5. Monte-Carlo methods I

Background

- The goal of Monte-Carlo methods is the sampling of a process in order to determine some statistical properties
- For instance, we toss a coin 4 times. What is the probability to obtain 3 tail and 1 head?
- Mathematics gives us the solution:

$$P(3 \text{ head}) = \left(\begin{array}{c} 4 \\ 3 \end{array} \right) \left(\frac{1}{2}\right)^3 \left(1 - \frac{1}{2}\right)^1 = \frac{1}{4}$$

- But we could also do a simulation

A Monte-Carlo computer simulation

```
from random import randint

success=0

attempts=10000
for i in range(attempts):
    if randint(0,1)+randint(0,1)+randint(0,1)+randint(0,1)==3:
        success+=1

print "Number of attempts=", attempts
print "Number of success=", success
```

We get for instance:

```
Number of attempts= 10000
Number of success= 2559
```

More difficult problems

- For the coin tossing problem, no need for a simulation

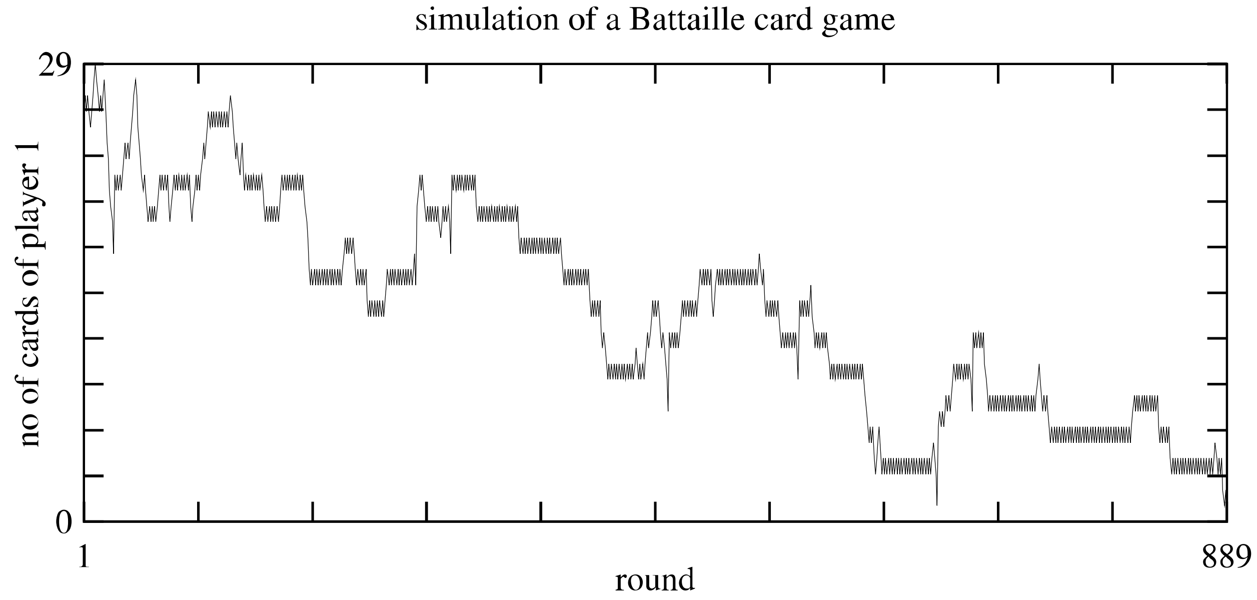
More difficult problems

- For the coin tossing problem, no need for a simulation
- But we can think of other problems for which probability theory could hardly be applied

More difficult problems

- For the coin tossing problem, no need for a simulation
- But we can think of other problems for which probability theory could hardly be applied
- For instance: what is the average duration of the card game called “war” (or battle)?

The *war* card game with 52 cards



Historical note

- The method was name in the 1940s by **John von Neumann**, **Stanislaw Ulam** and **Nicholas Metropolis** after the name of the *Monte-Carlo casino*, where Ulam's uncle used to gamble ...and loose his money
- The motivation was to find out the probability that a Canfield solitaire will finish sucessfully.
- Ulam found it easier to play many Canfield solitaires and estimate the number of successes, rather than trying to apply combinatorics and probability theory.
- Then the Monte-Carlo methods was successfully applied to the *Manhattan project* (nuclear weapon) in the Los Alamos National Laboratory.

End of module

Monte-Carlo Methods I

Coming next

Monte-Carlo Methods II

6. Monte-Carlo Methods II

Markov-Chain Monte-Carlo (MCMC)

- We consider a stochastic process whose goal is to explore the state space of a system of interest.

Markov-Chain Monte-Carlo (MCMC)

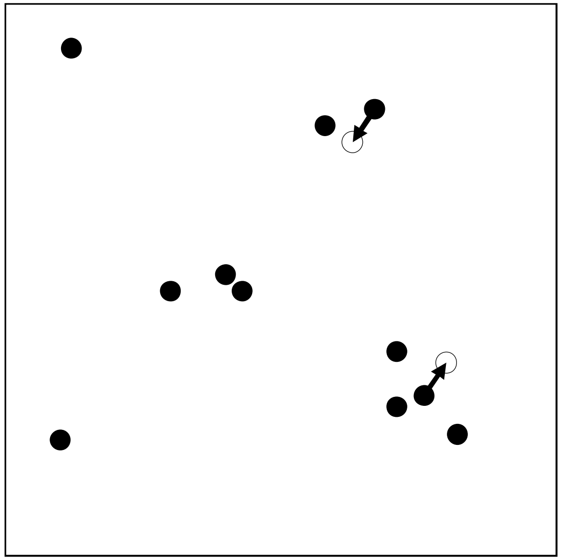
- We consider a stochastic process whose goal is to explore the state space of a system of interest.
- Let x be a point in this state space. Let us asssume that this point moves across the space by jumping randomly to another point x'.

Markov-Chain Monte-Carlo (MCMC)

- We consider a stochastic process whose goal is to explore the state space of a system of interest.
- Let x be a point in this state space. Let us asssume that this point moves across the space by jumping randomly to another point x'.
- The jump from location x to location x' takes place with probability $W_{x \to x'}$. This advanced the system time from t to $t + 1$ (Markov chain)

Markov-Chain Monte-Carlo (MCMC)

- We want this process to sample a prescribed probability $\rho(t, x)$. This stochastic process should be at point x at time t with a probability $\rho(t, x)$.
- How do we choose $W_{x \to x'}$?



$$\rho \propto \exp(-E(x)/k_B T)$$

Sampling the diffusion equation in 1D

The probability that our random exportation is at location x at time t is

$$p(t+1, x) = \sum_{x'} p(t, x') W_{x' \to x}$$

Sampling the diffusion equation in 1D

The probability that our random exportation is at location x at time t is

$$p(t+1,x) = \sum_{x'} p(t,x')W_{x'\to x}$$

- Let us consider a 1D discrete space: $x \in \mathbf{Z}$.

Sampling the diffusion equation in 1D

The probability that our random exportation is at location x at time t is

$$p(t+1,x) = \sum_{x'} p(t,x') W_{x' \to x}$$

- Let us consider a 1D discrete space: $x \in \mathbf{Z}$.
- where one can move to the right with probability W_+, to the left with probability W_- and stay still with probability W_0.

Sampling the diffusion equation in 1D

The probability that our random exportation is at location x at time t is

$$p(t+1,x) = \sum_{x'} p(t,x') W_{x' \to x}$$

- Let us consider a 1D discrete space: $x \in \mathbf{Z}$.
- where one can move to the right with probability W_+, to the left with probability W_- and stay still with probability W_0.
- The equation for $p(t,x)$ simplifies to

$$p(t+1,x) = p(t,x-1)W_+ + p(t,x)W_0 + p(t,x+1)W_-$$

Diffusion equation (1D)

- The diffusion equation is $\partial_t \rho = D \partial_x^2 \rho$

Diffusion equation (1D)

- The diffusion equation is $\partial_t \rho = D\partial_x^2 \rho$
- Which can be discretized as (see week 3)

$$\rho(t+\Delta t, x) = \rho(t,x) + \frac{\Delta t D}{\Delta x^2}\left(\rho(t,x-1) - 2\rho(t,x) + \rho(t,x+1)\right)$$

Diffusion equation (1D)

- The diffusion equation is $\partial_t \rho = D \partial_x^2 \rho$
- Which can be discretized as (see week 3)

$$\rho(t + \Delta t, x) = \rho(t, x) + \frac{\Delta t D}{\Delta x^2} \left(\rho(t, x-1) - 2\rho(t, x) + \rho(t, x+1)\right)$$

- to be compared with

$$p(t+1, x) = p(t, x-1) W_+ + p(t, x) W_0 + p(t, x+1) W_-$$

Diffusion equation (1D)

- The diffusion equation is $\partial_t \rho = D\partial_x^2 \rho$
- Which can be discretized as (see week 3)

$$\rho(t+\Delta t, x) = \rho(t,x) + \frac{\Delta t D}{\Delta x^2}\left(\rho(t,x-1) - 2\rho(t,x) + \rho(t,x+1)\right)$$

- to be compared with

$$p(t+1,x) = p(t,x-1)W_+ + p(t,x)W_0 + p(t,x+1)W_-$$

- In order to have $p = \rho$, one need $W_+ = W_- = \Delta t D/(\Delta x)^2$ and $W_0 = 1 - 2\Delta t D/(\Delta x)^2 = 1 - W_+ - W_-$, and thus $\Delta t D/(\Delta x)^2 \leq 1/2$

Monte-Carlo simulation of Diffusion

- Therefore a random walk is a way to sample a density ρ that obeys the diffusion equation.

Monte-Carlo simulation of Diffusion

- Therefore a random walk is a way to sample a density ρ that obeys the diffusion equation.
- With a random walk, it is easy to add obstacles, or aggregation processes, hard to include in the differential equation.

More general case: Master equation

The probability to find the random exploration at location x at time t is $p(t,x)$ given by

$$\begin{aligned} p(t+1,x) &= \sum_{x'} p(t,x')W_{x'\to x} \\ &= \sum_{x'\neq x} p(t,x')W_{x'\to x} + p(t,x)W_{x\to x} \\ &= \sum_{x'\neq x} p(t,x')W_{x'\to x} + p(t,x)(1-\sum_{x'\neq x} W_{x\to x'}) \\ &= p(t,x) + \sum_{x'\neq x} [p(t,x')W_{x'\to x} - p(t,x)W_{x\to x'}] \end{aligned} \qquad (1)$$

Detailed balance

In a steady state, the condition $p(x) = \rho(x)$ requires that

$$\sum_{x' \neq x} \left[\rho(x') W_{x' \to x} - \rho(x) W_{x \to x'}\right] = 0$$

We can then choose $W_{x \to x'}$ according to the **detailed balance** condition

$$\rho(x') W_{x' \to x} - \rho(x) W_{x \to x'} = 0$$

Metropolis Rule

Let us consider a physical system at equilibrium whose probability to be in state x is given by the Maxwell-Boltzmann distribution

$$\rho(x) = \Gamma \exp(-E(x)/kT)$$

We can sample this distribution with a stochastic process by choosing $W_{x\to x'}$ according to the **Metropolis rule**:

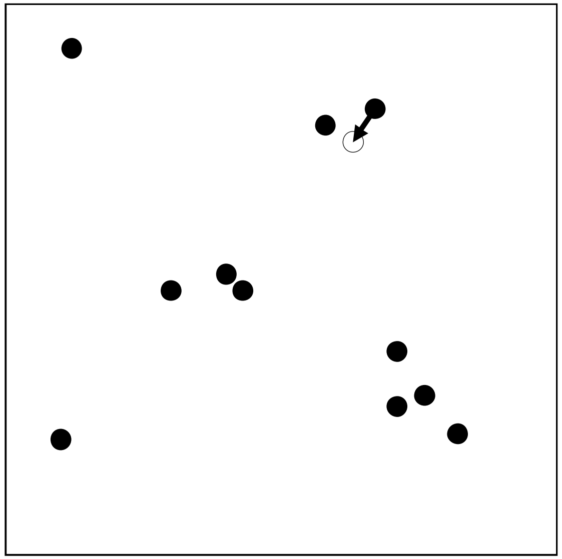
$$W_{x\to x'} = \left\{ \begin{array}{cl} 1 & \text{si } E' < E \\ \exp[-(E'-E)/kT] & \text{si } E' > E \end{array} \right.$$

The Metropolis Rule in practice

- In a gas, one selects one particle at random.
- One moves it by an amount Δx.
- One computes the energy E' of the gas with this new position.
- One accepts this change if

$$\mathrm{rand}(0,1) < \min(1, \exp[-(E' - E)/kT])$$

- By sampling ρ with $W_{x \to x'}$, one can compute average physical properties, such as for instance the pressure in the gas.



The Metropolis obeys the detailed balance

Let us assume that $E' > E$. Detailed balance is obeyed because

$$\begin{aligned}\rho(x)W_{x\to x'} &= \Gamma\exp(-E/kT)\exp[-(E'-E)/kT] \\ &= \Gamma\exp(-E'/kT) \\ &= \rho(x')\times 1 \\ &= \rho(x')W_{x'\to x}\end{aligned}$$

And similarly if $E' \leq E$

Glauber Rule

This is an alternative to the Metropolis rule. $W_{x\to x'}$ is given by

$$W_{x\to x'} = \frac{\rho(x')}{\rho(x) + \rho(x')}$$

which also clearly obeys detailed balance
With $\rho = \Gamma \exp(-E(x)/kT)$, one obtains

$$W_{x\to x'} = \frac{\exp(-E'/kT))}{\exp(-E/kT) + \exp(-E'/kT)}$$

End of module

Monte-Carlo Methods II

Coming next

Monte-Carlo Methods III

7. Monte-Carlo Methods III

Kinetic / Dynamic Monte-Carlo

Let us consider the chemical equations

$$A \xrightarrow{k_1} B \qquad B \xrightarrow{k_2} A$$

They can be written as an ordinary equation

$$\frac{d}{dt}\begin{pmatrix} A \\ B \end{pmatrix} = \begin{pmatrix} -k_1 & k_2 \\ k_1 & -k_2 \end{pmatrix}\begin{pmatrix} A \\ B \end{pmatrix}$$

Analytical solution

$$A(t) = \frac{k_2}{k_1 + k_2}(A_0 + B_0) + \frac{A_0 k_1 - B_0 k_2}{k_1 + k_2} e^{-(k_1+k_2)t}$$

$$B(t) = \frac{k_1}{k_1 + k_2}(A_0 + B_0) - \frac{A_0 k_1 - B_0 k_2}{k_1 + k_2} e^{-(k_1+k_2)t}$$

where A_0 and B_0 are the initial concentration of A and B.
When $t \to \infty$,

$$A \to A_\infty = \frac{k_2}{k_1 + k_2}(A_0 + B_0) \qquad B \to B_\infty = \frac{k_1}{k_1 + k_2}(A_0 + B_0)$$

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1 \Delta t$ et $k_2 \Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1\Delta t$ et $k_2\Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) = \text{const}$ of them. (In practice one chooses a A particle $\text{rand}(0, 1) < A/(A + B)$, and a B particle otherwise.

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1\Delta t$ et $k_2\Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) = \mathrm{const}$ of them. (In practice one chooses a A particle $\mathrm{rand}(0,1) < A/(A+B)$, and a B particle otherwise.

3a If a A particle was chosen, it is transformed into a B particle provided $\mathrm{rand}(0,1) < k_1\Delta t$. Then $A = A - 1$, $B = B + 1$.

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1 \Delta t$ et $k_2 \Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) = \text{const}$ of them. (In practice one chooses a A particle $\text{rand}(0,1) < A/(A+B)$, and a B particle otherwise.

3a If a A particle was chosen, it is transformed into a B particle provided $\text{rand}(0,1) < k_1 \Delta t$. Then $A = A - 1$, $B = B + 1$.

3b If a B particle was chosen, it is transformed into a A particle, provided $\text{rand}(0,1) < k_2 \Delta t$. Then $A = A + 1$, $B = B - 1$.

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1\Delta t$ et $k_2\Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) =$ const of them. (In practice one chooses a A particle rand$(0,1) < A/(A+B)$, and a B particle otherwise.

3a If a A particle was chosen, it is transformed into a B particle provided rand$(0,1) < k_1\Delta t$. Then $A = A - 1$, $B = B + 1$.

3b If a B particle was chosen, it is transformed into a A particle, provided rand$(0,1) < k_2\Delta t$. Then $A = A + 1$, $B = B - 1$.

4 (2) and (3) are repeated N times and the physical time t is incremented by Δt: $t = t + \Delta t$

Monte-Carlo Simulation

1 One defines a time step Δt, small enough so that $k_1 \Delta t$ et $k_2 \Delta t$ are smaller than 1. They are the **probabilities** that, during Δt, one A particle get transformed into one B particle, or conversely.

2 One chooses randomly a particle among the $N = A(t) + B(t) = \mathrm{const}$ of them. (In practice one chooses a A particle $\mathrm{rand}(0,1) < A/(A+B)$, and a B particle otherwise.

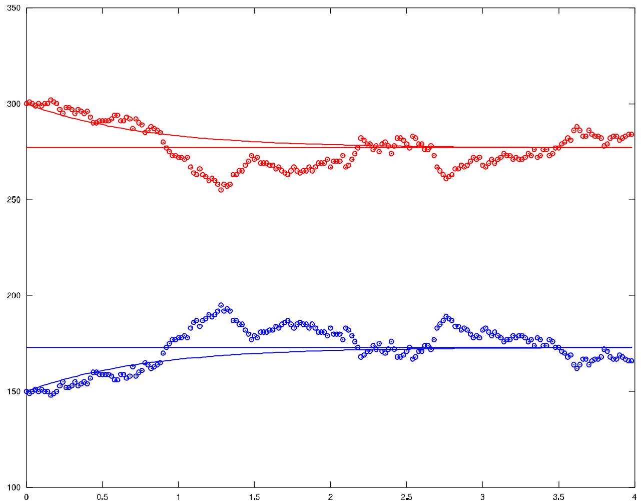
3a If a A particle was chosen, it is transformed into a B particle provided $\mathrm{rand}(0,1) < k_1 \Delta t$. Then $A = A - 1$, $B = B + 1$.

3b If a B particle was chosen, it is transformed into a A particle, provided $\mathrm{rand}(0,1) < k_2 \Delta t$. Then $A = A + 1$, $B = B - 1$.

4 (2) and (3) are repeated N times and the physical time t is incremented by Δt: $t = t + \Delta t$

5 One repeats (2)-(4) until $t = t_{max}$

Results



$\Delta t = 0.02$ and $k_1 = 0.5$, $k_2 = 0.8$.

The Monte-Carlo simulation fluctuate around analytic solution.

We should average over several runs

Gillespie's algorithm

- Let r_i be the rate at which the possible events occur in the system. $i = 1, \ldots n$.

Gillespie's algorithm

- Let r_i be the rate at which the possible events occur in the system. $i = 1, \ldots n$.
- For instance, $r_i = kAB$ for a reaction $A + B \to C$

Gillespie's algorithm

- Let r_i be the rate at which the possible events occur in the system. $i = 1, \ldots n$.
- For instance, $r_i = kAB$ for a reaction $A + B \rightarrow C$
- Let $R_i = \sum_{j=1}^{i} r_i$ be the cumulative rates.

Gillespie's algorithm

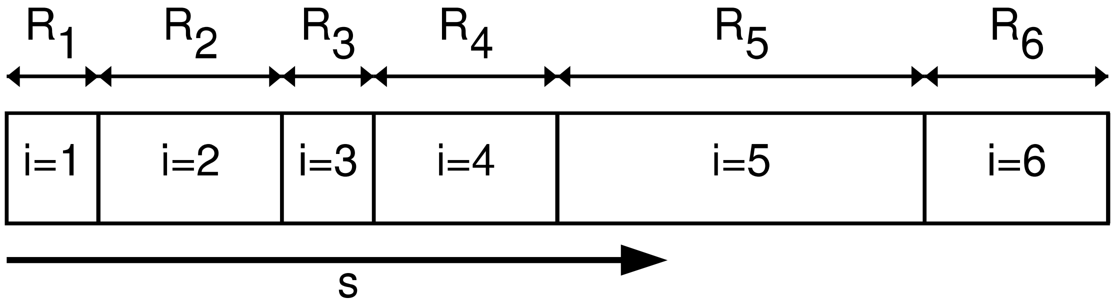
- Let r_i be the rate at which the possible events occur in the system. $i = 1, \ldots n$.
- For instance, $r_i = kAB$ for a reaction $A + B \to C$
- Let $R_i = \sum_{j=1}^{i} r_i$ be the cumulative rates.
- Choose one event k at random by picking a random number $s = \mathrm{rand}(0, 1)$: One chooses the k which verifies $R_{k-1} < sR_n < R_{k+1}$ (probability proportional to rate).

Gillespie's algorithm

- Let r_i be the rate at which the possible events occur in the system. $i = 1, \ldots n$.
- For instance, $r_i = kAB$ for a reaction $A + B \to C$
- Let $R_i = \sum_{j=1}^{i} r_i$ be the cumulative rates.
- Choose one event k at random by picking a random number $s = \mathrm{rand}(0,1)$: One chooses the k which verifies $R_{k-1} < sR_n < R_{k+1}$ (probability proportional to rate).

Gillespie's algorithm

- Let r_i be the rate at which the possible events occur in the system. $i = 1, \dots n$.
- For instance, $r_i = kAB$ for a reaction $A + B \rightarrow C$
- Let $R_i = \sum_{j=1}^{i} r_i$ be the cumulative rates.
- Choose one event k at random by picking a random number $s = \mathrm{rand}(0, 1)$: One chooses the k which verifies $R_{k-1} < sR_n < R_{k+1}$ (probability proportional to rate).



Gillespie's algorithm

- Execute the selected event (for instance a molecule A and molecule B disappear to produce a new molecule C).

Gillespie's algorithm

- Execute the selected event (for instance a molecule A and molecule B disappear to produce a new molecule C).
- Advance time as $\Delta t = \ln(1/s')R_n^{-1}$, with $s' = \mathrm{rand}(0,1)$.

Gillespie's algorithm

- Execute the selected event (for instance a molecule A and molecule B disappear to produce a new molecule C).
- Advance time as $\Delta t = \ln(1/s')R_n^{-1}$, with $s' = \mathrm{rand}(0,1)$.
- Note that here Δt is calculated according to a decreasing exponential distribution. It gives the average time of occurrence of the next event.

Gillespie's algorithm

- Execute the selected event (for instance a molecule A and molecule B disappear to produce a new molecule C).
- Advance time as $\Delta t = \ln(1/s')R_n^{-1}$, with $s' = \text{rand}(0,1)$.
- Note that here Δt is calculated according to a decreasing exponential distribution. It gives the average time of occurrence of the next event.
- Only one event takes place during the time interval Δt.

End of module

Monte-Carlo method III

End of Week 1

Thank you for your attention!