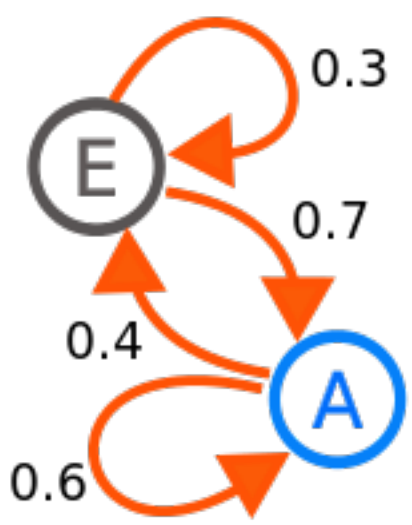
An Introduction to Python Programming using Markov Chain Modelling

Evan Sheridan



evan.sheridan@kcl.ac.uk

Schedule

- Session 1: introduction to Markov Chain modelling.
  (S2.28, 08/11/17, 15.00 - 16.00)

- Session 2: introduction to Python.
  (K4.32, 22/11/17, 15.00 - 17.00)

- Session 3: Markov Chain modelling in Python.
  (K4.32, 06/12/17, 15.00 - 16.00)

Lecture Structure

Part 1: Recap of Markov Chains

Part 2: Markov Chains and Monte Carlo in Action

Part 1

Recap of Markov Chains

Recap of Markov Chains

Some of the questions that we want to address:

- What is a Markov Chain?
- What is Monte Carlo?
- What is Markov Chain Monte Carlo (MCMC)?
- Why is MCMC important?

What is a Markov Chain? (1/)

Consider the following game:

(1) Bet £1 on outcome of coin flips

(2) Start with £10 and after 5 coin flips you have £13

(3) Next flip: you will have £14 or £12

The amount of money you have after the 6th coin flip
is only related to the money you had on the 5th coin flip.

This is a Markovian Process

What is a Markov Chain?

Formally, this means:

Conditional probability of being in x_n given only x_{n-1}

$$P_{1|n-1}(x_n, t_n | x_{n-1} t_{n-1}; \ldots ; x_1, t_1) = P_{1|1}(x_n, t_n | x_{n-1}, t_{n-1})$$

The defining Markov property: $P_{1|n-1} = P_{1|1}$

A Simple Markov Process

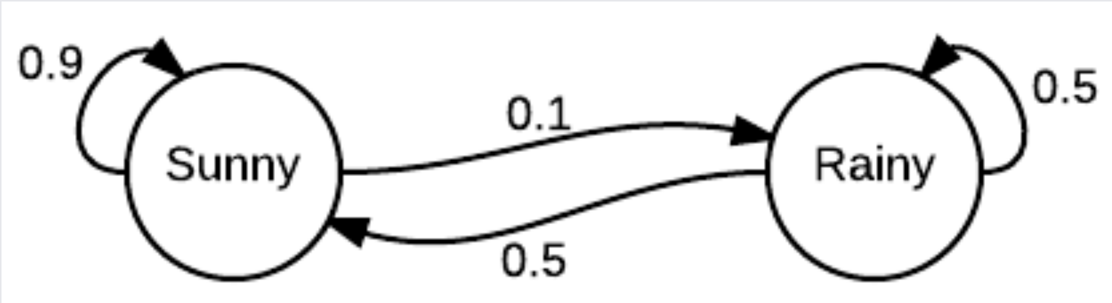
$$P_2(x_2, t_2; x_1, t_1) = P_{1|1}(x_2, t_2 | x_1, t_1) P_1(x_1, t_1)$$

The Markov Process is fully specified by P_1 and $P_{1|1}$

What is a Markov Chain?

- The corresponding sequence of states $\{x_t\}$is a Markov Chain
- A Markov Chain is fully determined by a *transition matrix*.

Example: the weather



$$\mathbf{P} = \begin{pmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{pmatrix}$$

$$\begin{pmatrix} S \to S & S \to R \\ R \to S & R \to R \end{pmatrix}$$

Transition matrix

- Conditional probabilities determine the transition matrix:

$$P_{ij} = P(S_i \to S_j) = P(X_{t_n} = S_j | X_{t_{n-1}} = S_i)$$

What is a Markov Chain?

- Given a system of N states, at t=0 the probabilities are $\mathbf{p}(0) = (p_1(0), \ldots, p_N(0))^T$

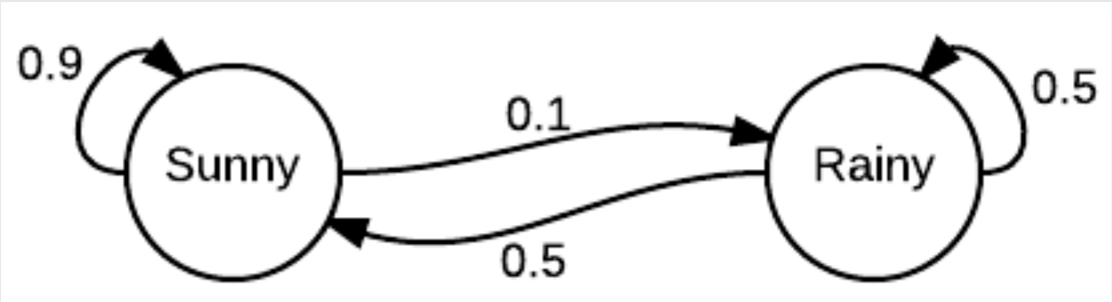
- Using the transition matrix $\mathbf{P}$ the state at time t is given by:

$$\mathbf{p}(t) = \mathbf{p}(0)\mathbf{P}^t$$

- It is interesting to consider the long time limit of this chain, i.e the stationary distribution π:

$$\mathbf{p}(t) = \mathbf{p}(t-1) \Rightarrow \quad \pi = \pi Q$$

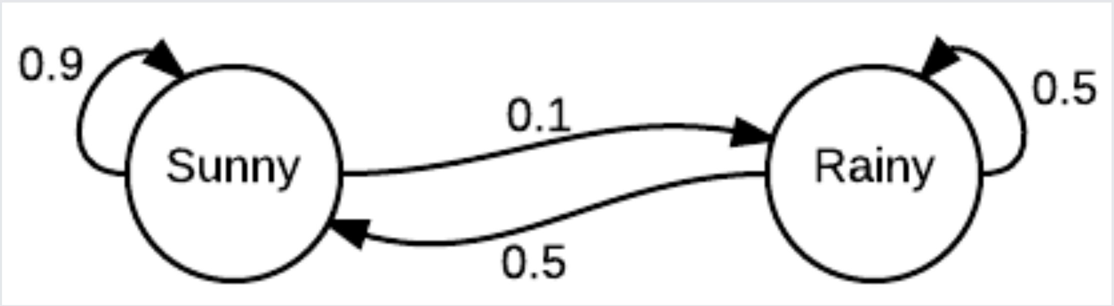
2 state system



- For MCMC we need $\mathbf{P}$ to be ergodic which requires:
  1. **Irreducible:** for every state there positive probability of moving to any other state.
  2. **Aperiodic:** no cyclical traps in the chain

Example: stationary distribution for the weather

- Suppose that the weather can be modelled with a Markov Chain, is it possible to find the stationary distribution of that chain?



$$\mathbf{P} = \begin{pmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{pmatrix}$$

$$\mathbf{qP} = \mathbf{q}$$

$$\downarrow$$

$$\mathbf{q}(\mathbf{P} - \mathbf{I}) = \mathbf{0}$$

$$\downarrow$$

$$\Rightarrow \mathbf{q} \times \begin{pmatrix} -0.1 & 0.1 \\ 0.5 & -0.5 \end{pmatrix} = \mathbf{0}$$

$$-0.1q_1 + 0.5q_2 = 0$$

$$\downarrow$$

$$q_1 + q_2 = 1$$

$$\downarrow$$

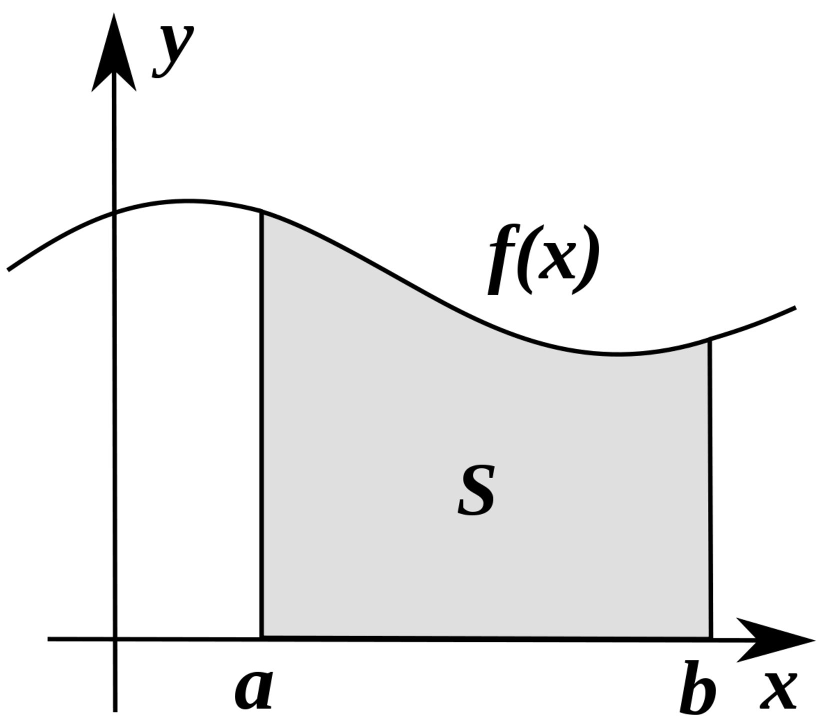
$$(q_1, q_2) = (0.833, 0.167)$$

- No matter the initial state that we are in, the chain eventually reaches a stationary distribution where it is always 83% likely to be sunny.

What is Monte Carlo?

- Consider calculating the integral:

$$I = \int_a^b f(x)dx$$



- Instead, what if we randomly sample over the integration range and approximate the integral? I.e:

$$\tilde{I} = (b-a)\langle f(x) \rangle \qquad \langle f(x) \rangle = \frac{1}{N}\sum_i^N f(x_i)$$

x_i uniformly sampled

- Using random numbers to calculate integrals: Monte Carlo.

Importance Sampling

- If $f(x)$ is negligible in different places over the integration interval, then it is a waste of time to sample these values.
- A way around this is to importantly (not uniformly) sample $f(x_i)$ on $[a, b]$ according to $p(x)$
- To do this, we need to weight $f(x)$ by $p(x)$

<u>Importance Sampling Procedure</u>

$$I = \int_a^b dx p(x) \frac{f(x)}{p(x)} \quad \xrightarrow[u(b)=1 \quad u(a)=0]{u'(x)=p(x)} \quad I_E = \int_0^1 du \frac{f[x(u)]}{p[x(u)]}$$

$$\approx \frac{1}{L} \sum_{i=1}^{L} \frac{f[x(u_i)]}{p[x(u_i)]}$$

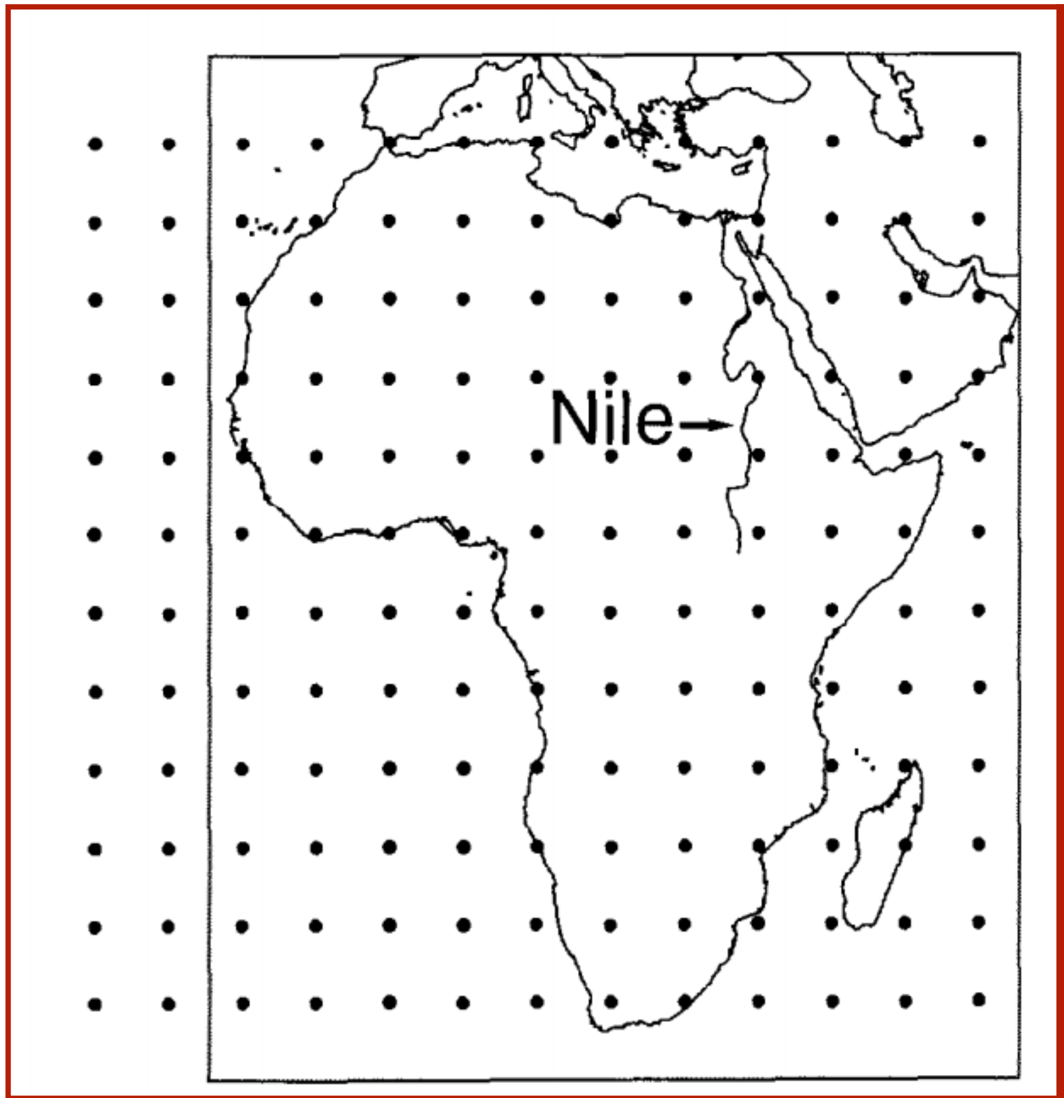
- ***To minimise the variance we must pick $p(x)$ proportional to $f(x)$***

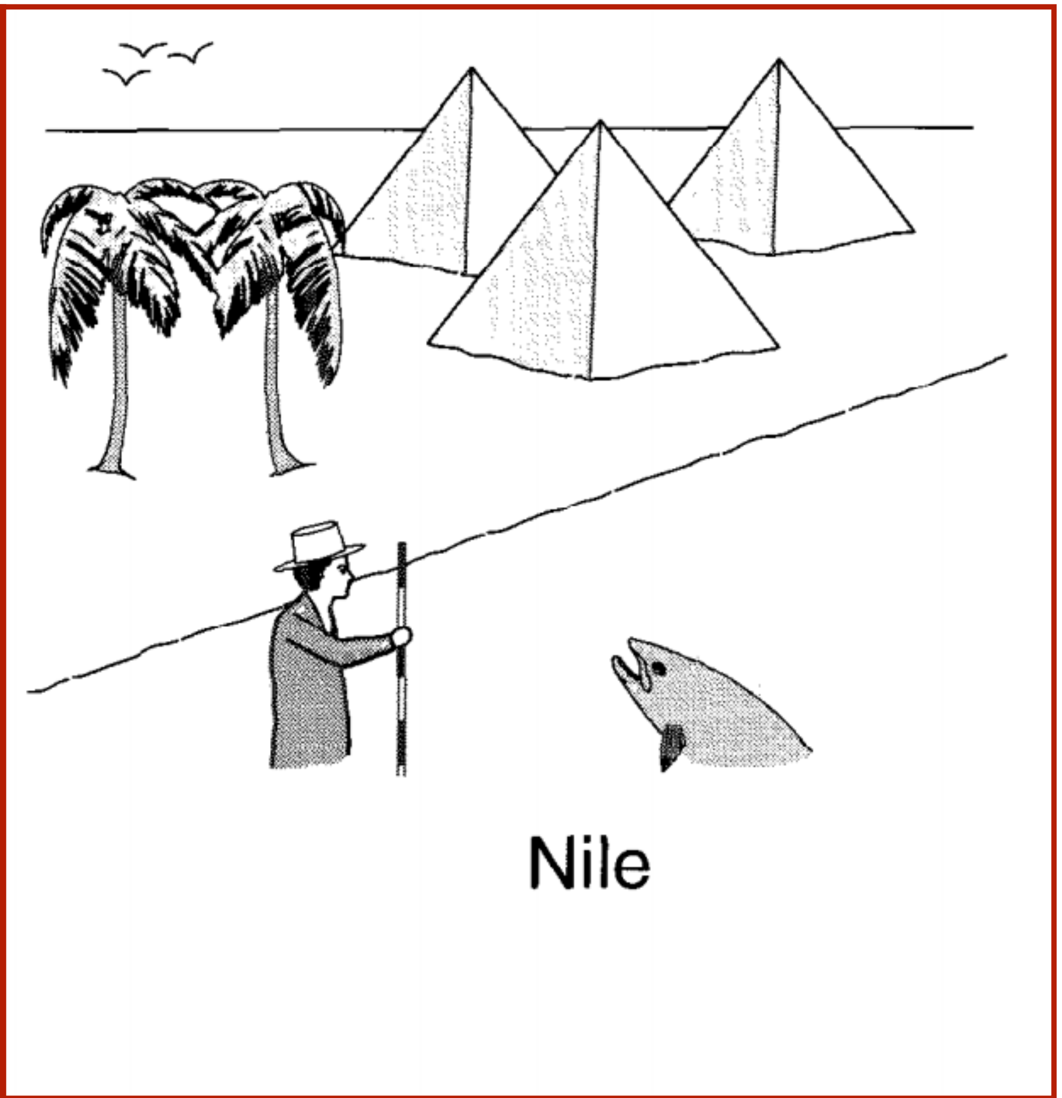
$$p(x) \propto f(x) \quad \longleftarrow \quad \sigma^2$$

variance in I_E

Metropolis Algorithm for Monte Carlo

Question: what is the average depth of the river Nile?



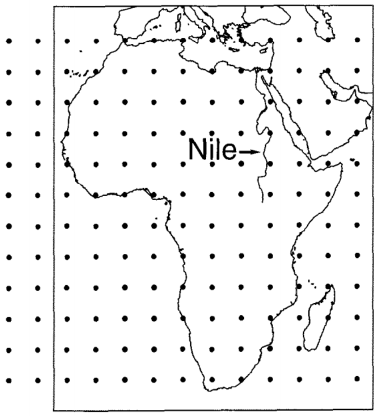


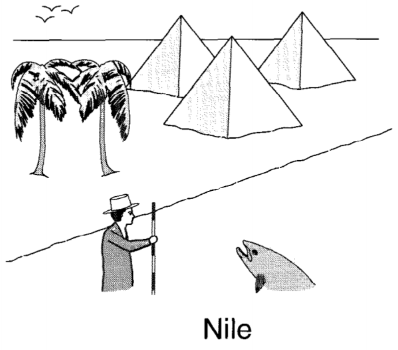
Naive Integration method	*Metropolis Monte Carlo*

Metropolis Algorithm for Monte Carlo

- This amounts to calculating

$$\langle D \rangle = \int_{x_{min}}^{x_{max}} \int_{y_{min}}^{y_{max}} p(x,y) D(x,y) dx dy$$





$$\langle D \rangle = \frac{1}{N^2} \sum_{i,j} D(x_i, y_j)$$

- x_i, y_j from uniform grid.
- There are many points which are not near the Nile.
- End up sampling useless points.

Naive Integration method

$$\langle D \rangle = \frac{1}{M} \sum_{ij} D(x_i, y_j)$$

- $x_i \; y_j$ from $p(x,y)$.
- Chain of movements:

$$x_i \to x_i + \delta$$

$$y_j \to y_j + \tilde{\delta}$$

is accepted with probability

- Generates random walk through the Nile.

Metropolis Monte Carlo

What is Markov Chain Monte Carlo?

- Suppose we now want to calculate $\langle A \rangle = \int dx A(x) p(x)$ where $p(x)$ is a (possibly unnormalised) probability distribution.

- A Markov Chain of values starting at x_i can be generated that are $\propto p(x)$.

- Usually, the famous Metropolis algorithm governs the criteria under which x_i are generated.

- Ultimately, this allows the expectation value to be written as:

$$\langle A \rangle = \sum_{i=1}^{N} A(x_i)$$

- We will see how this works in practise for a famous statistical physics problem.

Why is Markov Chain Monte Carlo important?

- Quite often, a system cannot be solved analytically because of complexity and so computational methods need to developed to solve these systems.
- The computational implementation is relatively straightforward as long as you can trust the generation of random numbers.
- It can be used to model so many systems in finance, biology, physics, maths, …

Part 2

Markov Chains and Monte Carlo in Action

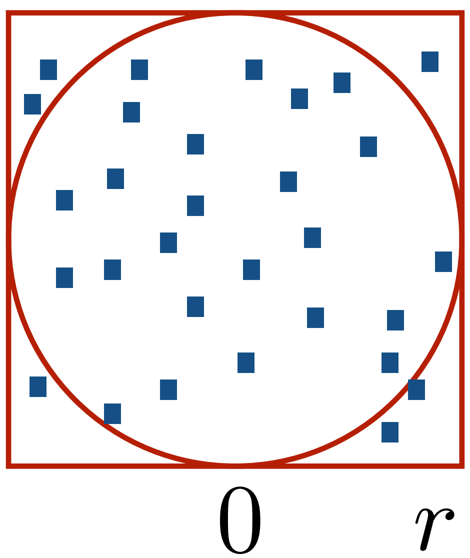
Markov Chains in Action

1. Calculating π using Monte Carlo
2. The Ising Model

Each of the above examples will be implemented in the practical tutorial sessions.

Calculating Pi with Monte Carlo

- *Goal*: using random numbers we can make an estimate of π



Algorithm

- Choose pairs of points randomly: $-r < x < r\ ,\ -r < y < r$
- $N_s \to N_s + 1$
- **IF** $x^2 + y^2 < r^2$ **THEN** $N_c \to N_c + 1$
- As more points are drawn: $A_c/A_s \approx N_c/N_s$

$$\frac{A_c}{A_s} = \frac{\pi r^2}{(2r)^2} = \frac{\pi}{4} \quad \Longrightarrow \quad \pi = 4 \times \left(\frac{N_c}{N_s}\right)$$

- The above algorithm is of the type acceptance/rejection.
- Amounts to calculating the integral:

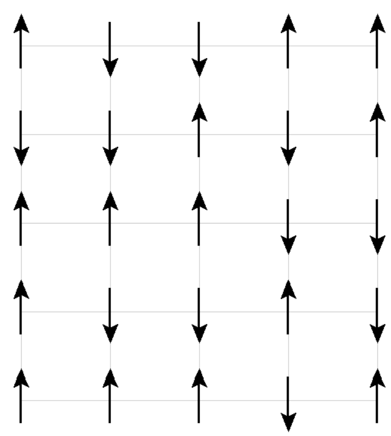
$$\int_{x,y\in\square} dxdy\,\pi(x,y)f(x,y)$$

$$\pi(x,y) = 1$$

$$f(x,y) = \begin{cases} 1 \text{ if } x^2 + y^2 < r^2 \\ 0 \text{ otherwise} \end{cases}$$

The Ising Model

- The Ising Model is a simple model of interacting magnetic spins on a lattice



Simple Ising Model Setup

- Spins are constrained to have values $s_i = \pm 1$

Ising Model Hamiltonian

$$\mathcal{H} = -J \sum_{\langle ij \rangle} s_i s_j$$

sum over nearest neighbours

- At a critical temperature T_c there is a phase transition.

- This phase transition can be predicted theoretically on a infinite lattice.

- ***Goal: using Markov Chain Monte Carlo is there evidence of a phase transition (even though it will have to be a finite system)?***

- ***Caveat:*** all of what is discussed here is quite general and relevant to other models

The Ising Model: setup

- How can we calculate thermodynamic observables of the Ising Model as a function of temperature?

- For example, to calculate the ground state energy

$$\langle E \rangle = \frac{\int dE e^{-\beta E} E}{\mathcal{Z}} \qquad \mathcal{Z} = \int dE e^{-\beta E}$$

- The state space for a lattice of N sites contains 2^N Ising configurations.

- This state space is far too large to apply a simple accept/reject Monte Carlo method

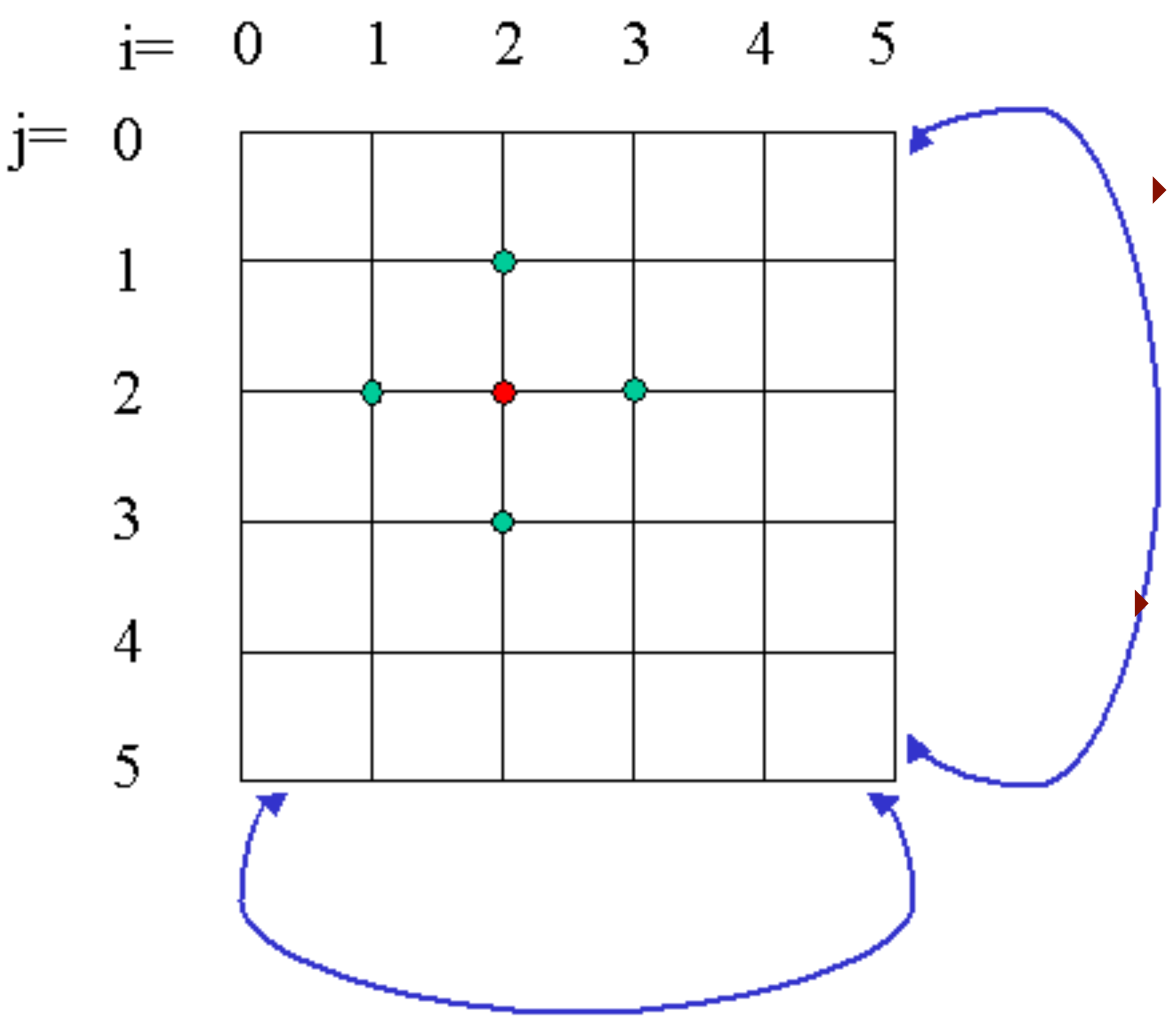
- To sample from relevant configurations a MCMC Metropolis algorithm for the Ising Model will have to be used.

The Ising Model: setup (1/N)

Metropolis importance sampling Monte Carlo scheme

(1) Choose an initial state
(2) Choose a site i
(3) Calculate the energy change ΔE which results if the spin at site i is overturned
(4) Generate a random number r such that $0 < r < 1$
(5) If $r < \exp(-\Delta E/k_B T)$, flip the spin
(6) Go to the next site and go to (3)

Boundary conditions



- Recall that the Ising Hamiltonian has a sum over nearest neighbours.
- How is it possible to deal with points on the boundaries, that seem to only have 2 or 3 nearest neighbours?

- The strategy is to employ ***periodic boundary conditions*** on the lattice as illustrated above.
- For example, we can impose that: $s_{N+1} = s_1$

References

- *A guide to Monte Carlo Simulations - David P. Landau & Kurt Binder*
- *Understanding Molecular Simulation - Daan Frenkel & Berend Smit*
- *Dynamical Analysis of Complex Systems Lecture Notes - Alessia Annibale*