NTNU | Norwegian University of Science and Technology

THE HASTINGS ALGORITHM AT FIFTY

Paper discussion

Ola Rasmussen & Khanh Truong

February 12, 2024

Contents

Summary

- 1970, W. K. Hastings, Markov chain algorithms, sampling
- Stationary distribution is target distribution
- Hastings improved Metropolis, allowed asymmetry
- Bayesian posterior distributions

1. INTRODUCTION

- Study, characteristics, probability distribution $f(\cdot)$
- If simple, analytically
- If complicated, numerical integration, issues, accuracy, stability, and scalability to higher dimensions
- Solution, Monte Carlo algorithms, estimate features, samples
- Example:
  - Estimate mean
- Solution:
  - With samples $x_t \sim f$, we can estimate the mean of $f(x)$ as $\hat{\mu} = \frac{1}{T}\sum_{t=1}^{T} x_t$

1. INTRODUCTION

- Key challenge, efficiently generate samples
- Univariate case, many, inverse cumulative distribution function algorithm is popular
- Arbitrary multivariate distributions, challenging
- Rejection sampling attempts to solve this problem
- Problem, how to select good $g(x)$ that is easy to sample from

1. INTRODUCTION

- Markov chain Monte Carlo algorithms
- Markov chain $\{x_t\}_{t=1}^{T}$, transition kernel $K\left(x_t \mid x_{t-1}\right)$
- Samples $\{x_t\}$, converge, stationary distribution is target distribution
- Burn in, isn't stationary at the beginning
- Particularly popular in Bayesian inference

1. INTRODUCTION

- Metropolis, 1953, built on rejection sampling
- Samples candidate $\tilde{x}$ from proposal density (symmetric)
- Set $x_t = \tilde{x}$ with probability $\alpha\left(\tilde{x} \mid x_{t-1}\right) = min\left\{1, \frac{f(\tilde{x})}{f(x_{t-1})}\right\}$ and $x_t = x_{t-1}$ otherwise
- Big problem, only symmetric distributions

1. INTRODUCTION

- Hastings, 1970
- Improved, asymmetry
- Set $x_t = \tilde{x}$ with probability $\alpha\left(\tilde{x} \mid x_{t-1}\right) = min\left\{1, \frac{f(\tilde{x})}{f(x_{t-1})} \cdot \frac{g(x_{t-1}|\tilde{x})}{g(\tilde{x}|x_{t-1})}\right\}$ and $x_t = x_{t-1}$ otherwise
- Most popular MCMC algorithm
- Most common approach to modern Bayesian computation

1. INTRODUCTION

- Top 10 most important algorithms of the 20th century
- Hydrogen bomb, "mathematical analyzer, numerical integrator, and computer"
- Didn’t mention Bayesian statistics, but is most prominent today
- Made Bayesian statistics feasible

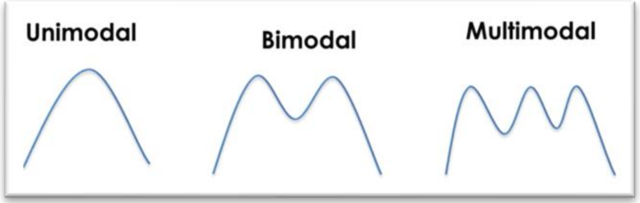
2.1. Overview

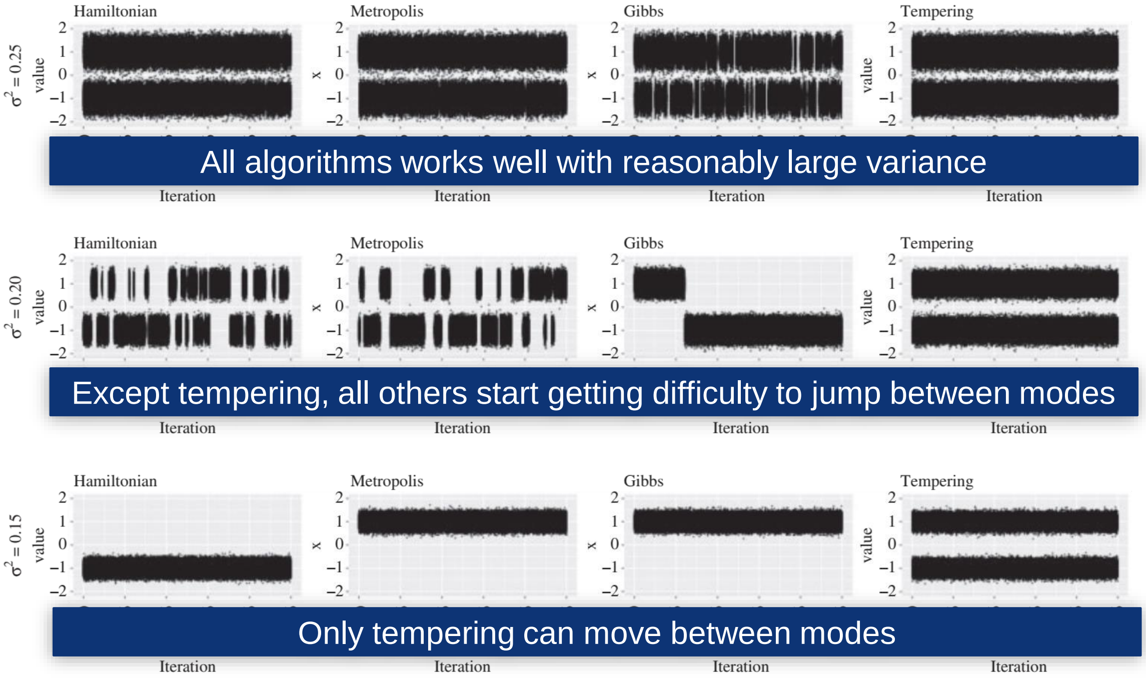
- How to choose a good proposal having high computational efficiency?
  - **(i)** Computational cost per iteration of the sampler
  - **(ii)** Mixing rate of the Markov chain $\{x_t\}$
- (i), dependent, cost of sampling, calculating acceptance probability
- (ii), samples are not independent, x_t and $x_{t+\Delta}$ are correlated
- Slow mixing, correlation between x_t and $x_{t+\Delta}$ decreases slowly, samples contribute less information
- Effective sample size

2. Extensions

- Gibbs
- Metropolis-within-Gibbs
- Blocking
- Adaptive algorithms
- Gradient-based algorithms
  - Metropolis-adjusted Langevin
  - Hamiltonian Monte Carlo

3. Challenging – Multimodal targets





Solutions for multimodal:

- simulated
- tempering
- equi-energy sampler
- split–merge samplers
- birth–death algorithm

Hasting algorithm can fail to move among modes of the multimodal distribution.

3. Challenging – Intractable likelihoods

Example of intractable likelihoods:

- g-and-k distribution:

$$Q(u;A,B,g,k) = A + B\left[1 + c\frac{1-\exp\{-g\Phi(u)\}}{1+\exp\{-g\Phi(u)\}}\right]\{1+\Phi(u)^2\}^k\Phi(u)$$

Q(u;A,B,g,k)=A + B\left[1+c\dfrac{1-\exp\{-g\Phi(u)\}}{1+\exp\{-g\Phi(u)\}}\right]\{1+\Phi(u)^2\}^k\Phi(u)

- Two-dimensional summary statistics:

$$x_1, \dots, x_n \overset{\text{iid}}{\sim} \mathrm{N}(\theta, \sigma^2)\,,$$
$$S(x_1, \dots, x_n) = (\mathrm{med}(x_1, \dots, x_n), \mathrm{mad}(x_1, \dots, x_n))\,,$$

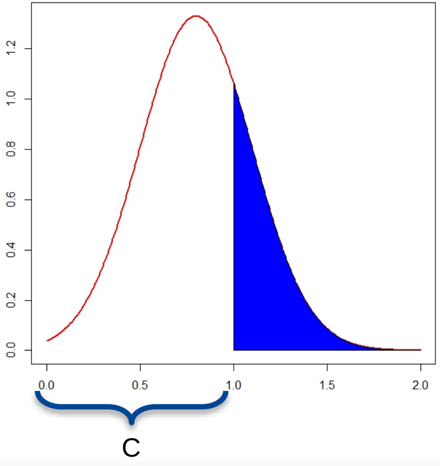
Solutions for incomputable likelihoods:
- auxiliary variable scheme
- rejection sampling
- Pseudo-marginal Metropolis Hastings

=> We need unbiased estimate of the likelihood in the acceptance probability

Likelihood functions can be intractable, meaning it is not computable even up to a normalizing constant.

NTNU

3. Challenging – Distributions with constrained support



Solution for constrained support distribution:

- ignore the constraint and simply reject proposals falling outside of C;
- reparameterize to an unconstrained space before running the sampler
- Gibbs sampling with the conditional posterior distributions truncated to reflect the constraint

Hard to implement appropriate proposal distribution with the same support

NTNU

Bonus: The effect of proposal selection in MCMC

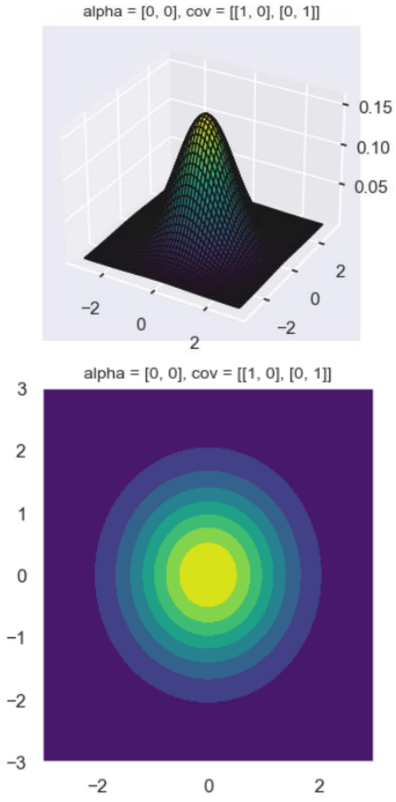
Given a target distribution,

- **What is the difference we select one proposal over another?**
- **Is there a "better" proposal distribution over some others?**

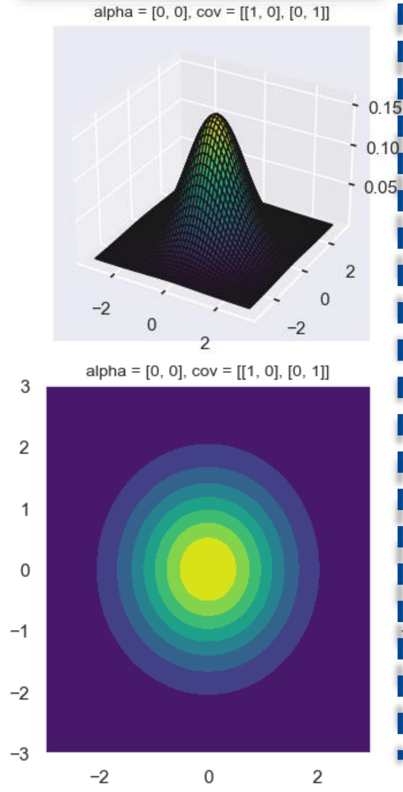
Case 1: Target and proposal are both Gaussian

Proposals

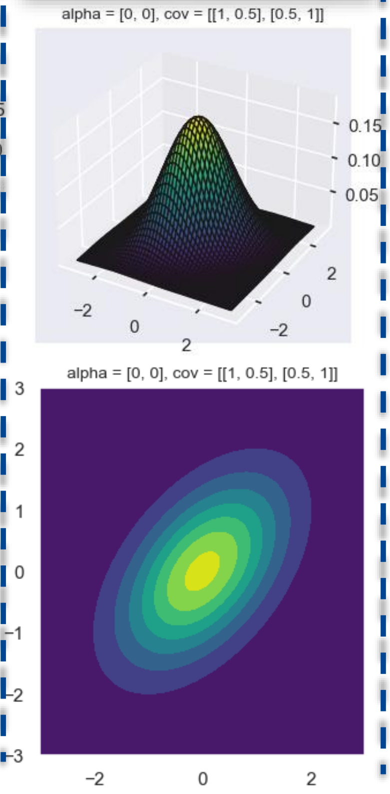
Target



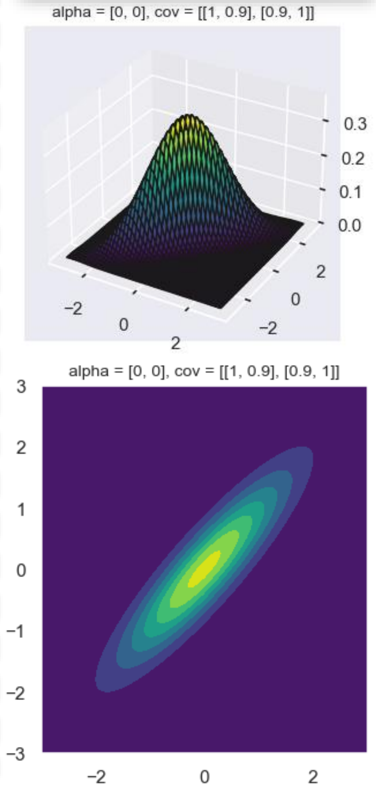
Cov = [[1, 0], [0, 1]]



Cov = [[1, 0.5], [0.5, 1]]

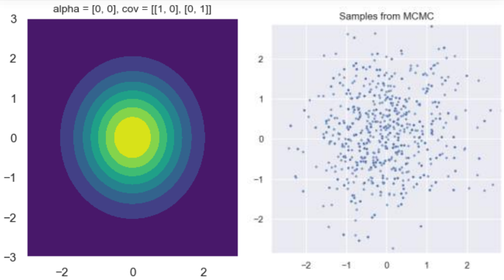


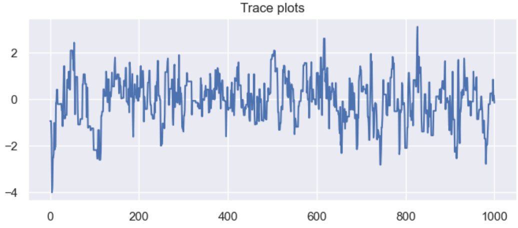
Cov = [[1, 0.9], [0.9, 1]]

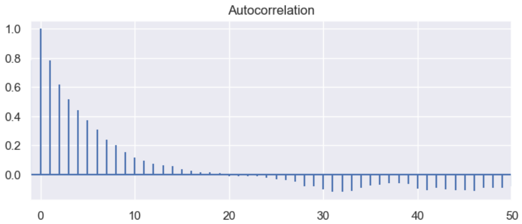


Case 1: Target and proposal are both Gaussian

Cov = [[1, 0], [0, 1]]

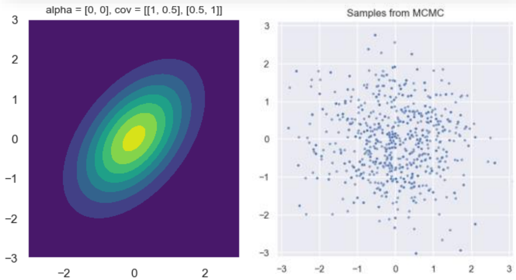


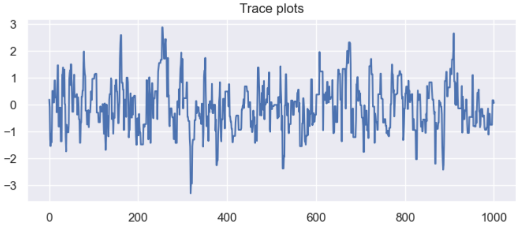


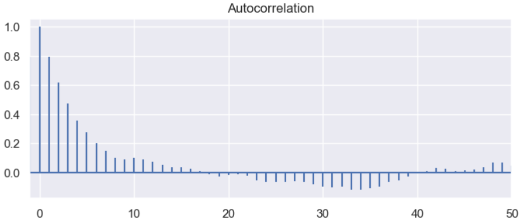


Acceptance rate: 57%

Cov = [[1, 0.5], [0.5, 1]]

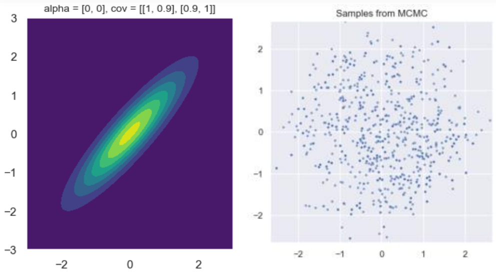


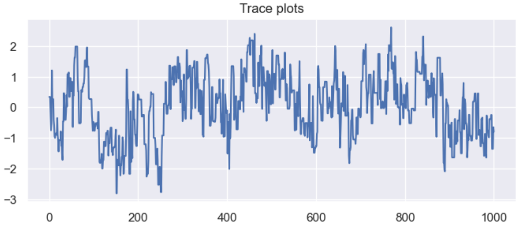


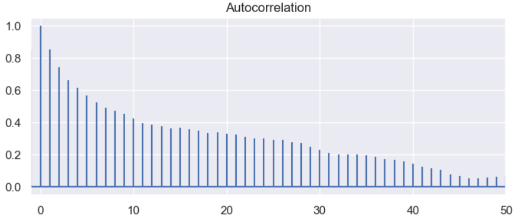


Acceptance rate: 55%

Cov = [[1, 0.9], [0.9, 1]]



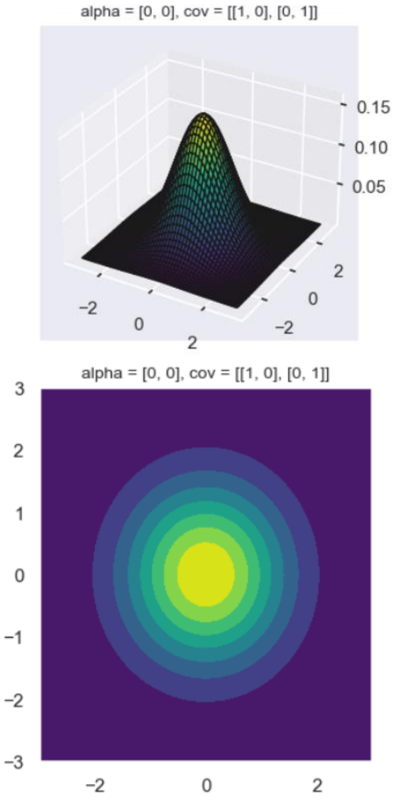




Acceptance rate: 58%

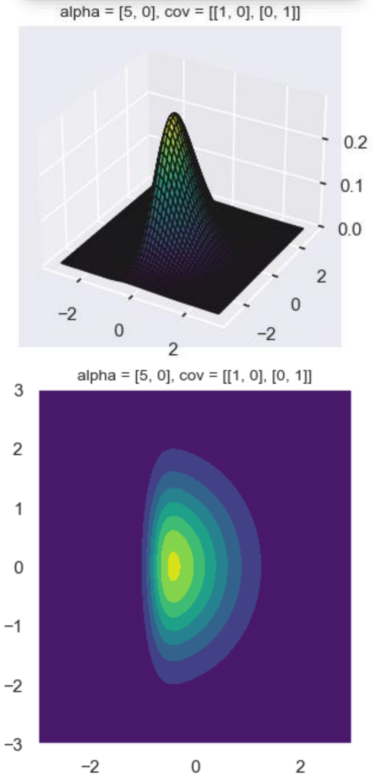
Case 2: Target is Gaussian, proposals are skew Gaussian

Target

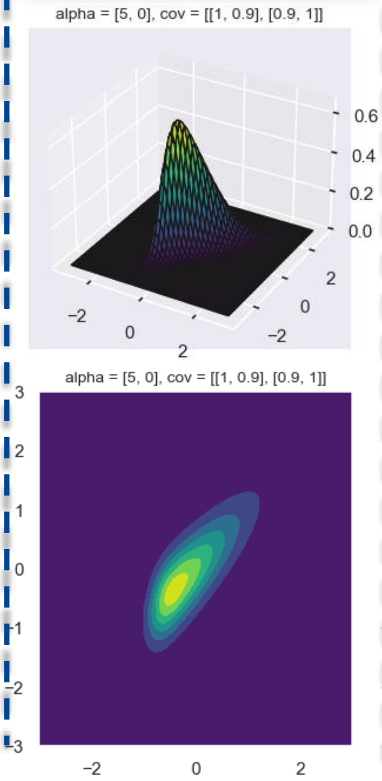


Proposal

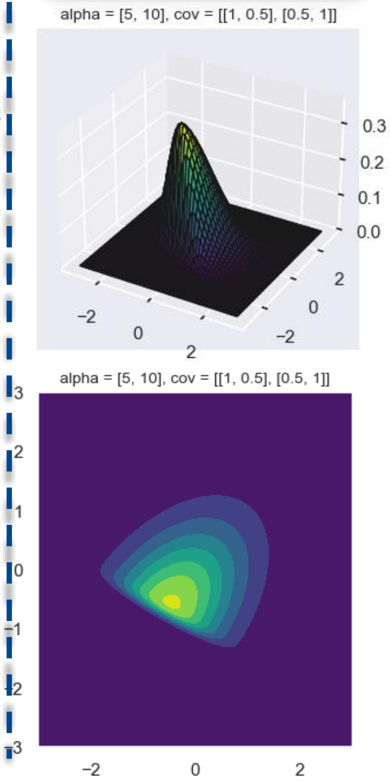
shape = [5, 0],
cov = [[1, 0], [0, 1]]



shape = [5, 0],
cov = [[1, 0.9], [0.9, 1]]

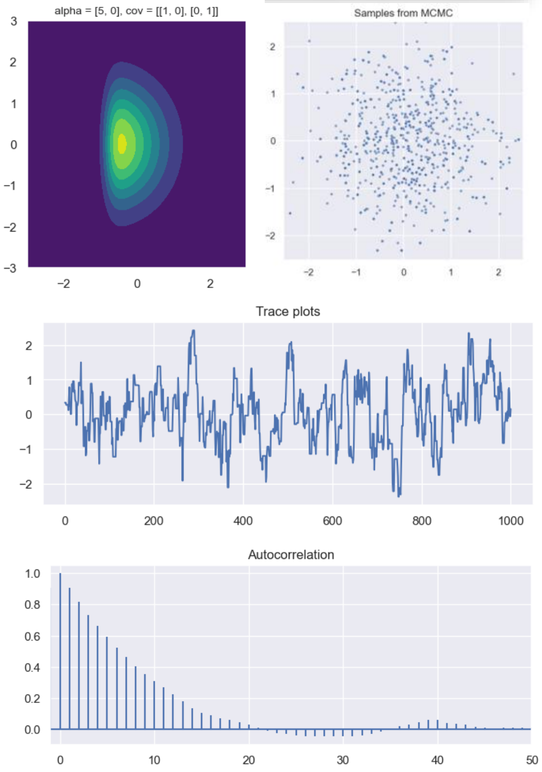


shape = [5, 10],
cov = [[1, 0.5], [0.5, 1]]



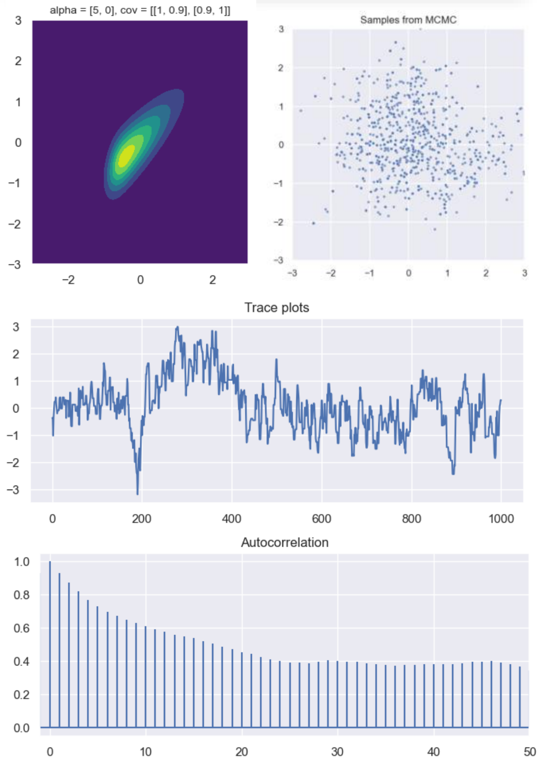
Case 2: Target is Gaussian, proposals are skew Gaussian

shape = [5, 0], cov = [[1, 0], [0, 1]]



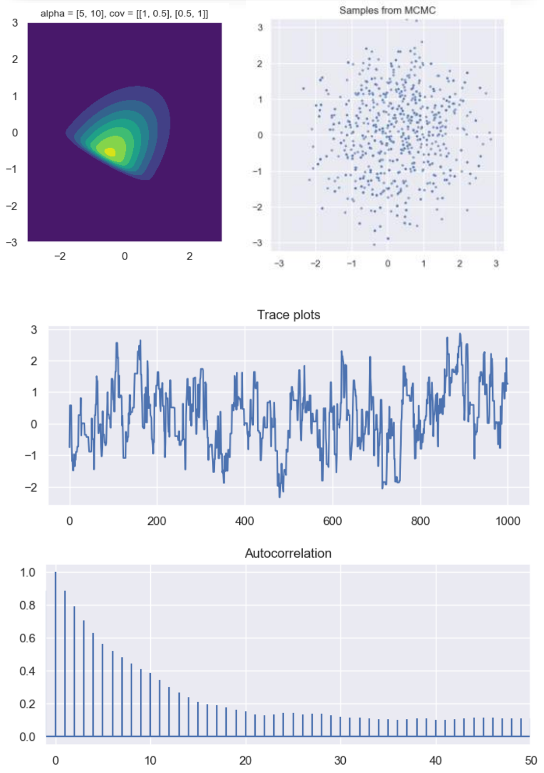
Acceptance rate: 56%

shape = [5, 0], cov = [[1, 0.9], [0.9, 1]]



Acceptance rate: 62%

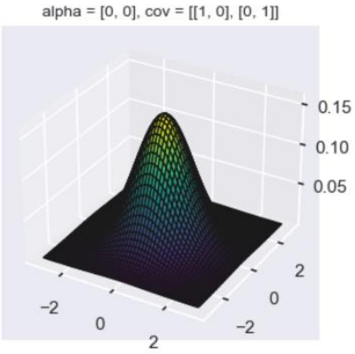
shape = [5, 10], cov = [[1, 0.5], [0.5, 1]]

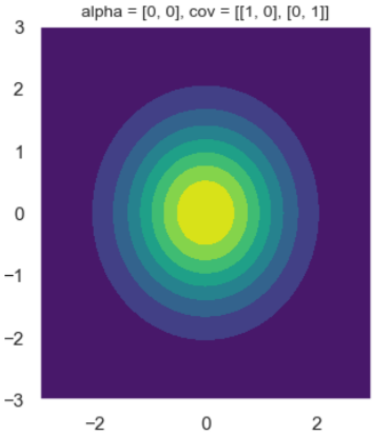


Acceptance rate: 56%

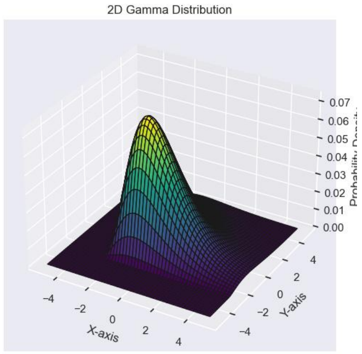
Case 3: Target is Gaussian, proposals are gama / log

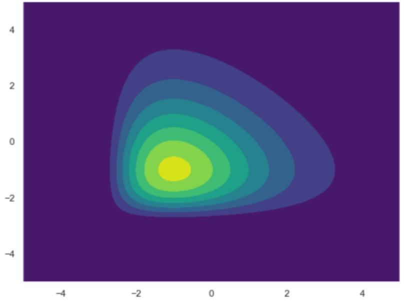
Target

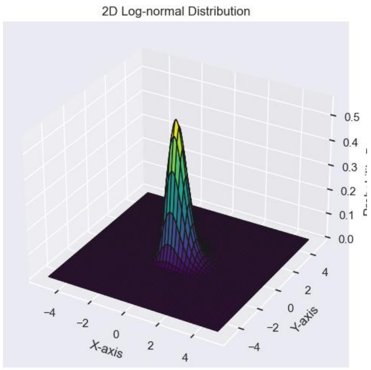


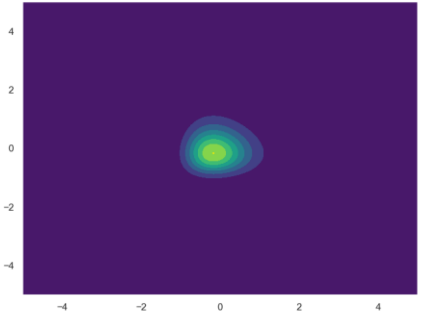


Proposal



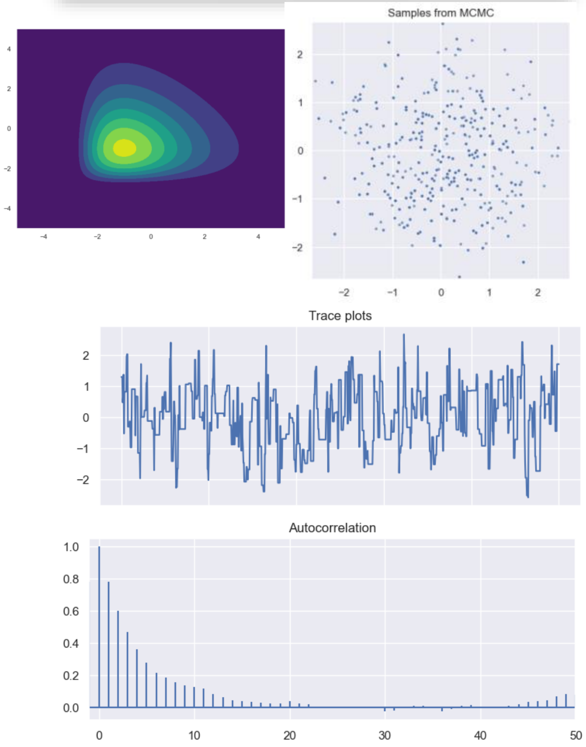






Case 3: Target is Gaussian, proposals are gama / log

Gama distribution

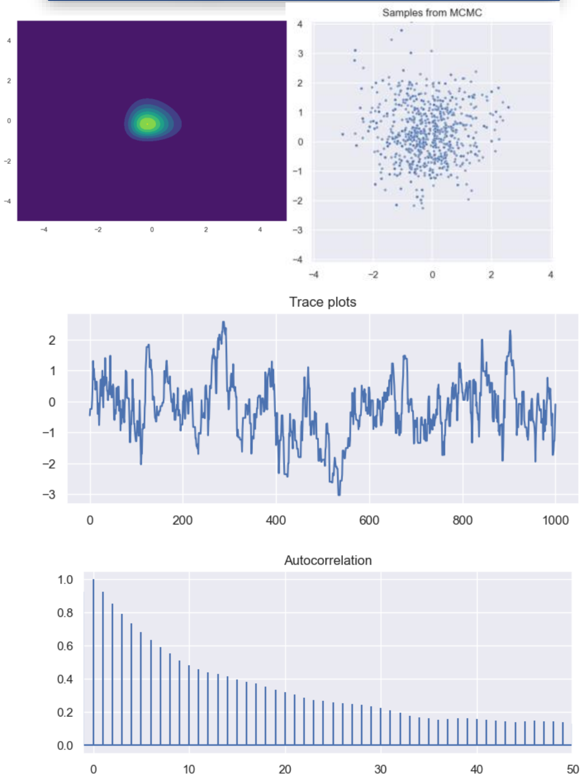


Acceptance rate: 37%

Note that the support of gama and log normal are not the same as Gaussian.

But still, they give reasonable result.

Lognormal distribution



Acceptance rate: 66%

The effect of proposal selection in MCMC

Given a target distribution,

- What is the difference we select one proposal over another?

-> It seems that the more “similar” to the target the proposal is, the better the samples are (in term of autocorrelation)

- Is there a “better” proposal distribution over some others?

-> Choose the best-knowledge proposal that is similar to target , i.e. the prior distribution?

Questions

1. Adaptive algorithms
2. What are some advantages of gradient-based algorithms compared to other MCMC methods?
3. What is the best proposal distribution? Should we choose the one that is similar to prior distribution?