# PHYS 142/242

## Lecture 14: Markov chain Monte Carlo (Continued)

Javier Duarte — February 10, 2025

# Markov process and chain

A ***Markov process*** is a set of probabilistic rules that tell us how to choose a new state of the system, based only on the system's current state. If the system is currently in state $n$ then the probability of choosing state $m$ on the next step is given by the “transition probability” $P(m \mid n)$.

By repeatedly applying the Markov process, we move the system through a random sequence of states, $\{n^{(0)}, n^{(1)}, n^{(2)}, \ldots\}$ where $n^{(k)}$ denotes the state on step $k$.

This random sequence is called a ***Markov chain***.

Because the system must transition to some state,

$$\sum_m P(m \mid n) = 1 \quad \text{for all } n \in \{0,1,\ldots\}\,.$$

# Markov process and chain

The probability to be found in state $m$ at step $k+1$ depends only on step $k$

$$p_m^{(k+1)} = \sum_n P(m\,|\,n)\, p_n^{(k)}$$

or in terms of the transition matrix,

$$\begin{bmatrix} p_0^{(k+1)} \\ p_1^{(k+1)} \\ \vdots \end{bmatrix} = \begin{bmatrix} P(0\,|\,0) & P(0\,|\,1) & \cdots \\ P(1\,|\,0) & P(1\,|\,1) & \cdots \\ \vdots & \vdots & \end{bmatrix} \begin{bmatrix} p_0^{(k)} \\ p_1^{(k)} \\ \vdots \end{bmatrix},$$

The vector of stationary probabilities $[\pi_0, \pi_1, \pi_2, \ldots]$ is an eigenvector of the transition matrix, with eigenvalue 1.

We will use Markov processes to model thermodynamic systems, such that a stationary distribution represents the distribution of thermodynamic microstates in thermal equilibrium

# Detailed balance

Detailed balance requires that the rate at which the $n \to n+1$ transition occurs is the same as the rate at which the opposite transition occurs $n+1 \to n$.

$$P(n+1\,|\,n)\,\pi_n = P(n\,|\,n+1)\,\pi_{n+1} \quad \text{for all} \quad n \in \{0,\ldots,N\}\,.$$

If this holds for every pair of states, then the probability distribution is necessarily stationary.

$$\begin{bmatrix} \pi_0 \\ \pi_1 \\ \vdots \end{bmatrix} = \begin{bmatrix} P(0\,|\,0) & P(0\,|\,1) & \cdots \\ P(1\,|\,0) & P(1\,|\,1) & \cdots \\ \vdots & \vdots & \end{bmatrix} \begin{bmatrix} \pi_0 \\ \pi_1 \\ \vdots \end{bmatrix},$$

Note: not all Markov chains obey detailed balance!

# 2D Ising model

The Hamiltonian is

$$E(\sigma) = -J \sum_{\langle ij \rangle} \sigma_i \sigma_j$$

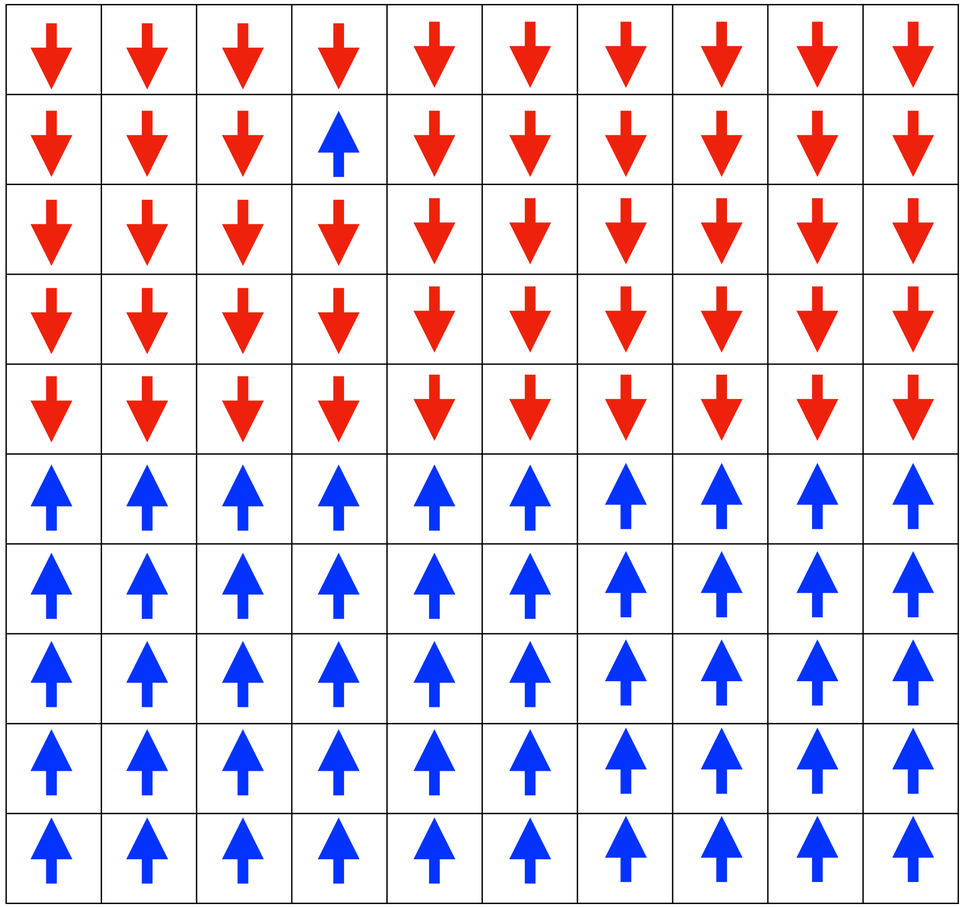
where $\sigma_k \in \{-1, +1\}$ represents the spin of each site. The probability of any given state is

$$p(\sigma) = \frac{1}{Z} \exp\left(-\frac{E(\sigma)}{k_{\mathrm{B}} T}\right)$$

where

$$Z = \sum_{\sigma} \exp\left(-\frac{E(\sigma)}{k_{\mathrm{B}} T}\right).$$

With $10 \times 10$ sites $2^{100} \approx 10^{30}$ possible states

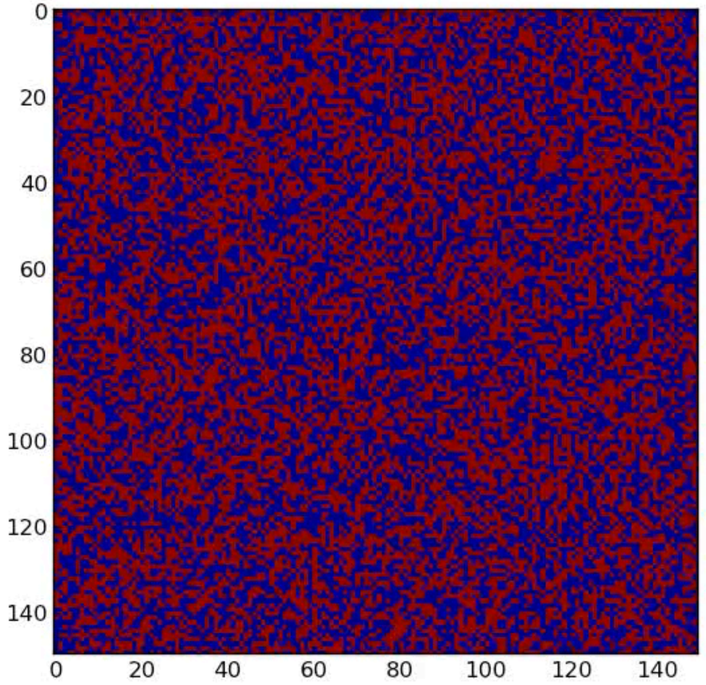


# Ising model phase transition

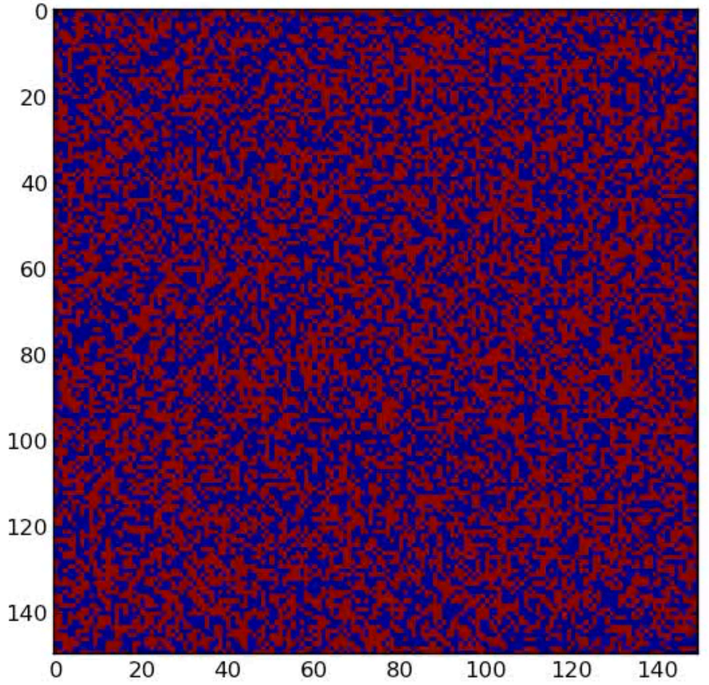
With $150 \times 150$ lattice, $2^{22500} \approx 10^{6773}$ possible states!

Phase transition occurs at a critical temperature: ordered phase at low $T$, disordered phase at high $T$. How can we compute this?

High $T$




Low $T$



# Ising model phase transition

Often interested in quantities like the average magnetization, i.e. first we average over the lattice

$$\sigma_{\text{ave}} = \frac{1}{|\Lambda|} \sum_{k \in \Lambda} \sigma_k$$

then we take a thermodynamic average

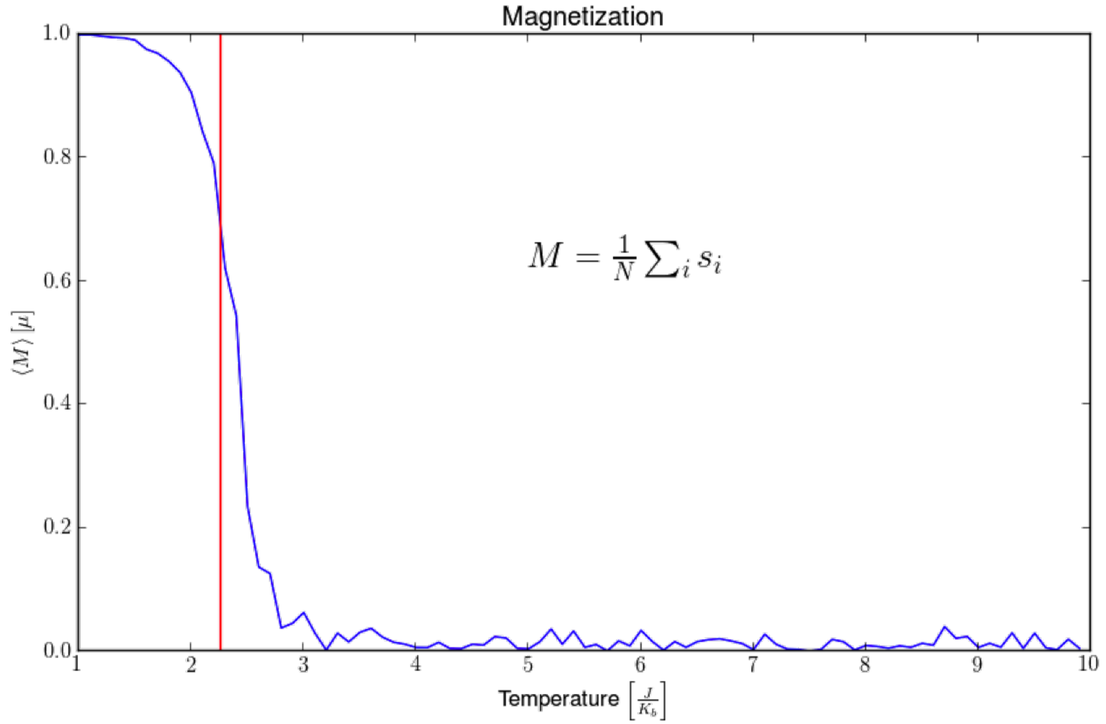
$$\langle \sigma_{\text{ave}} \rangle = \sum_{\sigma} \sigma_{\text{ave}} p(\sigma)$$

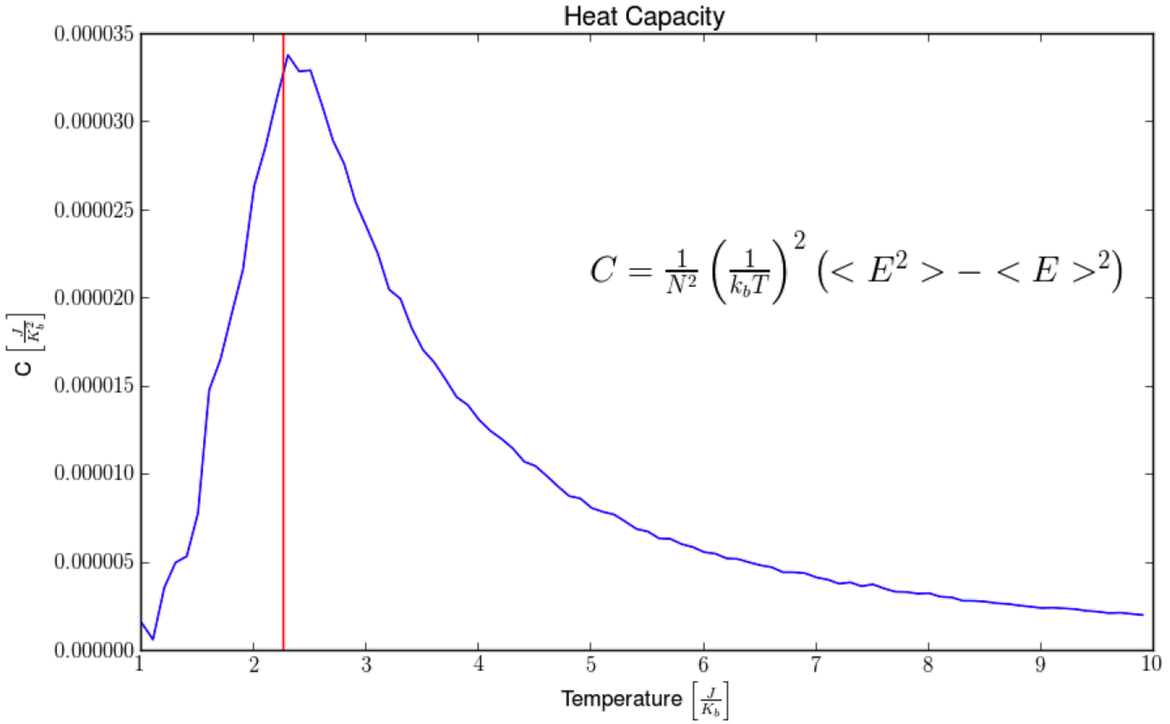
If we plot this versus temperature, we can see there is some critical temperature $T_{\text{c}}$where the system goes from being ferromagnetic (spins are aligned) to paramagnetic (spins are not aligned)

# Ising model phase transition

In 2D, we can actually analytically calculate the critical temperature as solved in L. Onsager, Phys. Rev. 65, 117 (1944)

$$\frac{k_{\mathrm{B}}T_{\mathrm{c}}}{J} = \frac{2}{\ln(1+\sqrt{2})} \approx 2.269$$



# Markov chain MC

Trying to calculate the partition function directly by summing over all states is prohibitively time-consuming.

Instead the MCMC method selectively samples the states.

To accomplish this, we design a Markov process whose stationary distribution is equal to the correct distribution of probabilities.

Once we have this Markov process, we can generate a long Markov chain and calculate moving averages of our desired quantities like $\langle \sigma_{\text{ave}} \rangle$.

If the Markov chain is long enough, this will converge to the true expectation value

# Metropolis-Hastings algorithm

Design a Markov process to match a given stationary distribution $\pi_n$: ***Metropolis-Hastings algorithm*** based on detailed balance

$$P(n\,|\,m)\,\pi_m = P(m\,|\,n)\,\pi_n \quad \text{for all } m, n$$

1. On step $k$, the system is in state $n$. Randomly choose a candidate state $m$ by making an unbiased random step through the space of possible states.

2. Compare the probabilities $\pi_n$ and $\pi_m$

   If $\pi_m \geq \pi_n$, accept the candidate

   If $\pi_m < \pi_n$, accept the candidate with probability $\pi_m/\pi_n$. Otherwise, reject the candidate

3. If the candidate is accepted, the state in step $k+1$ is $m$. Otherwise, the state in step $k+1$ remains $n$

4. Repeat

# Checking detailed balance

Let's verify that the stationary distribution of the Markov process satisfies detailed balance.

Consider two states $a, b$ with $\pi_a \leq \pi_b$.

Starting from $a$, suppose we choose candidate step $a \to b$ with some probability $q$. Then, the transition probability for $a \to b$ is $q$ times the acceptance probability 1.

Starting from $b$, we will choose candidate step $b \to a$ with the same probability $q$ (assuming this is unbiased). Then, the transition probability for $b \to a$ is $q$ times the acceptance probability $\pi_a / \pi_b$.

$$\left.\begin{aligned} P(b \mid a) &= q \\ P(a \mid b) &= q\frac{\pi_a}{\pi_b} \end{aligned}\right\} \quad \Rightarrow \quad P(a \mid b)\,\pi_b = P(b \mid a)\,\pi_a \quad \text{(detailed balance)}$$

# Stepping through state space

One way of thinking about the Metropolis algorithm is that it takes a scheme for performing an unbiased random walk through the space of possible states (represented by our candidate choices), and converts it into a scheme for performing a biased random walk

The biased random walk corresponds to a Markov process with the stationary distribution we are interested in

# Applying MCMC to the 2D Ising model

To apply the MCMC method to the Ising model, we design a Markov process using the Metropolis algorithm as follows

1. On step $k$, randomly choose one of the spins $i$ and consider flipping it $\sigma_i \rightarrow -\sigma_i$
2. Calculate the change in energy that would result from flipping spin $i$, i.e. the quantity:

$$\Delta E = -\left[ J \sum_{\langle ij \rangle} \sigma_j \right] \Delta\sigma_i,$$

where $\Delta\sigma_i = -2\sigma_i$ is the change in $\sigma_i$ due to the spin flip

If $\Delta E \leq 0$, accept the spin flip

If $\Delta E > 0$, accept the spin flip with probability $\exp(-\Delta E/k_{\mathrm{B}}T)$. Otherwise, reject the flip.

3. Update the moving average of $\langle \sigma_{\mathrm{ave}} \rangle$ (or whatever quantity we are interested in).

4. Repeat.

# Practical considerations for MCMC

- Several practical considerations
  - Acceptance rate: the rate at which we accept proposals is important. We want this to be relatively high (to reduce total running time), but not always 1 (which may mean we're not sufficiently sampling the full space). Typical values are 15% to 50%
  - Burn in
  - Error and (integrated) autocorrelation time

# Burn-in

"Burn-in": common to throw out the first few states of a Markov chain, maybe the first 100 or the first 1000

The idea is to get rid of "transient behavior" connected to an improbable intial configuration

Equivalent: pick an initial state near equilibrium (i.e. not unlikely)

Autoregressive series

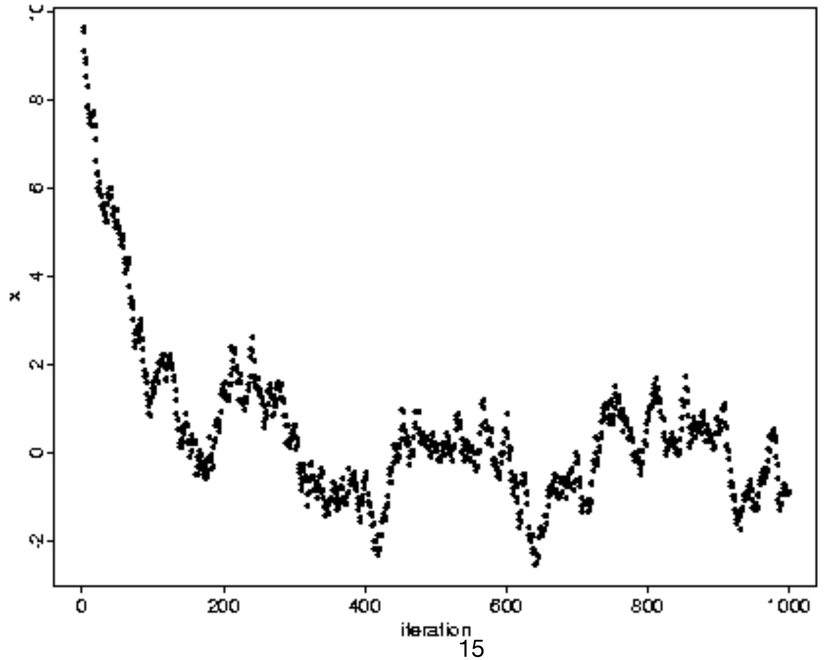
$$x_{n+1} = rx_n + e_n$$

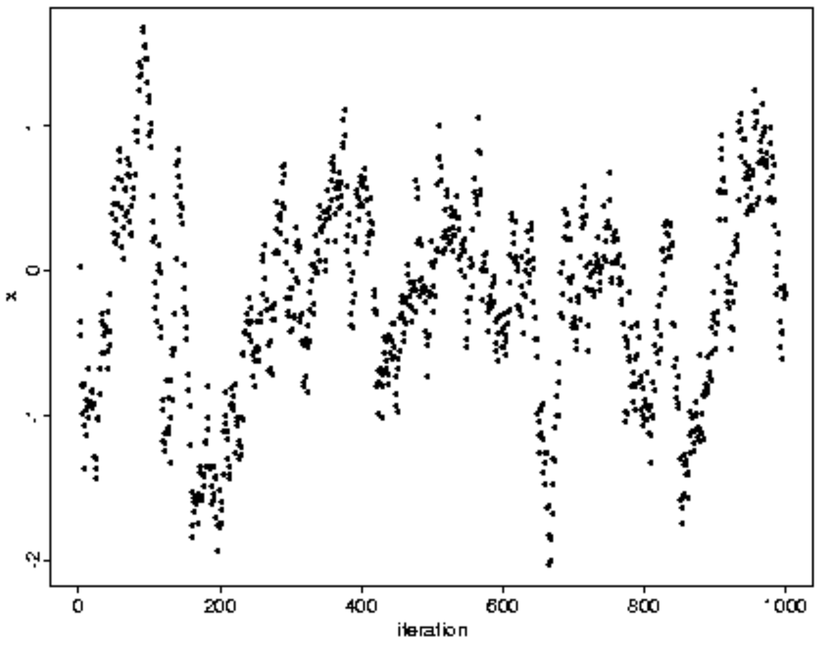
with $r = 0.98$ and

$$e_n \sim \mathcal{N}(0,1)$$

starting from $x_0 = 10$ (left)

and $x_0 = 0$ (right)



# MC error

Our goal is to estimate the expectation value of some physical observable $f$.

Using traditional MC sampling, the estimate is given by the average over $N$ samples

$$\bar{f} = \frac{1}{N}\sum_{i=1}^{N} f_i$$

If these samples are independent, then the variance on this estimator is

$$\sigma^2_{\bar{f}} = \frac{1}{N}\mathrm{Var}[f_i] = \frac{1}{N}\left(\langle f^2\rangle - \bar{f}^2\right)$$

and the error decreases as $1/\sqrt{N}$ as we generate more samples.

# MCMC error

However, in MCMC, the samples are not independent! What is the variance?

$$\sigma_{\bar{f}}^2 = \frac{\tau_f}{N} \mathrm{Var}[f_i]$$

where $\tau_f$ is the integrated autocorrelation time for the chain.

$N/\tau_f$ is the effective number of samples and $\tau_f$ is the number of steps that are needed before the chain “forgets” where it started.

# Autocorrelation function

Can estimate the autocorrelation function of the observable $f$ as

$$c_f(t) = \langle (f_i - \bar{f})(f_{i+t} - \bar{f}) \rangle = \frac{1}{N-t} \sum_{i=1}^{N-t} (f_i - \bar{f})(f_{i+t} - \bar{f})$$

Note the variance is a special case: $c_f(0) = \mathrm{Var}[f_i]$

We can also define the normalized autocorrelation (like a Pearson correlation coefficient)

$$\rho_f(t) = c_f(t)/c_f(0)$$

It's probably more appropriate to define $c_f(t)$ as "autocovariance," but in practice you have to figure out what the literature means from context

# Integrated autocorrelation time

The integrated autocorrelation time is then given by

$$\tau_f = \sum_{t=-\infty}^{\infty} \rho_f(t) = 1 + 2 \sum_{t=1}^{N} \rho_f(t)$$

Trying to approximate this directly is difficult. At longer lags, $\rho_f(t)$ starts to contain more noise than signal and summing all the way out to $N$ will result in a very noisy estimate of $\tau_f$. Instead we want to cut off the some at some $M \ll N$

$$\tau_f(M) = 1 + 2 \sum_{t=1}^{M} \rho_f(t)$$

One suggestion: using smallest $M$ where $M \geq 5\tau_f(M)$

# Estimating

Estimating integrated autocorrelation function: https://emcee.readthedocs.io/en/stable/tutorials/autocorr/