Monte Carlo Markov Chains: A Brief Introduction and Implementation

Jennifer Helsby
Astro 321

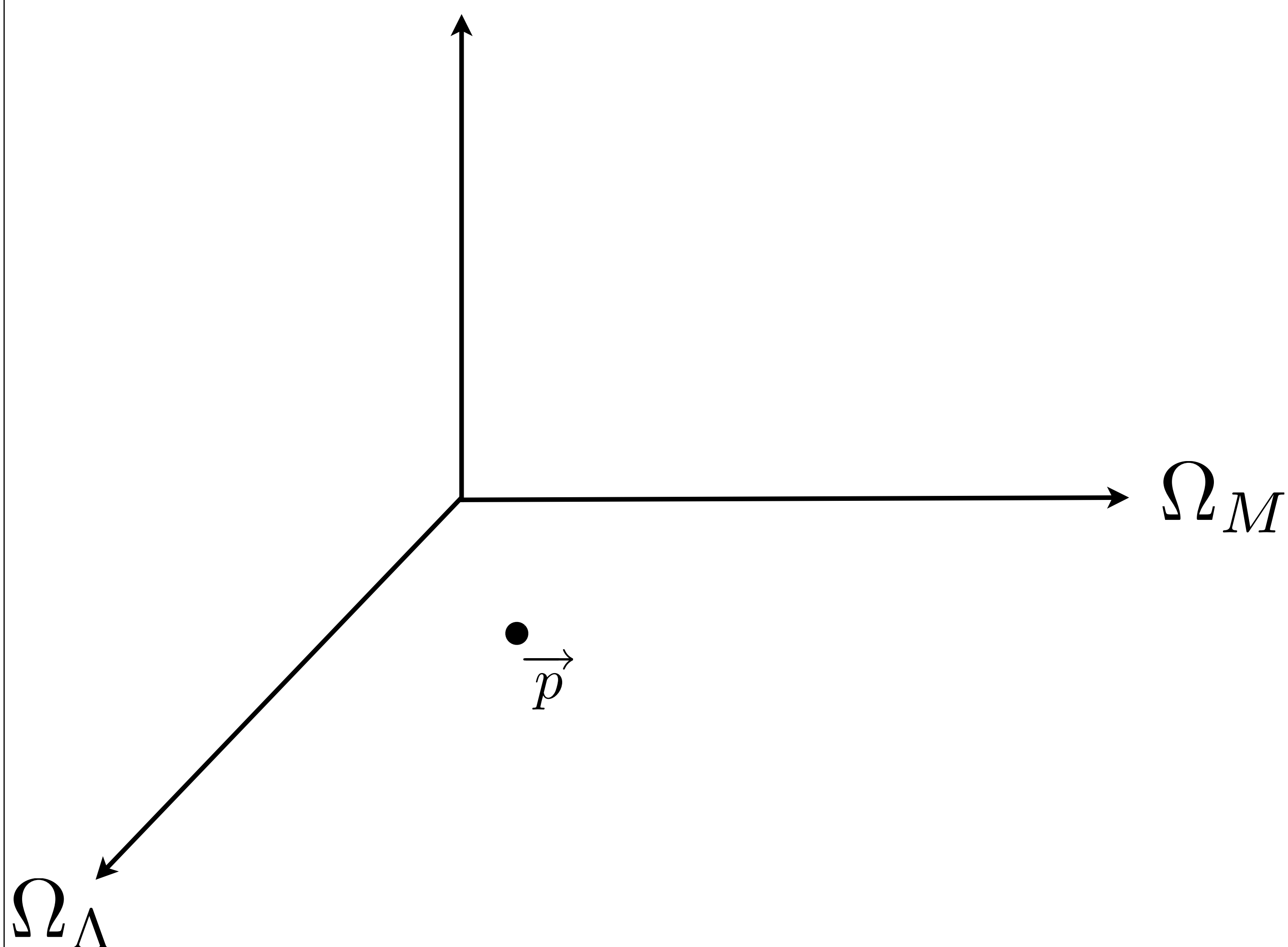
What are MCMC: Markov Chain Monte Carlo Methods?

- Set of algorithms that generate posterior distributions by sampling likelihood function in a representative way in parameter space

- Why does anybody do this?

    - Scales linearly with the number of parameters considered; thus, much more efficient that other methods

Anatomy of a Markov Chain: Diagram



1. Begin at a location in parameter space,

$$\overrightarrow{p} = (\Omega_{M,i}, \Omega_{\Lambda,i}, w_i)$$

2. Compute likelihood at this point,

$$\mathcal{L}_{\overrightarrow{p}}$$

3. Jump function proposes new location,

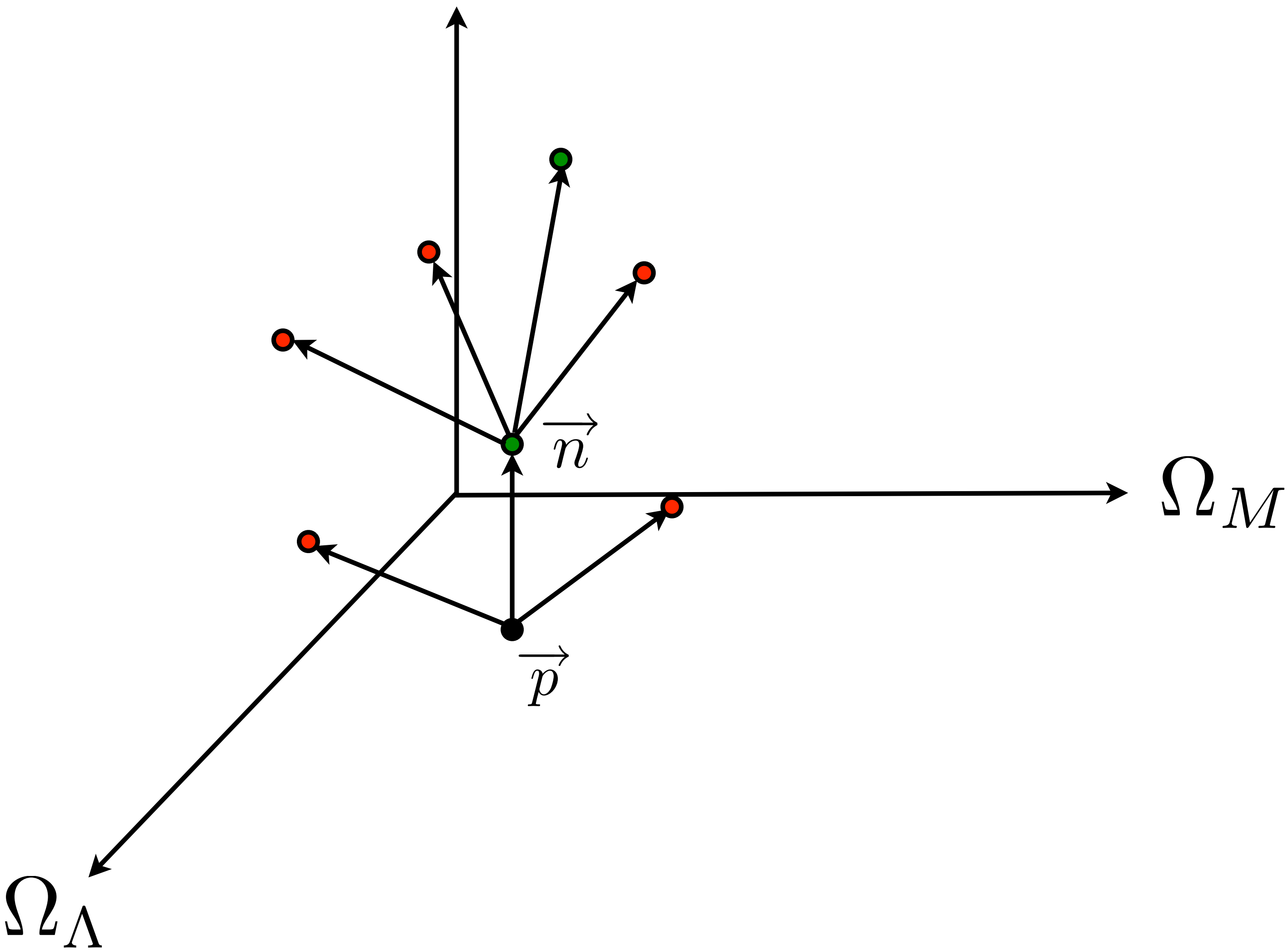
$$\overrightarrow{n} = (\Omega_{M,j}, \Omega_{\Lambda,j}, w_j)$$

4. Compute likelihood at this point,

$$\mathcal{L}_{\overrightarrow{n}}$$

5. An algorithm determines whether to move there (if jump rejected, repeat steps 3-4 until accepted)

Anatomy of a Markov Chain: Diagram



1. Begin at a location in parameter space,

$$\overrightarrow{p} = (\Omega_{M,i}, \Omega_{\Lambda,i}, w_i)$$

2. Compute likelihood at this point,

$$\mathcal{L}_{\overrightarrow{p}}$$

3. Jump function proposes new location,

$$\overrightarrow{n} = (\Omega_{M,j}, \Omega_{\Lambda,j}, w_j)$$

4. Compute likelihood at this point,

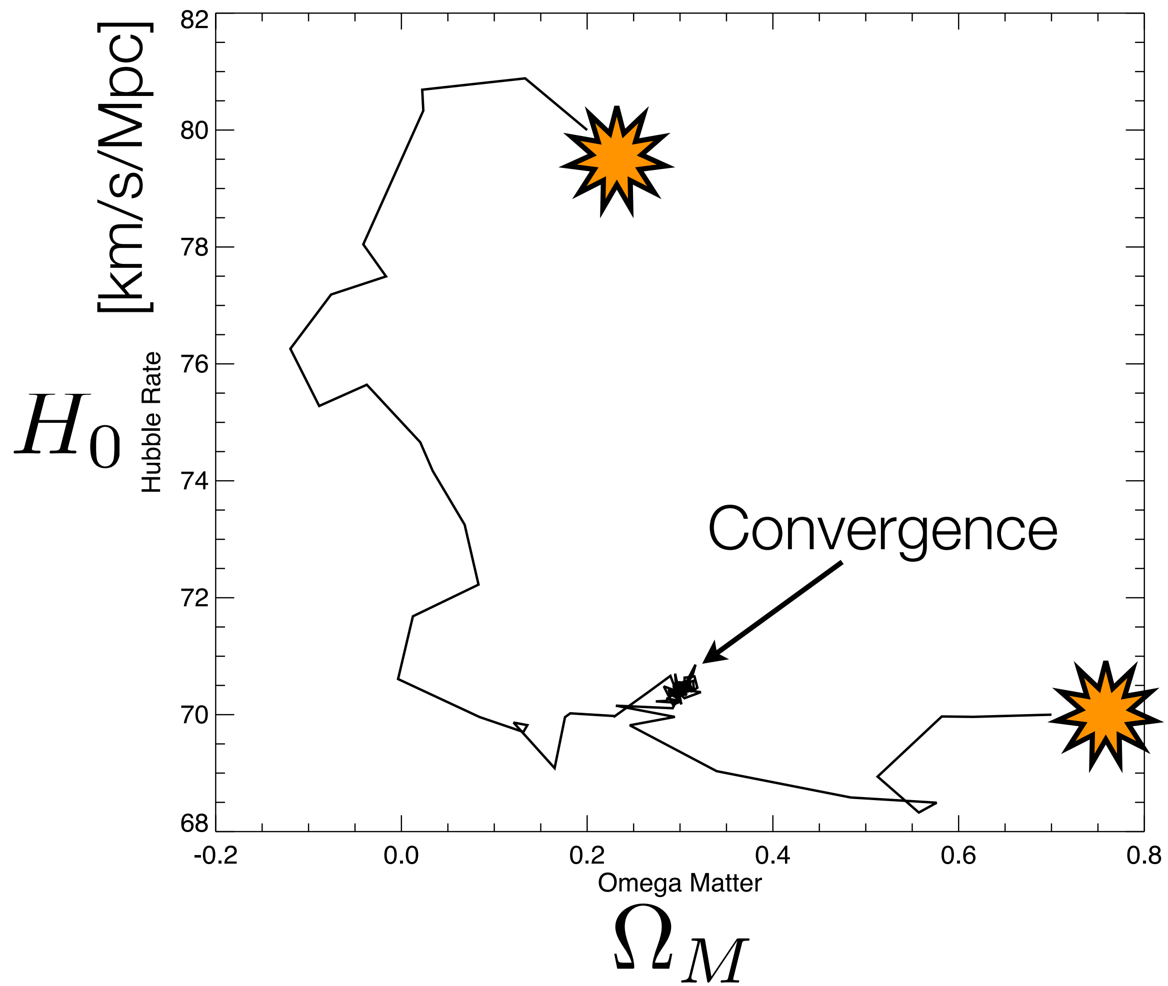
$$\mathcal{L}_{\overrightarrow{n}}$$

5. An algorithm determines whether to move there (if jump rejected, repeat steps 3-4 until accepted)

Burning in

- Beginning the Markov chain at any point in the parameter space should result in convergence
- First ~5-10% steps thrown away: "burn in" process
- Reduces dependence of reconstructed distribution on the initial position

Parameter space for 2 chains:



The Decider: Metropolis-Hastings Algorithm

- Metropolis-Hastings is the algorithm that determines whether to reject or accept the proposed jump

Current location: $\overrightarrow{p} = (\Omega_{M,i}, \Omega_{\Lambda,i}, w_i)$

Proposed jump: $\overrightarrow{n} = (\Omega_{M,j}, \Omega_{\Lambda,j}, w_j)$

- Calculate ratio of likelihoods (becomes more complicated if proposal density is not symmetric):

$$\alpha = \frac{\mathcal{L}_{\overrightarrow{n}}}{\mathcal{L}_{\overrightarrow{p}}}$$

- Acceptance criteria:

$\alpha \geq 1$ Always accept

$\alpha < 1$ Accept with probability α

Art of the MCMC

- Variable parameters:
  - selection of jump function
  - length of chain (typical: ~1000-10000)
  - number of Markov chains (typical: ~5)
  - length of burn-in (typical: ~5-10%)
  - simulated annealing

Jump Function

Current location: $\overrightarrow{p} = (\Omega_{M,i}, \Omega_{\Lambda,i}, w_i)$
Proposed jump: $\overrightarrow{n} = (\Omega_{M,j}, \Omega_{\Lambda,j}, w_j)$

- Simple jump function that takes a Gaussian with mean zero and variance σ^2 and adds it to the current location:

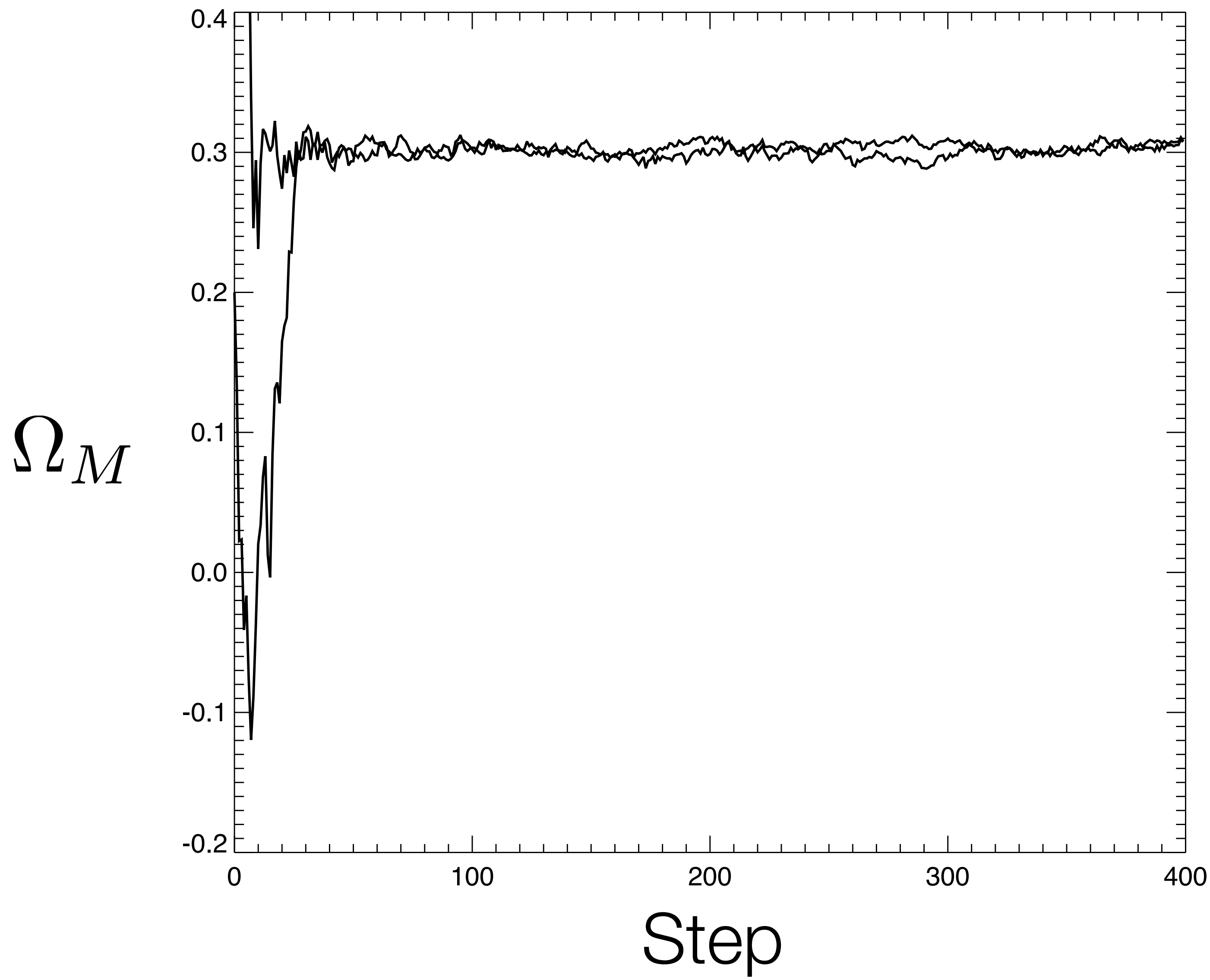
$$\overrightarrow{n} = \overrightarrow{p} + \mathrm{Gaussian}(\sigma^2)$$

- Adjust the variance of the Gaussian term based on the acceptance rate
  - High acceptance rate: Proposed jumps are very close to current location, increase the variance
  - Low acceptance rate: Many rejections means an inefficient chain (wasted computation time), decrease the variance
- Ideal acceptance rate: ~23%

Simulated Annealing

- Algorithm used during the burn-in process
- Gives the chain a “Temperature” which slowly cools during burn in to a final temperature which then remains constant through the rest of the Markov chain
  - High temperature: allows Metropolis-Hastings to accept “bad” jumps in order to travel across a large region of parameter space
- Effectively flattens the likelihood, allows more rapid movement across the parameter space during burn-in

Simulated Annealing

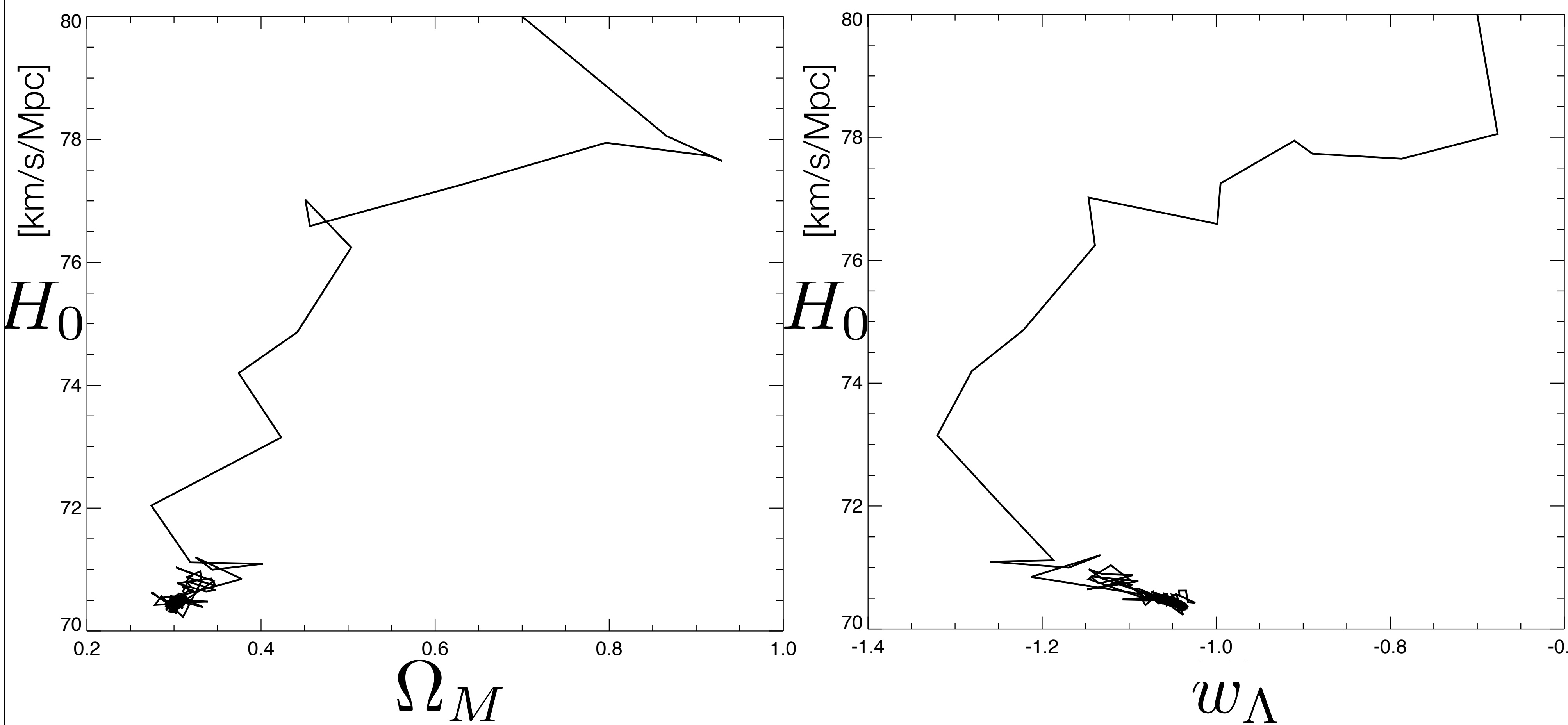


The Dataset: Supernovae

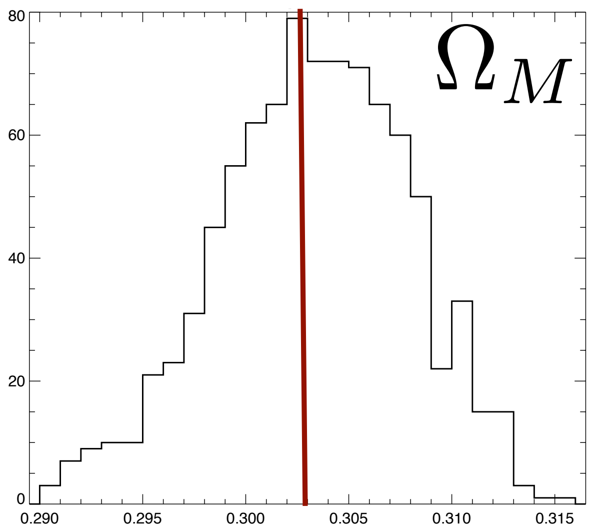
- Supernovae dataset consists of 288 SNe combined from several surveys (SDSS-II first year supernovae + ESSENCE + SNLS + HST + LOWZ) (Kessler et al. 2009)
- Computed χ^2 between luminosity distance calculated for the parameter values at that point in space and the luminosity distance from the measured distance modulus:

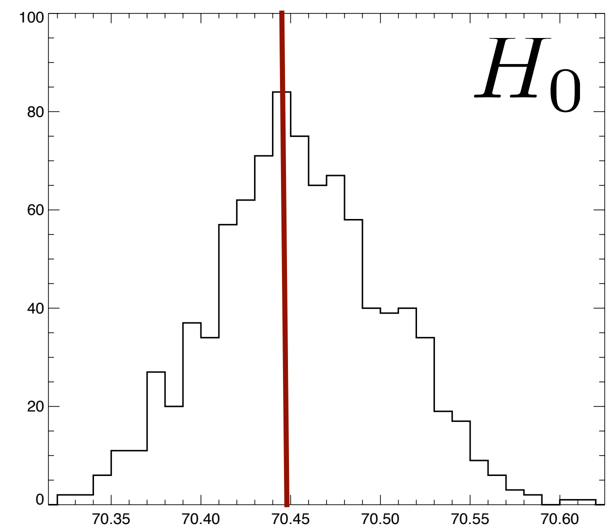
$$\chi^2 = -2 \ln \mathcal{L}$$

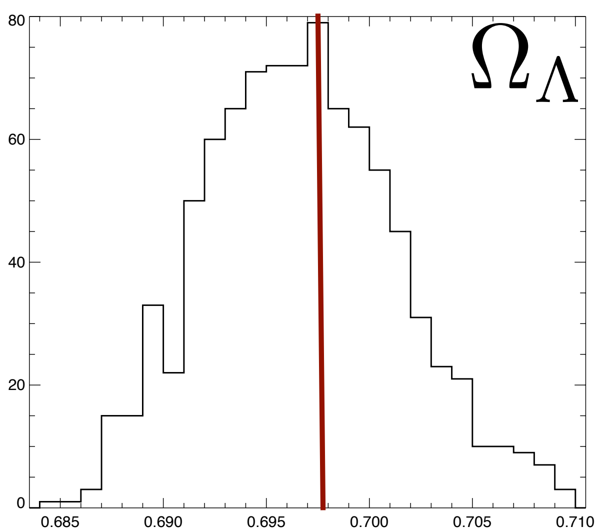
Visualizing the Parameter Space

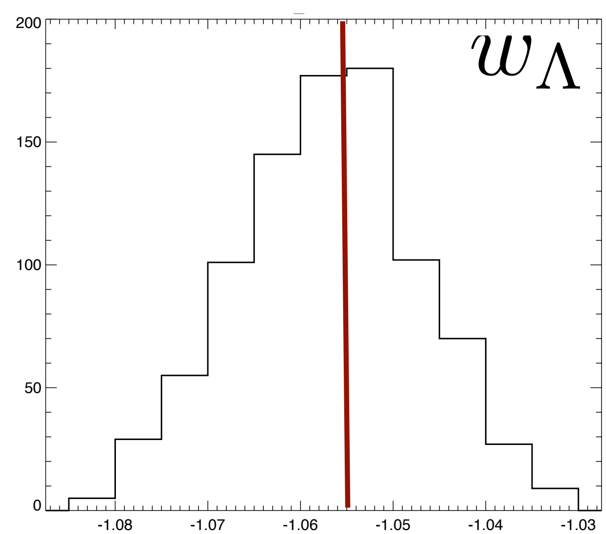


Reconstructed Distributions









Summary

- Markov Chain Monte Carlo is a powerful method for determing parameters and their posterior distributions, especially for a parameter space with many parameters

- Selection of jump function critical in improving the efficiency of the chain, i.e. reducing computation time

- A simple MCMC is able to recover the posterior distribution of several cosmological parameters by sampling from the likelihood function computed based on supernovae data