Understand and be able to describe how MCMC works.
- Implement MCMC in Python.

OPENING: SALARIES OF DATA SCIENTISTS

- What is relevant when attempting to predict the salary of data scientists?
- In doing a project about data science salaries, what might our goal be?

OPENING: BAYES' THEOREM

$$f(\theta|y) = \frac{f(y|\theta)f(\theta)}{f(y)} = \frac{L(\theta|y)f(\theta)}{f(y)} \propto L(\theta|y)f(\theta)$$

- Why would we want the posterior distribution?

OPENING: BAYES' THEOREM

$$f(\theta|y) = \frac{f(y|\theta)f(\theta)}{f(y)} = \frac{L(\theta|y)f(\theta)}{f(y)} \propto L(\theta|y)f(\theta)$$

- Why would we want the posterior distribution?
- What is conjugacy?

OPENING: BAYES' THEOREM

$$f(\theta|y) = \frac{f(y|\theta)f(\theta)}{f(y)} = \frac{L(\theta|y)f(\theta)}{f(y)} \propto L(\theta|y)f(\theta)$$

- Why would we want the posterior distribution?
- What is conjugacy?
- What happens when we don't have conjugacy?

OPENING: BAYES' THEOREM

$$f(\theta|y) = \frac{f(y|\theta)f(\theta)}{f(y)} = \frac{L(\theta|y)f(\theta)}{f(y)} \propto L(\theta|y)f(\theta)$$

- Why would we want the posterior distribution?
- What is conjugacy?
- What happens when we don't have conjugacy?
- If we can't find a "closed-form" solution, can we at least approximate one?

MAXIMUM LIKELIHOOD ESTIMATION

MCMC METHODS

MCMC COMPONENTS

- There are three main components to MCMC methods:
  - Monte Carlo Simulations
  - Markov Chains
  - Metropolis-Hastings Algorithm

MCMC COMPONENTS

- There are three main components to MCMC methods:
    - **Monte Carlo Simulations**
    - Markov Chains
    - Metropolis-Hastings Algorithm

MONTE CARLO

- Monte Carlo methods allow us to approximate a complicated system with a statistical sample.

- This is a way of generating random numbers.

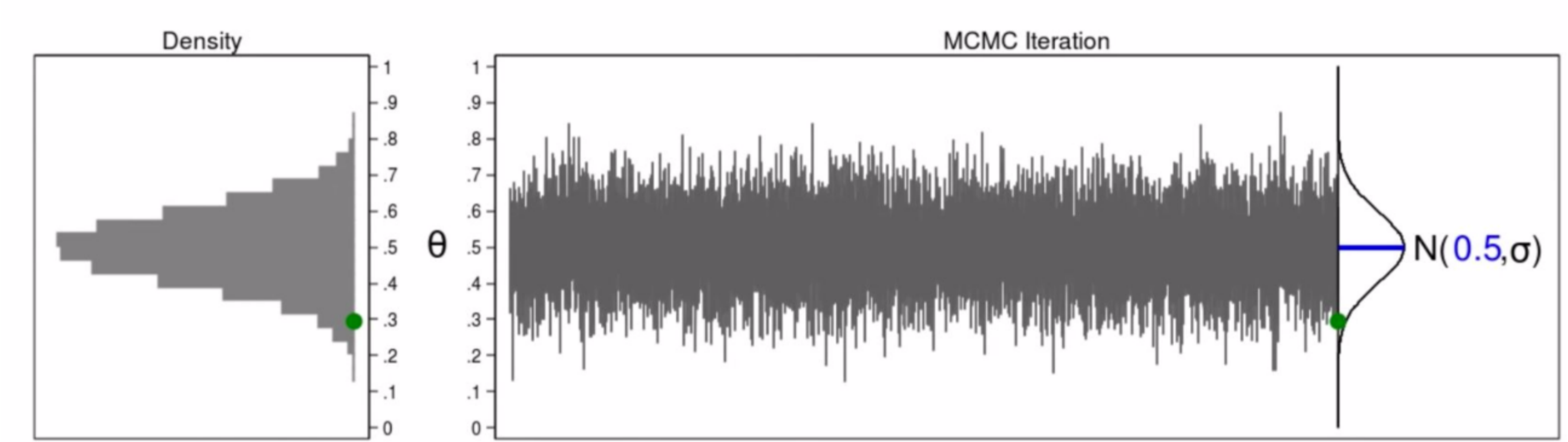
$$\theta_t \sim N(0.5, \sigma)$$

- We can use this to solve a number of problems – like approximating the value of π!

MONTE CARLO

- This is a way of generating random numbers.

$$\theta_t \sim N(0.5, \sigma)$$



MCMC COMPONENTS

- There are three main components to MCMC methods:
    - Monte Carlo Simulations
    - **Markov Chains**
    - Metropolis-Hastings Algorithm

MARKOV CHAINS

- Remember we've discussed Markov chains in our linear algebra lecture.
  - What is the "Markov property?"
  - What was something that we noticed when we looked at the long-run behavior of Markov chains?

MARKOV CHAINS

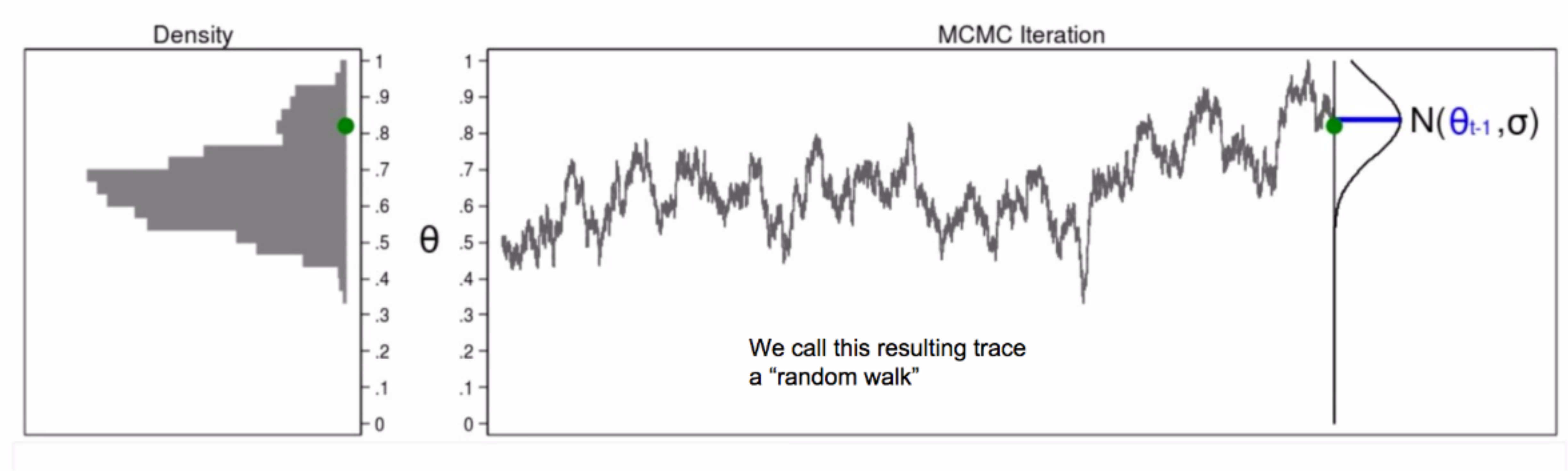
- Remember we've discussed Markov chains in our linear algebra lecture.
    - What is the "Markov property?"
    - What was something that we noticed when we looked at the long-run behavior of Markov chains?

$$\theta_t \sim N(\theta_{t-1}, \sigma)$$

MARKOV CHAINS

- Remember we've discussed Markov chains in our linear algebra lecture.

$$\theta_t \sim N(\theta_{t-1}, \sigma)$$



MCMC COMPONENTS

- There are three main components to MCMC methods:
  - Monte Carlo Simulations
  - Markov Chains
  - **Metropolis-Hastings Algorithm**

METROPOLIS-HASTINGS ALGORITHM

- Thus far, we’ve been generating all of these random numbers but have no idea if the numbers we’re generating are headed in the right direction.

METROPOLIS-HASTINGS ALGORITHM

- Thus far, we've been generating all of these random numbers but have no idea if the numbers we're generating are headed in the right direction.

- The Metropolis-Hasting algorithm allows us to, at each step, identify whether we are getting "hotter" or "colder."

METROPOLIS-HASTINGS ALGORITHM

- Thus far, we've been generating all of these random numbers but have no idea if the numbers we're generating are headed in the right direction.
- The Metropolis-Hasting algorithm allows us to, at each step, identify whether we are getting "hotter" or "colder."
- Recall that we're simulating $\theta_t \sim N(\theta_{t-1}, \sigma)$.

METROPOLIS-HASTINGS ALGORITHM

- Thus far, we've been generating all of these random numbers but have no idea if the numbers we're generating are headed in the right direction.
- The Metropolis-Hasting algorithm allows us to, at each step, identify whether we are getting "hotter" or "colder."
- Recall that we're simulating $\theta_t \sim N(\theta_{t-1}, \sigma)$.
- Let's not move so quickly and call this θ_t. Let's call it θ_{new} instead.

$$\theta_{new} \sim N(\theta_{t-1}, \sigma)$$

METROPOLIS-HASTINGS ALGORITHM

- Naturally, our question is… well, where did θ_t go and how do we find it?

$$r(\theta_{new}, \theta_{t-1}) = \frac{L(y|\theta_{new})P(\theta_{new})/P(y)}{L(y|\theta_{t-1})P(\theta_{t-1})/P(y)} = \frac{L(y|\theta_{new})P(\theta_{new})}{L(y|\theta_{t-1})P(\theta_{t-1})} = \frac{P(\theta_{new}|y)}{P(\theta_{t-1}|y)}$$

METROPOLIS-HASTINGS ALGORITHM

- Naturally, our question is… well, where did θ_t go and how do we find it?

$$r(\theta_{new}, \theta_{t-1}) = \frac{P(\theta_{new}|y)}{P(\theta_{t-1}|y)}$$

METROPOLIS-HASTINGS ALGORITHM

- Naturally, our question is… well, where did θ_t go and how do we find it?

$$r(\theta_{new}, \theta_{t-1}) = \frac{P(\theta_{new}|y)}{P(\theta_{t-1}|y)}$$

$$\alpha_t = \min\{r, 1\}$$

METROPOLIS-HASTINGS ALGORITHM

- Naturally, our question is… well, where did θ_t go and how do we find it?

$$r(\theta_{new}, \theta_{t-1}) = \frac{P(\theta_{new}|y)}{P(\theta_{t-1}|y)}$$

$$\alpha_t = \min\{r, 1\}$$

$$u_t \sim Uniform(0,1)$$

METROPOLIS-HASTINGS ALGORITHM

- Naturally, our question is… well, where did θ_t go and how do we find it?

$$r(\theta_{new}, \theta_{t-1}) = \frac{P(\theta_{new}|y)}{P(\theta_{t-1}|y)}$$

$$\alpha_t = \min\{r, 1\}$$

$$u_t \sim Uniform(0,1)$$

$$\theta_t = \begin{cases} \theta_{t-1} & \text{if } u_t > \alpha_t \\ \theta_{new} & \text{if } u_t \le \alpha_t \end{cases}$$

METROPOLIS-HASTINGS ALGORITHM

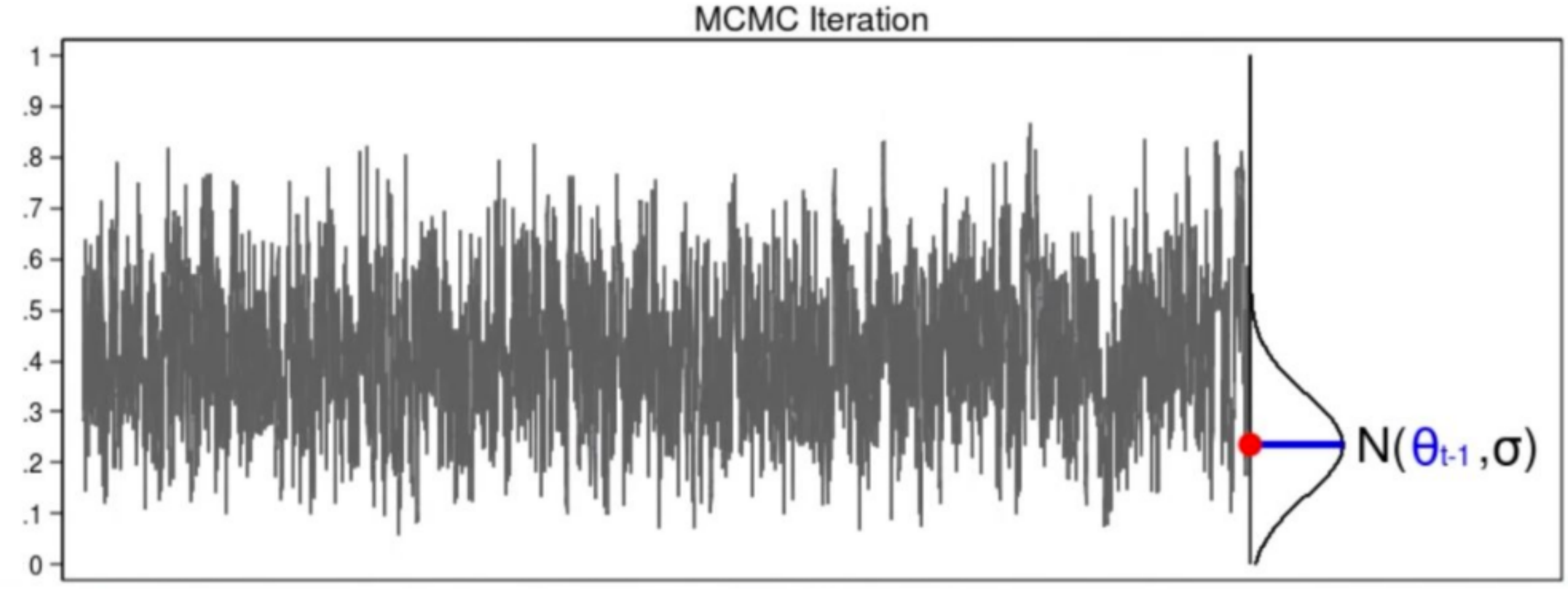
- Naturally, our question is… well, where did θ_t go and how do we find it?
  - Step 1: Generate $\theta_{new} \sim N(\theta_{t-1}, \sigma)$.
  - Step 2: Calculate $r(\theta_{new}, \theta_{t-1})$, the ratio of posteriors of θ_{new} and θ_{t-1}.
  - Step 3: Calculate the acceptance probability $\alpha_t = \min\{r, 1\}$.
  - Step 4: Generate u_t from a $Uniform(0,1)$ distribution.
  - Step 5: If the generated number u_t is greater than the acceptance probability α_t, then $\theta_t = \theta_{t-1}$. Otherwise, $\theta_t = \theta_{new}$.

METROPOLIS-HASTINGS ALGORITHM

- We do this many, many times.

METROPOLIS-HASTINGS ALGORITHM

- We do this many, many times.



Source: STATA

METROPOLIS-HASTINGS ALGORITHM

- We do this many, many times.
- Once we do this long enough and visually inspect plots to convince ourselves that we have “converged” to the posterior distribution of interest, then we sample some large n from this posterior.

METROPOLIS-HASTINGS ALGORITHM

- We do this many, many times.

- Once we do this long enough and visually inspect plots to convince ourselves that we have “converged” to the posterior distribution of interest, then we sample some large n from this posterior.

- Once we have our sample of size n, we can conduct whatever inference we want.
  - Find the mean.
  - Find the median.
  - Find the 95% ‘highest posterior density.’
  - Find the variance.
  - Find the…