LECTURE 7: Markov Chain Monte Carlo

Modeling and Simulation 2

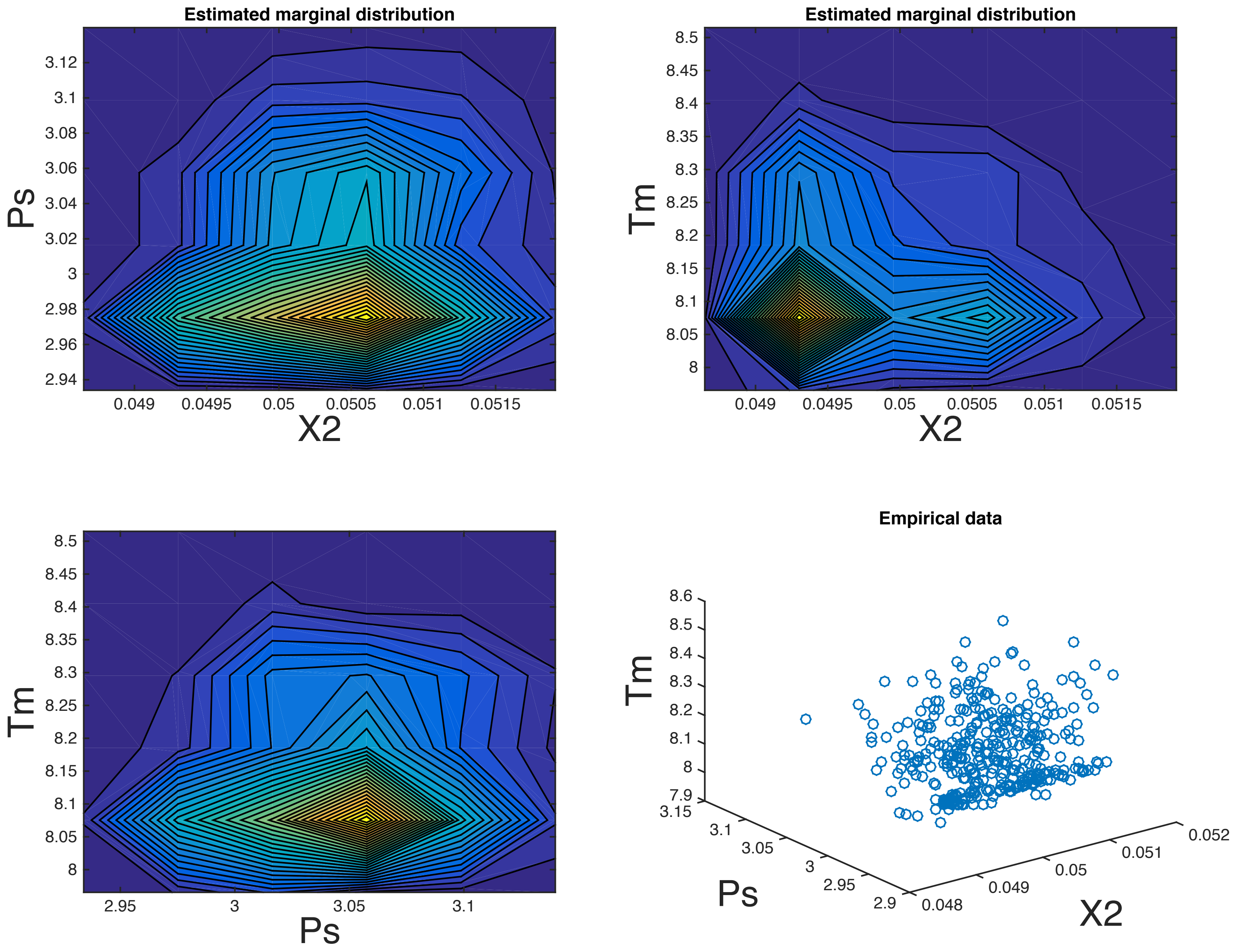
Daniel Georgiev

OUTLINE

- Review
- Markov Chain Monte Carlo
- Metropolis-Hastings algorithm
    - reversible Markov chain
    - perfect balance
    - derivation of transition probabilities
    - thinning
    - burn in
    - example

CASE STUDY DATA

Histogram data for rainy days



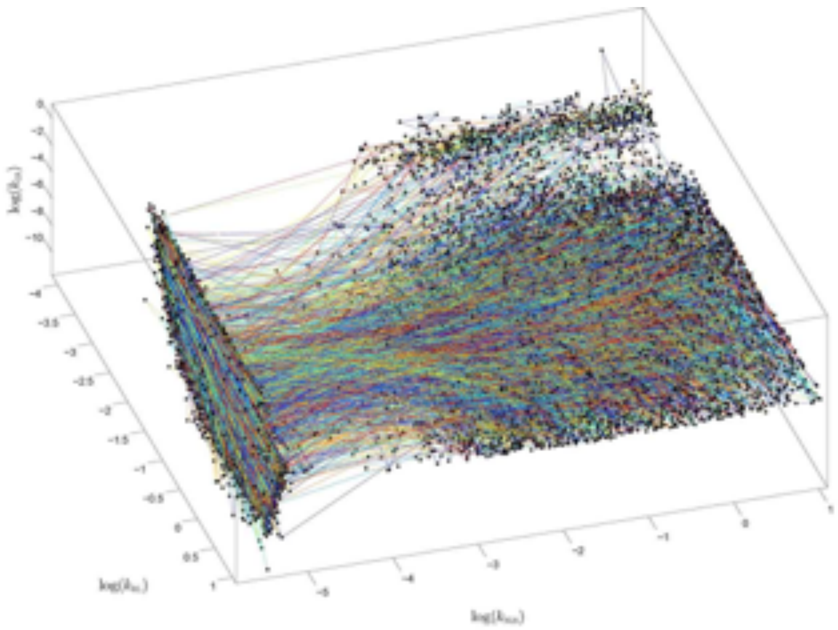
Estimated marginal distribution

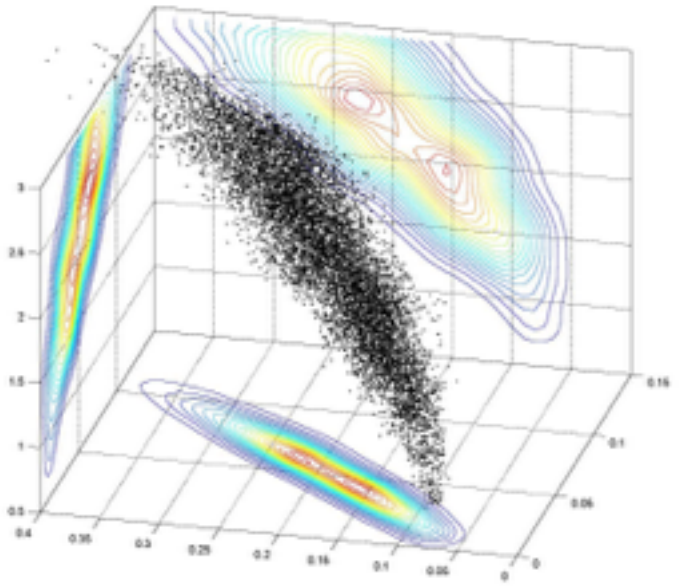
Estimated marginal distribution

Empirical data

MARKOV CHAIN MONTE CARLO (MCMC)

The inverse method allows sampling 1d RVs with arbitrary distributions. Many real world systems include correlated inputs. Markov Chain Monte Carlo (MCMC) is a method for sampling arbitrary multivariate RVs.





REVERSIBLE MARKOV CHAIN

Kolmogorov's criterion: A Markov chain with a transition probability P is said to be reversible if

$$p_{j_1,j_2}p_{j_2,j_3}\cdots p_{j_{n-1},j_n}p_{j_n,j_1} = p_{j_1,j_n}p_{j_n,j_{n-1}}\cdots p_{j_3,j_2}p_{j_2,j_1}$$

for all finite sequences of states.

Same holds for continuous time MC, only propensities substituted for probabilities.

PERFECT BALANCE

Theorem: A reversible Markov Chain satisfies

$$\pi_i p_{i,j} = \pi_j p_{j,i}$$

Hence, an ergodic Markov Chain with a probability transition matrix

$$\frac{\pi_i}{\pi_j} = \frac{p_{j,i}}{p_{i,j}}$$

converges with probability one to the stationary distribution π

TRANSITION PROBABILITIES

Approach: Separate the transition probability into a proposition probability and an acceptance probability

$$P\left(x_i|x_j\right) = pr\left(x_i|x_j\right) ac\left(x_i|x_j\right)$$

From detailed balanced we obtain

$$\frac{ac(x_i|x_j)}{ac(x_j|x_i)} = \frac{P(x_i)pr(x_j|x_i)}{P(x_j)pr(x_i|x_j)}$$

TRANSITION PROBABILITIES

Approach: Separate the transition probability into a proposition probability and an acceptance probability

$$P\left(x_i|x_j\right) = pr\left(x_i|x_j\right) ac\left(x_i|x_j\right)$$

From detailed balanced we obtain

$$\frac{ac(x_i|x_j)}{ac(x_j|x_i)} = \frac{P(x_i)pr(x_j|x_i)}{P(x_j)pr(x_i|x_j)}$$

One choice for the acceptance function is Metropolis choice

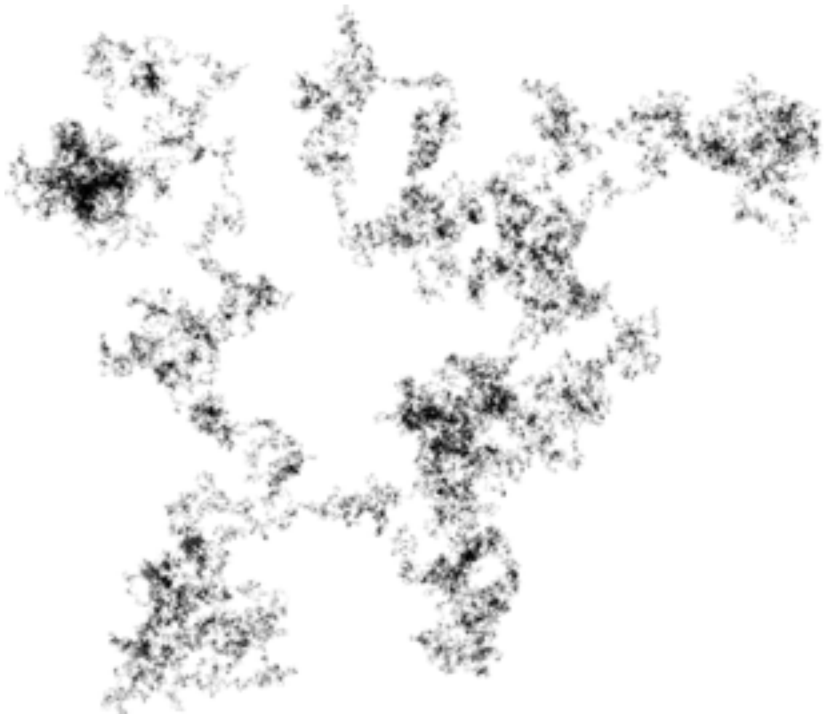
$$ac\left(x_i|x_j\right) = \min\left(1, \frac{P(x_i)pr(x_j|x_i)}{P(x_j)pr(x_i|x_j)}\right)$$

PROPOSAL FUNCTION

The proposal distribution is usually Gaussian, i.e.,

$$Q(x|x_i) = \frac{1}{(2\pi)^{d/2}} e^{-\frac{1}{2}(x-x_i)'(x-x_i)}$$

For a uniform target distribution, where the *P(xi) = P(xj),* MCMC reduces to a continuous random walk.



TRANSITION PROBABILITIES

Gaussian proposition function: for an unbiased random walk the acceptance function reduces to

$$ac\left(x_i|x_j\right) = \min\left(1, \frac{P(x_i)}{P(x_j)}\right)$$

Hence, if the proposed state has a higher probability, it is always accepted. If the proposed state has a lower probability, it is accepted at a rate proportional to the two state probability quotient.

AUTOCORRELATION

Monte Carlo method requires generation of iid samples, MCMC samples are not independent. To generate approximately iid samples, discard d samples for every 1 kept sample. The number d depends on the sample space, the standard deviation the proposal function, and the target distribution. A test whether d is large enough is given by the autocorrelation function.

$$autocorr\left(\tau\right) = \frac{\mathbb{E}[(X_t - \mu)(X_{t+\tau} - \mu)]}{\sigma^2}$$

Where a sample autocorrelation using circular convolution is

$$autocorr\left(i\right) = \sum_{j=0}^{N-1} x_j x_{j+i}^*$$

The practice of discarding samples is called thinning.

BURN IN

A Markov chain takes some time to reach the station distribution. Although convergence to the station distribution is guaranteed from any initial state, the amount of time it takes depends on many parameters. Standard practice is to discard enough of the initial samples to ensure samples are only drawn from the target distribution. The discard period is called the burn in period and can again be computed from the autocorrelation function. In other words, the samples after the burn in period should be independent of the initial state.

MARKOV CHAIN MONTE CARLO (MCMC)

MCMC (Metropolis-Hastings Algorithm): Consider a RV $X : \Omega \to \mathbb{R}^d$

1. Initialisation (x_0) - select an initial value of X and let $i = 0$.
2. Proposal (x_p) - propose a new sample value of X from the last sampled value x_i according to the conditional distribution $Q(x/x_i)$. The distribution may be the multivariate distribution
$$Q(x|x_i) = \frac{1}{(2\pi)^{d/2}} e^{-\frac{1}{2}(x-x_i)'(x-x_i)}$$
that can be sampled by generating d independent normal RVs.
3. Acceptance (x_{i+1}) - compute the value $\alpha(x_p, x_i)$ by the Metropolis formula
$$\alpha(x_p, x_i) = \min\left(1, \frac{f(x_p)Q(x_i|x_p)}{f(x_i)Q(x_p|x_i)}\right)$$
accept the proposed value with probability $\alpha(x_p, x_i)$, i.e., if $U < \alpha(x_p, x_i)$, $x_{i+1} = x_p$, $i = i + 1$, and go to step 2. Otherwise, reject x_p and go to 2. U is uniformly distributed on $[0,1]$.

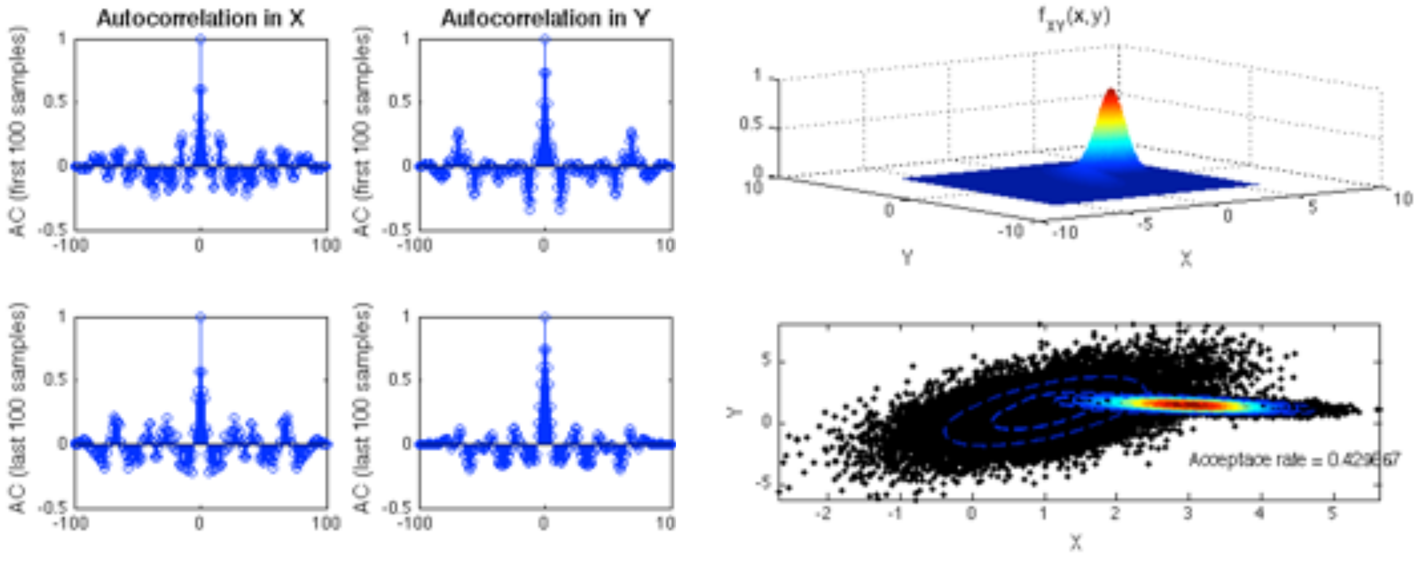
MARKOV CHAIN MONTE CARLO THEOREM

Consider the Markov chain $\{X_i$, i = 1,2,3...$\}$ generated by the Metropolis-Hastings Algorithm with the acceptance function

Then the probability density of X_i converges and is equal to f, i.e., for sufficiently large i, the generated states are approximately sampled from the desired distribution.

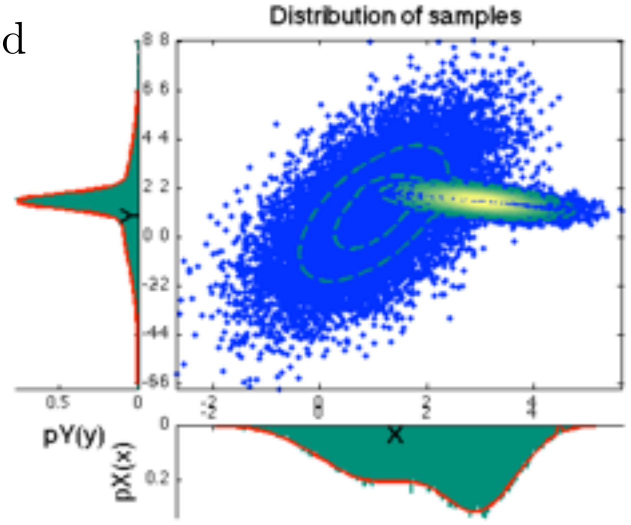
MCMC EXAMPLE

MCMS generates autocorrelated samples. Monte Carlo requires IID samples. Discarding samples reduces the autocorrelation.



Acceptance function results in higher probability values being sampled more often. This increases the sampling efficiency. Note, the computationally expensive part of a simulation is the deterministic function evaluation, not the RV generation.

The Markov process is guaranteed to have a stationary distribution that is equal to the target distribution. Depending on the mixing rate, however, it takes some time for the Markov process to converge.


Distribution of samples
Y
X
pY(y)
pX(x)