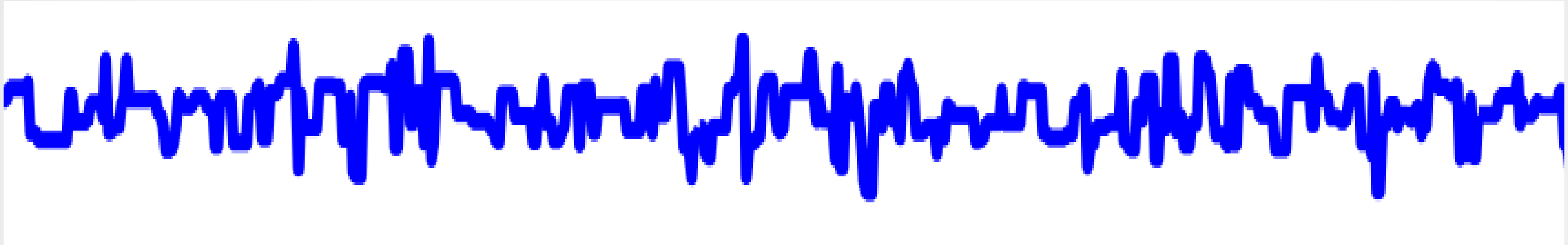
STATS331

Credit:
lesswrong.com

Introduction to Bayesian Statistics
Semester 2, 2016

Markov Chain Monte Carlo

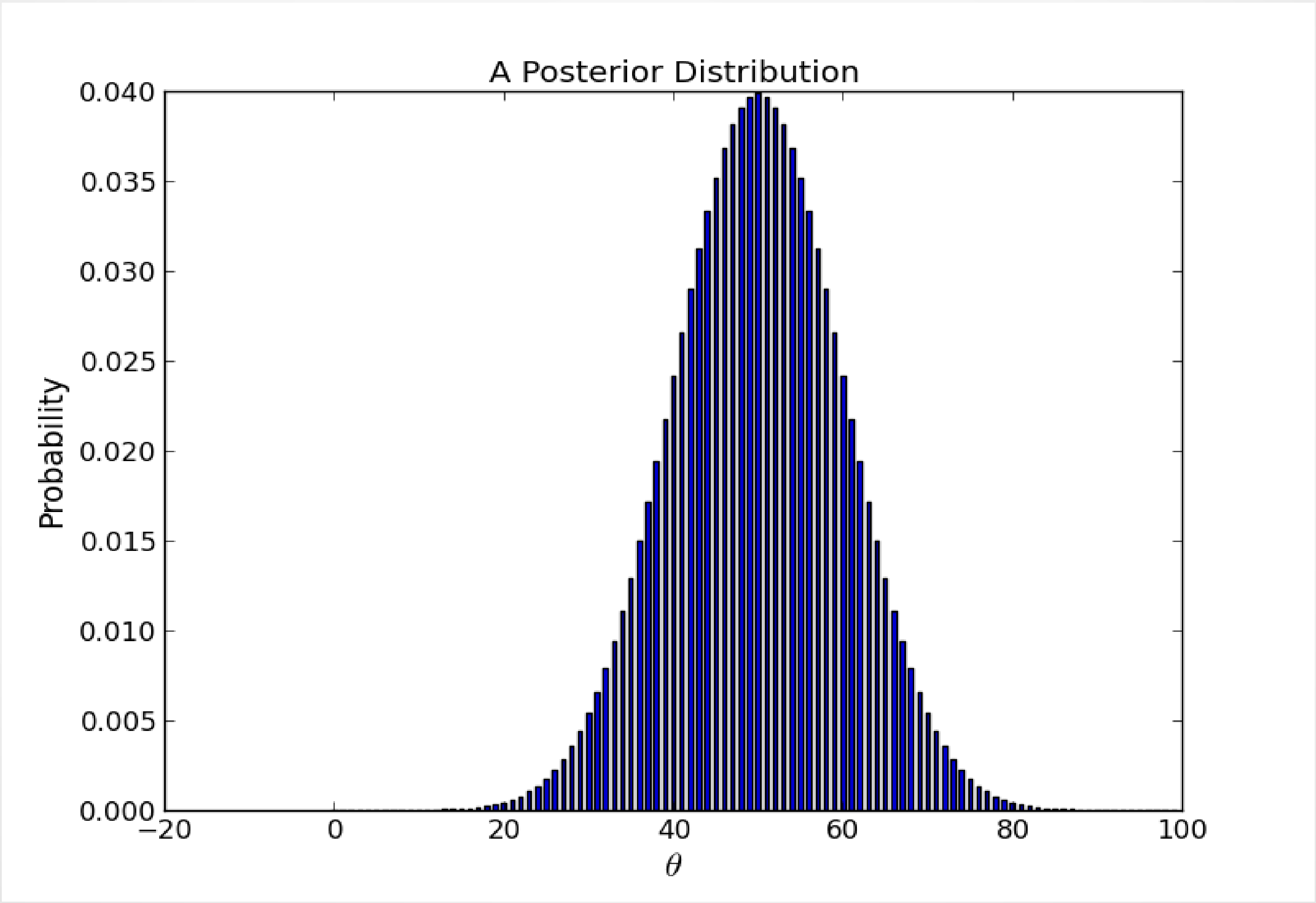


MCMC

Philosophy

- MCMC is not Bayesian
- It is a numerical algorithm that happens to be useful for us.

P(D)Fs and Samples

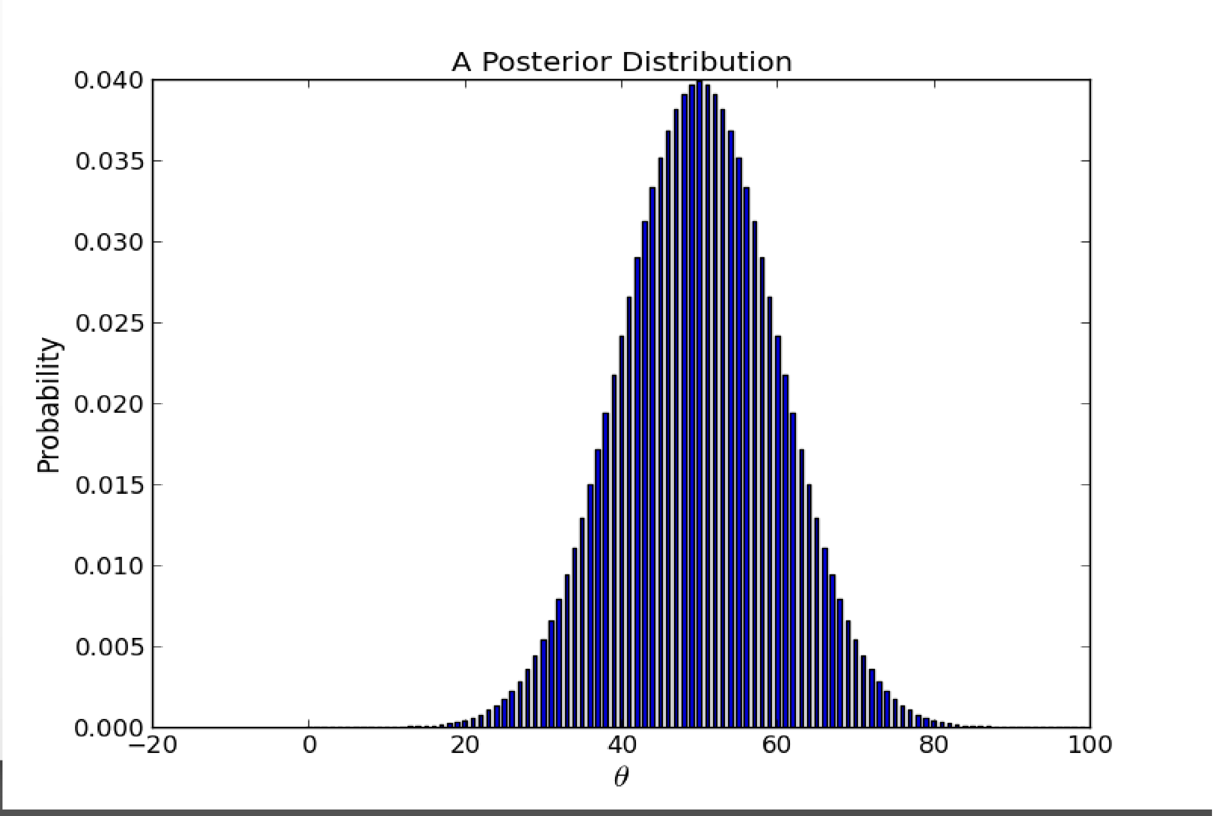


To Get Posterior Summaries

- You would need to do things like

```
> sum(theta*post)
```

to get the expectation value (posterior mean) point estimate



More than one parameter

- Imagine doing a Bayes' Box for more than one parameter

Possible Answers

(theta1, theta2)

(0, 0)

(0, 0.01)

…

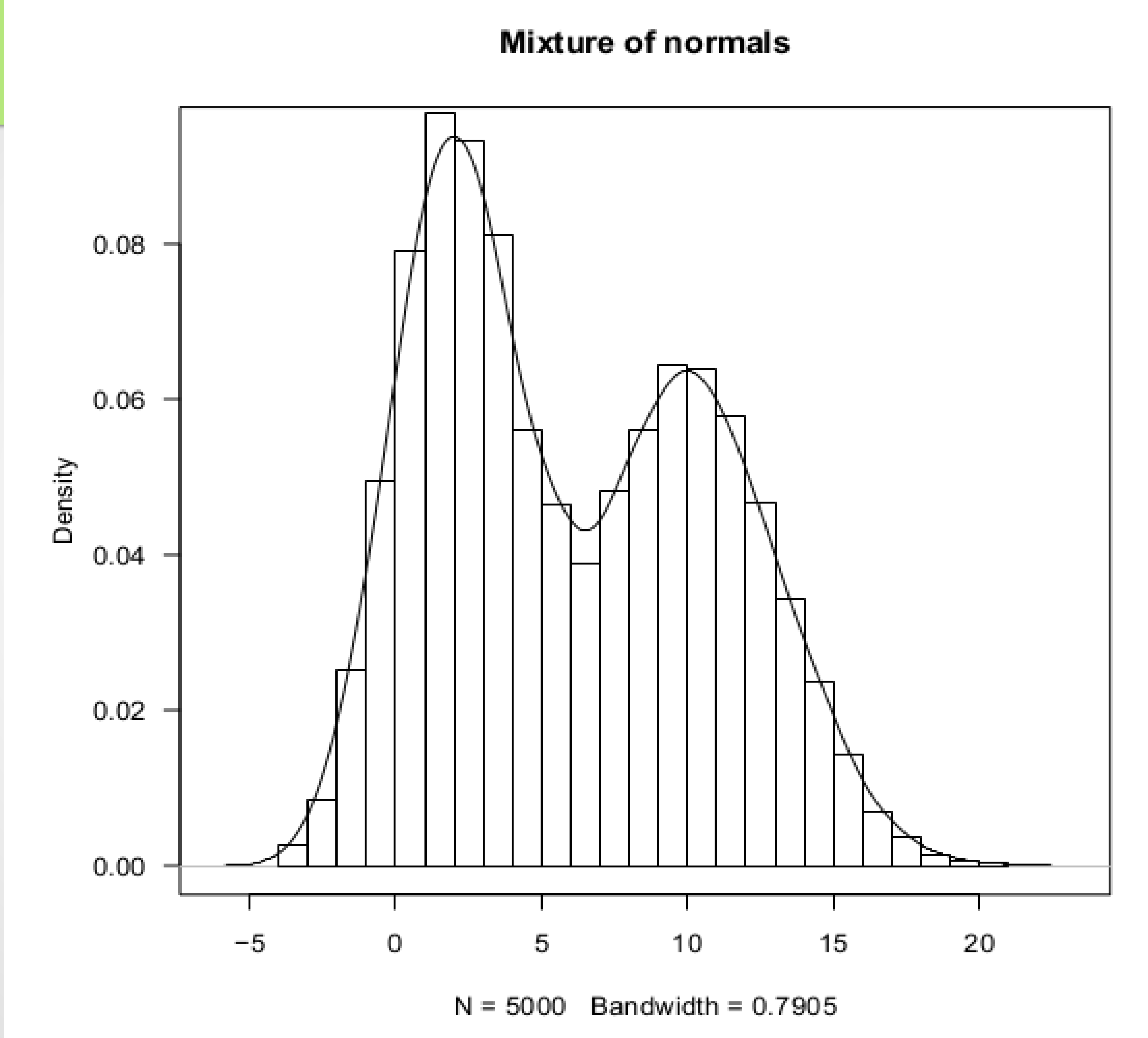
(0.01, 0)

(0.01, 0.01)

...

P(D)Fs and Samples

- If we can generate random numbers from the posterior...then the Histogram will look just like the P(D)F!
- If we have enough random numbers, we can use the **sample mean** which will be almost the same as the actual posterior mean
- Same with sd or any other summary


Mixture of normals
Density
0.00
0.02
0.04
0.06
0.08
−5
0
5
10
15
20
N = 5000   Bandwidth = 0.7905

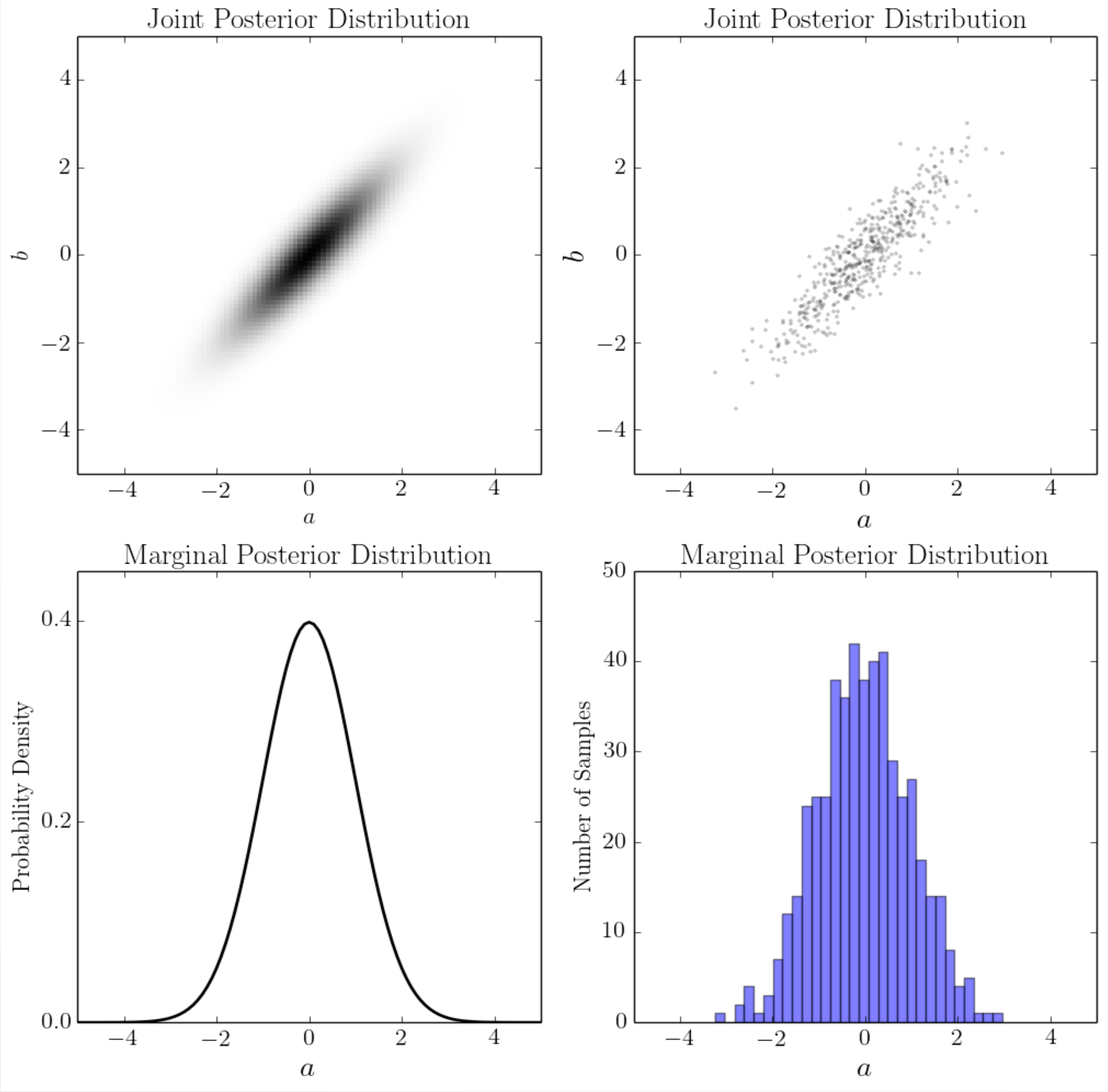
```
# If you have a vector of values
# and posterior probs
> sum(theta*post)

> # If you have posterior samples
> mean(theta_samples)
```

More Samples

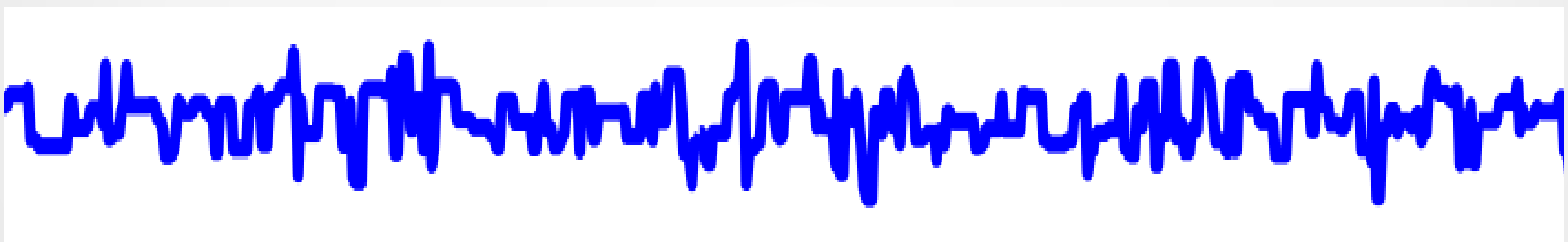
- More samples means the sample more accurately reflects the posterior
- But the posterior still has the uncertainty built in
- We will assume that we always have a ‘large enough’ sample from the posterior. Won’t worry about extra ‘Monte Carlo’ uncertainty

Marginalisation



Markov Chain

- Means a sequence of random numbers, where each one depends on the previous one but not on the ones before that



$$
\begin{aligned}
p(X_1, X_2, ..., X_N) &= p(X_1)p(X_2|X_1)p(X_3|X_2, X_1)...p(X_N|X_{N-1}, ..., X_1) \\
&= p(X_1)p(X_2|X_1)p(X_3|X_2)...p(X_N|X_{N-1})
\end{aligned}
$$

Example of a Markov Chain

- Random Walk
- Start at x=0
- Flip a coin.
- Heads? Add 1. Tails? Subtract 1.

0, 1, 2, 3, 2, 1, 2, 3, 4, 3, 4, 5, 6, 5, 4, 3, …

- Like a drunk person trying to walk home :-)

Monte Carlo

- Means “a method that uses random numbers”
- In an alternative universe, MCMC could have been MCLV

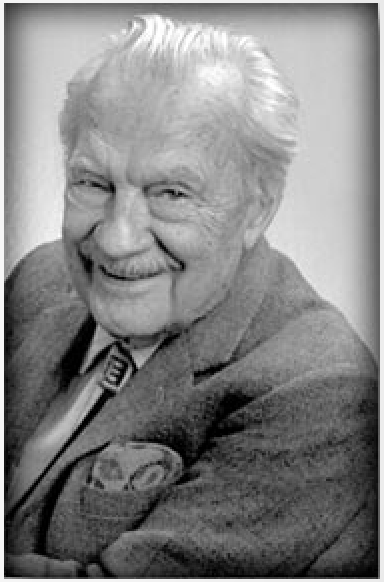
- Markov Chain **Las Vegas**


DEAL OR NO DEAL
CASH
DENOM
1¢
Game Over
DEAL OR NO DEAL
Atronic

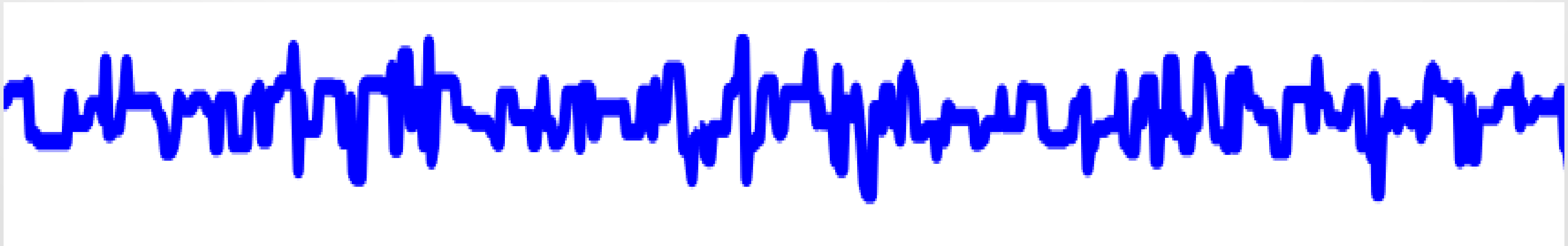
Metropolis

The original MCMC method

This is not a picture of a metropolis



Public domain image



Metropolis: The First MCMC Method

- Want to sample the posterior distribution
- In MCMC land, the distribution you want to sample is called the “target distribution”
- Start with something you *can* do, called the “proposal distribution”

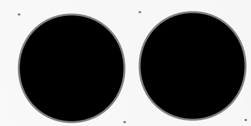
Massaging the Proposal

- Left to its own devices, the proposal would sample *some* probability distribution
- We use an acceptance/rejection rule to change that distribution into the one we want

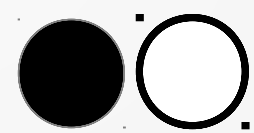
Example: Two Possibilities

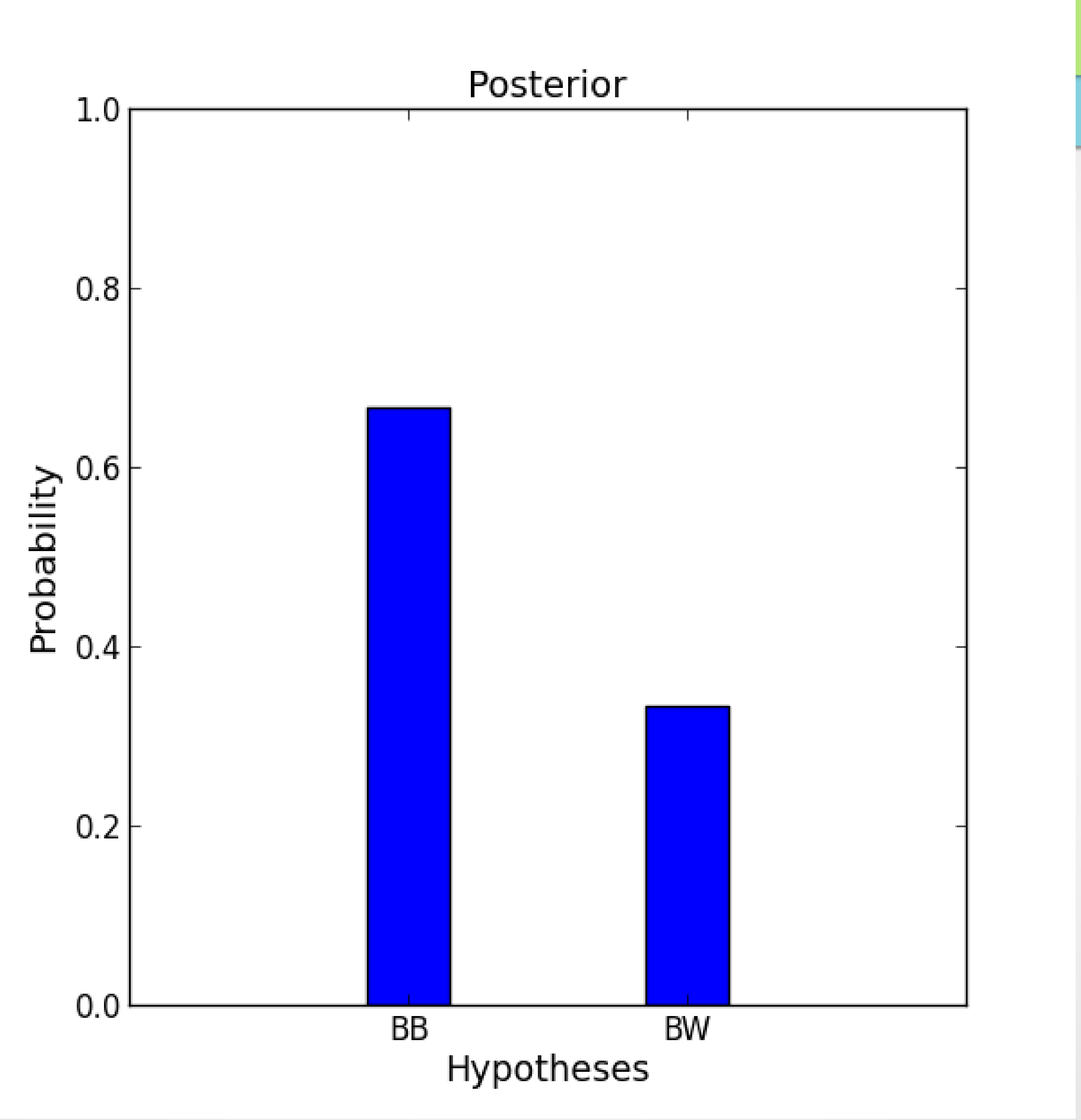
- Here we go again...!!!

- BB. Post. Prob = 2/3



- BW. Post. Prob = 1/3




Posterior
Probability
1.0
0.8
0.6
0.4
0.2
0.0
BB
BW
Hypotheses

Making a Sampler

- We need to make a Markov Chain that will sample this posterior

- It should spend 2/3 of the time in state 1 (BB)

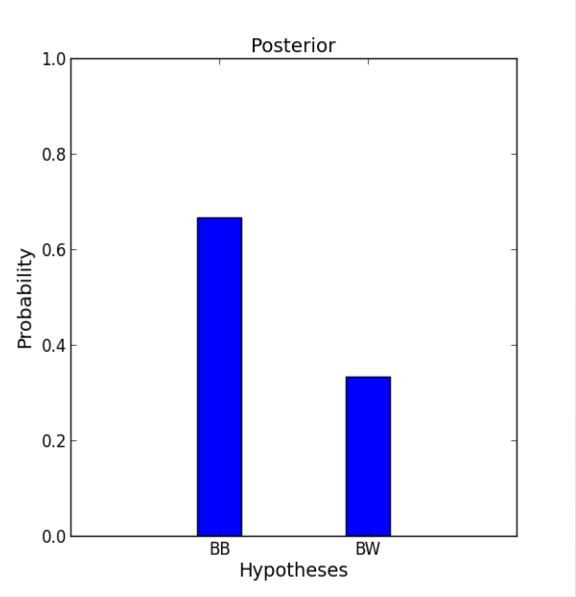
and 1/3 of the time in state 2 (BW)

- It would be nice to use `prior x likelihood` values (aka the unnormalised posterior)

How About Flipping Coins?

- Uniform proposal: ½ and ½

Iteration (Time)	State
1	1
2	1
3	2
4	1
5	2
6	2
7	1
8	2
9	2



We Need Something Else

- We need to do something to the process so that it spends more time in State 1 than State 2.
- How?
- When the state tries to change from State 1 to State 2, **we won't let it**

Full Rejection

- Uniform proposal: ½ and ½

Iteration (Time)	Proposal	Accept?	State
1	1	Yes	1
2	2	No	1
3	1	Yes	1
4	1	Yes	1
5	2	No	1
6	2	No	1
7	2	No	1
8	1	Yes	1
9	2	No	1

Ideas That Don't Work

- Accepting all moves from 1 → 2. Leads to steady state distribution {½, ½}
- Rejecting all moves from 1 → 2. Leads to steady state distribution {1, 0}

- Need something in between!

Something In Between

- What if we reject **some** moves from 1 $\rightarrow$ 2?
- We can accept with probability α and reject with probability $1 - \alpha$.
- Let's try a value of $\alpha = h_2/h_1 = 1/2$

Iteration	Proposal State (COIN)	Alpha	Acceptance Set	Die Outcome	State
1	-	-	-	-	1
2	1	1	{1, 2, 3, 4, 5, 6}	4	1
3	2	0.5	{4, 5, 6}	3	1
4	2	0.5	{4, 5, 6}	6	2
5					
6					
7					
8					
9					
10					
11					
12					
13					

The Proposal Is Not The Prior!

- In our example, the proposal happens to be the same as the prior
- This isn't usually true

- Prior = what we knew about θ before the data
- Proposal = just something to help the computer generate samples!

Aside

- How to make a computer do something

  “with probability α”?

```
alpha = 0.71
if(runif(1) < alpha)
{
    # Do stuff
}
```