Markov Chain MC

Marcin Chrząszcz
mchrzasz@cern.ch

Experimental Methods in Particle Physics,
7 October, 2018

Trivial example

⇛ Lets start with a TRIVIAL example: we want to calculate $S = A + B$. We can rewrite it in:

$$S = p\frac{A}{p} + (1-p)\frac{B}{1-p}$$

and one can interpret the sum as expected value of:

$$W = \begin{cases} \frac{A}{p} & \text{with propability } p \\ \frac{A}{1-p} & \text{with propability } 1-p \end{cases}$$

⇛ The algorithm:

- We generate a random variable W and calculate:

$$\hat{S} = \frac{1}{N}\sum_{i=1}^{N} W_i$$

⇛ The $\hat{S}$ is an unbias estimator of S.

Trivial example2

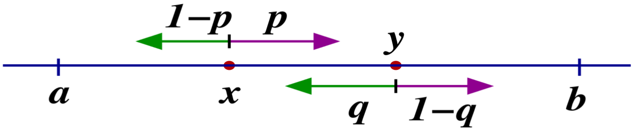
- Lets say we have a linear equation system:

$$\begin{aligned} X &= pY + (1-p)A \\ Y &= qX + (1-q)B \end{aligned}$$

- We know A, B, p, q; X and Y are meant to be determined.
- Algorithm:
  1. We choose first element of the first equation with probability p and second with probability $1-p$.
  2. I we choose the second one, the outcome of this MCMC is $W = A$.
  3. If we choose the first we go to second equation and choose the first element with probability q and the second with $1-q$.
  4. We we choose the second one, the outcome of this MCMC is $W = B$.
  5. If we choose the first we go to the first equation back again.
  6. We repeat the procedure.
- We can estimate the solution of this system:

$$\hat{X} = \frac{1}{N}\sum_{i=1} W_i \qquad \sigma_{\hat{X}} = \frac{1}{\sqrt{N-1}}\sqrt{\frac{1}{N}\sum_{i=1}^{N} W_i^2 - \hat{X}^2}$$

Random walk



⇛ We are in the point x and we walk accordingly to the following rules:

- From point x we walk with probability p to point y or with $1 - p$ to a.
- From point y we walk with probability q to point x and with $1 - Q$ to b.
- The walks ends when you end up in a or b.
- You get a ”reward” A if you end up in point a and B if you end up in b.
- X is expected ”reward” when you start the walk from x, Y when you start from y.

⇛ The algorithm above is so-called random walk on the set $\{a, x, y, b\}$

⇛ The described walked can solve the linear equation system that we discussed above.

Markov Chain MC

- Consider a finite (or Countable set) possible states: S_1, S_2, ...
- The X_t is the state of the system in the time t
- We are looking at discrete time steps: $1, 2, 3, ...$
- The conditional probability is defined as:

$$P(X_t = S_j | X_{t-1} = S_{j-1}, ..., X_1 = S_1)$$

- The Markov chain is then if the probability depends only on previous step.

$$P(X_t = S_j | X_{t-1} = S_{j-1}, ..., X_1 = S_1) = P(X_t = S_j | X_{t-1} = S_{j-1})$$

- For this reason MCMC is also knows as drunk sailor walk.
- Very powerful method. Used to solve linear eq. systems, invert matrix, solve differential equations, etc.
- Also used in physics problems: Brown motions, diffusion, etc.

Linear equations system

$\Rrightarrow$ Lets start from a linear equation system:

$$\mathbf{A}\overrightarrow{x} = \overrightarrow{b}, \qquad \det \mathbf{A} \neq 0,$$

where $\mathbf{A} = (a_{ij}, i, j = 1, 2, ..., n$ -matrix, $\overrightarrow{b} = (b_1, b_2, ..., b_n)$-vector,
$\overrightarrow{x} = (x_1, x_2, ..., x_n)$ - vector of unknowns.

$\Rrightarrow$ The solution we mark as $\overrightarrow{x}^0 = (x_1^0, x_2^0, ..., x_n^0)$

$\Rrightarrow$ The above system can be transformed into the iterative representation:

$$\overrightarrow{x} = \overrightarrow{a} + \mathbf{H}\overrightarrow{x}$$

where $\mathbf{H}$ is a matrix, $\overrightarrow{a}$ is a vector.

$\Rrightarrow$ We assume that the matrix norm:

$$\|H\| = \max_{1 \leqslant i \leqslant n} \sum_{j=1}^{n} |h_{h_{ij}}| < 1$$

Linear equations system

⇛ Lets start from a linear equation system:

$$\mathbf{A}\overrightarrow{x} = \overrightarrow{b}, \qquad \det \mathbf{A} \neq 0,$$

where $\mathbf{A} = (a_{ij}, i, j = 1, 2, ..., n$ -matrix, $\overrightarrow{b} = (b_1, b_2, ..., b_n)$-vector, $\overrightarrow{x} = (x_1, x_2, ..., x_n)$ - vector of unknowns.

⇛ The solution we mark as $\overrightarrow{x}^0 = (x_1^0, x_2^0, ..., x_n^0)$

⇛ The above system can be transformed into the iterative representation:

$$\overrightarrow{x} = \overrightarrow{a} + \mathbf{H}\overrightarrow{x}$$

where $\mathbf{H}$ is a matrix, $\overrightarrow{a}$ is a vector.

⇛ We assume that the matrix norm:

$$\|H\| = \max_{1 \leqslant i \leqslant n} \sum_{j=1}^{n} |h_{h_{ij}}| < 1$$

⇛ We can always change transform every system to the iteration form: $\mathbf{A} = \mathbf{V} - \mathbf{W}$.

$$(\mathbf{V} - \mathbf{W})\overrightarrow{x} = \overrightarrow{b} \quad \mapsto \quad \overrightarrow{x} = \mathbf{V}^{-1}\overrightarrow{b} + \mathbf{V}^{-1}\mathbf{W}\overrightarrow{x}$$

Linear equations system

⇛ Now we further modify the equation system:

$$\overrightarrow{x} = \overrightarrow{a} + \mathbf{H}\overrightarrow{x} \Rightarrow (\mathbf{I} - \mathbf{H})\overrightarrow{x} = \overrightarrow{a}$$

where $\mathbf{I} = \delta_{ij}$ - unit matrix, δ_{ij} is the Kronecker delta.
⇛ What one can do is to represent the solution in terns of Neumann series:

$$\overrightarrow{x}^0 = (\mathbf{I} - \mathbf{H})^{-1}\overrightarrow{a} = \overrightarrow{a} + \mathbf{H}\overrightarrow{a} + \mathbf{H}^2\overrightarrow{a} + \mathbf{H}^3\overrightarrow{a} + ...$$

⇛ So for the i^{th} component we have:

$$x_i^0 = a_i + \sum_{j=1}^{n} h_{ij}a_j + \sum_{j_1=1}^{n}\sum_{j_2=1}^{n} h_{ij_1}h_{j_1j_2}a_{j_2} + ... + \sum_{j_1=1}^{n} ... \sum_{j_n=1}^{n} h_{ij_1}...h_{j_{n-1}j_n}a_{j_n}$$

⇛ We will construct a probabilistic interpretation using MCMC and then we show that the expected value is equal to the above formula.

Neumann-Ulam method

- To do so we add to our matrix an additional column of the matrix:

$$h_{i,0} = 1 - \sum_{j=1}^{n} h_{ij} > 0$$

- The system has states: $\{0, 1, 2..., n\}$
- State at t time is denoted as $i_t (i_t = 0, 1, 2, ..., n; t = 0, 1, ...)$
- We make a random walk accordingly to to the following rules:
  - At the beginning of the walk ($t = 0$) we are at i_0.
  - In the t moment we are in the i_t position then in $t + 1$ time stamp we move to state i_{t+1} with the probability $h_{i_t i_{t+1}}$.
  - We stop walking if we are in state 0.
- The path $\gamma = (i_0, i_1, i_2, ..., i_k, 0)$ is called trajectory.
- For each trajectory we assign a number:

$$X(\gamma) = X(i_0, i_1, i_2, ..., i_k, 0) = \frac{a_{i_k}}{h_{i_k 0}}$$

Neumann-Ulam method

⇛ The $X(\gamma)$ variable is a random variable from: $\{a_1/h_{1,0}, a_2/h_{2,0}, ..., a_n/h_{n,0}\}$. The probability that $X(\gamma) = a_j/h_{j,0}$ is equal to the probability that the last non zero state of the γ trajectory is j.

⇛ The expected value of the $X(\gamma)$ trajectory if the trajectory begins from $i_0 = s$ is:

$$E\{X(\gamma)|i_0 = s\} = \sum_{k=0}^{\infty} \sum_{\{\gamma_k\}} X(\gamma)P(\gamma)$$

where γ_k is a trajectory of length k, which starts in $i_0 = s$ and $P(\gamma)$ is the probability of occurrence of this trajectory. ⇛ Yes you guest it lets do Taylor expansion:

$$E\{X(\gamma)|i_0 = s\} = \sum_{\gamma_0} X(\gamma)P(\gamma) + \sum_{\gamma_1} X(\gamma)P(\gamma) + ... + \sum_{\gamma_k} X(\gamma)P(\gamma)$$

⇛ Now let's examine the elements of the above series.

Neumann-Ulam method

$\{\gamma_0\}$: One trajectory: $\gamma_0 = (i_0 = s|0)$, $P(\gamma_0) = h_{s,0}$ and $X(\gamma_0) = a_s/h_{s,0}$. So:

$$\sum_{\gamma_0} X(\gamma)P(\gamma) = \frac{a_s}{h_{s,0}} h_{s,0} = a_s$$

$\{\gamma_1\}$: Trajectories: $\gamma_1 = (i_0 = s, i_1|0),\ i_1 \neq 0$, $P(\gamma_1) = P(s, i_1, 0) = h_{s,i_1} h_{i_1,0}$ and $X(\gamma_1) = a_{i_1}/h_{i_1,0}$. So:

$$\sum_{\gamma_1} X(\gamma)P(\gamma) = \sum_{i_1=1}^{n} \frac{a_{i_1}}{h_{i_1,0}} h_{s,i_1} h_{i_1,0} = \sum_{i=1}^{n} h_{s,i_1} a_{i_1}$$

$\{\gamma_2\}$: Trajectories: $\gamma_2 = (i_0 = s, i_1, i_2|0),\ i_1, i_2 \neq 0$,
$P(\gamma_2) = P(s, i_1, i_2, 0) = h_{s,i_1} h_{i_1,i_2} h_{i_1,0}$ and $X(\gamma_2) = a_{i_2}/h_{i_2,0}$. So:

$$\sum_{\gamma_2} X(\gamma)P(\gamma) = \sum_{i_1=1}^{n}\sum_{i_2=1}^{n} \frac{a_{i_2}}{h_{i_2,0}} h_{s,i_1} h_{i_1,i_2} h_{i_2,0} = \sum_{i_1=1}^{n}\sum_{i_2=1}^{n} h_{s,i_1} h_{i_1,i_2} a_{i_2}$$

⇛ etc...

Neumann-Ulam method

⇛ After summing up:

$$E\{X(\gamma)|i_0 = s\} = a_s + \sum_{i_1=1}^{n} h_{s,i_1} a_{i_1} + \sum_{i_1=1}^{n}\sum_{i_2=1}^{n} h_{s,i_1} h_{i_1,i_2} a_{i_2} + ...$$
$$+ \sum_{i_1=1}^{n}\sum_{i_2=1}^{n} ... \sum_{i_k=1}^{n} h_{s,i_1} h_{i_1,i_2} ... h_{i_{k-1},i_k} a_{i_k} + ...$$

⇛ If you compare this expression with the Neumann series we will they are the same so:

$$x_i^0 = E\{X(\gamma)|i_0 = i\}$$

To sum up:

We have proven that solving a linear system can be represented by an expectation value of the random variable $X(\gamma)$. The error is computed using standard deviation equation.

Neumann-Ulam method

- For example lets try to solve this equation system:

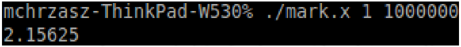
$$\overrightarrow{x} = \begin{pmatrix} 1.5 \\ -1.0 \\ 0.7 \end{pmatrix} + \begin{pmatrix} 0.2 & 0.3 & 0.1 \\ 0.4 & 0.3 & 0.2 \\ 0.3 & 0.1 & 0.1 \end{pmatrix} \overrightarrow{x}$$

- The solution is $\overrightarrow{x}_0 = (2.154303, 0.237389, 1.522255)$.

- The propability matrix h_{ij} has the shape:

i/j	1	2	3	0
1	0.2	0.3	0.1	0.4
2	0.4	0.3	0.2	0.1
3	0.3	0.1	0.1	0.5

- An example solution:



```
mchrzasz-ThinkPad-W530% ./mark.x 1 1000000
2.15625
```

Neumann-Ulam dual method

- The problem with Neumann-Ulam method is that you need to repeat it for each of the coordinates of the $\overrightarrow{x}_0$ vector.
- The dual method calculates the whole $\overrightarrow{x}_0$ vector.
- The algorithm:
  - On the indexes: $\{0, 1, ..., n\}$ we set a probability distribution: $q_1, q_2, ..., q_n$, $q_i > 0$ and $\sum_{i=1}^{n} q_i = 1$.
  - The starting point we select from q_i distribution.
  - If in t time we are in i_t state then with probability $p(i_{t+1}|i_t) = h_{i_{t+1},i_t}$ in $t+1$ we will be in state i_{t+1}. For $i_{t+1} = 0$ we define the probability: $h_{0,i_t} = 1 - \sum_{j=1}^{n} h_{j,i_t}$. Here we also assume that $h_{j,i_t} > 0$.
  - NOTE: there the matrix is transposed compared to previous method: H^T.
  - Again we end our walk when we are at state 0.
  - For the trajectory: $\gamma = (i_0, i_1, ..., i_k, 0)$, we assign the vector:

$$\overrightarrow{Y}(\gamma) = \frac{a_{i_0}}{q_{i_0} p(0|i_k)} \widehat{e}_{i_k} \in \mathcal{R}^n$$

- The solution will be : $\overrightarrow{x}^0 = \frac{1}{N} \sum \overrightarrow{Y}(\gamma)$

Neumann-Ulam dual method

- Let’s try to solve the equation system:

$$\overrightarrow{x} = \begin{pmatrix} 1.5 \\ -1.0 \\ 0.7 \end{pmatrix} + \begin{pmatrix} 0.2 & 0.3 & 0.1 \\ 0.4 & 0.3 & 0.2 \\ 0.1 & 0.1 & 0.1 \end{pmatrix} \overrightarrow{x}$$

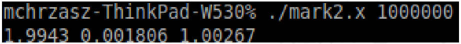
- The solution is: $\overrightarrow{x}_0 = (2.0, 0.0, 1.0)$.
- Let’s put the initial probability as constant:

$$q_1 = q_2 = q_3 = \frac{1}{3}$$

- The propability matrix h_{ij} has the shape:

i/j	1	2	3	4
1	0.2	0.4	0.1	0.3
2	0.3	0.3	0.1	0.3
3	0.1	0.2	0.1	0.6

- An example solution:



```
mchrzasz-ThinkPad-W530% ./mark2.x 1000000
1.9943 0.001806 1.00267
```

Generalization, the algorithm

⇛ We set the P matrix in a arbitrary way.
⇛ If in the t moment the point is in the i_t state, then with the probability $p_{i_t,i_{t+1}}$ he can go to i_{t+1} state.
⇛ We stop the walk once we reach 0.
⇛ For the given trajectory we assign the value: $X(\gamma_k)$
⇛ We repeat the procedure N times and take the mean and RMS.
⇛ We repeat this also for every of the $\overrightarrow{x}^0$ vector components.

Partial differential equations, intro

⇛ Let's say we are want to describe a point that walks on the $\mathbb{R}$ axis:

- At the beginning $(t = 0)$ the particle is at $x = 0$
- If in the t the particle is in the x then in the time $t + 1$ it walks to $x + 1$ with the known probability p and to the point $x - 1$ with the probability $q = 1 - p$.
- The moves are independent.

⇛ So let's try to described the motion of the particle.

⇛ The solution is clearly a probabilistic problem. Let $\nu(x, t)$ be a probability that at time t particle is in position x. We get the following equation:

$$\nu(x, t + 1) = p\nu(x - 1, t) + q\nu(x + 1, t)$$

with the initial conditions:

$$\nu(0, 0) = 1, \qquad \nu(x, 0) = 0 \ \text{ if } x \neq 0.$$

⇛ The above functions describes the whole system (every (t, x) point).

Partial differential equations, intro

⇛ Now in differential equation language we would say that the particle walks in steps of Δx in times: $k\Delta t$, $k = 1, 2, 3....$:

$$\nu(x, t + \Delta t) = p\nu(x - \Delta x, t) + q\nu(x + \Delta x, t).$$

⇛ To solve this equation we need to expand the $\nu(x, t)$ funciton in the Taylor series:

$$\begin{aligned}\nu(x,t) + \frac{\partial \nu(x,t)}{\partial t}\Delta t = p\nu(x,t) - p\frac{\partial \nu(x,t)}{\partial x}\Delta x + \frac{1}{2}p\frac{\partial^2 \nu(x,t)}{\partial x^2}(\Delta x)^2 \\ + q\nu(x,t) + q\frac{\partial \nu(x,t)}{\partial x}\Delta x + \frac{1}{2}q\frac{\partial^2 \nu(x,t)}{\partial x^2}(\Delta x)^2\end{aligned}$$

⇛ From which we get:

$$\frac{\partial \nu(x,t)}{\partial t}\Delta t = -(p-q)\frac{\partial \nu(x,t)}{\partial x}\Delta x + \frac{1}{2}\frac{\partial^2 \nu(x,t)}{\partial x^2}(\Delta x)^2$$

⇛ Now We divide the equation by Δt and take the $\Delta t \to 0$:

$$(p-q)\frac{\Delta x}{\Delta t} \to 2c, \qquad \frac{(\Delta x)^2}{\Delta t} \to 2D,$$

⇛ We get the Fokker-Planck equation for the diffusion with current:

$$\frac{\partial \nu(x,t)}{\partial t} = -2c\frac{\partial \nu(x,t)}{\partial x} + D\frac{\partial^2 \nu(x,t)}{\partial x^2}$$

Laplace equation, Dirichlet boundary conditions

⇛ The aforementioned example shows the way to solve the partial differential equation using Markov Chain MC.
⇛ We will see how different classes of partial differential equations can be approximated with a Markov Chain MC, whose expectation value is the solution of the equation.
⇛ The Laplace equation:

$$\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} + ... + \frac{\partial^2 u}{\partial x_k^2} = 0$$

The $u(x_1, x_2, ..., x_k)$ function that is a solution of above equation we call harmonic function. If one knows the values of the harmonic function on the edges $\Gamma(D)$ of the D domain one can solve the equation.

The Dirichlet boundary conditions:

Find the values of $u(x_1, x_2, ..., x_k)$ inside the D domain knowing the values of the edge are given with a function:

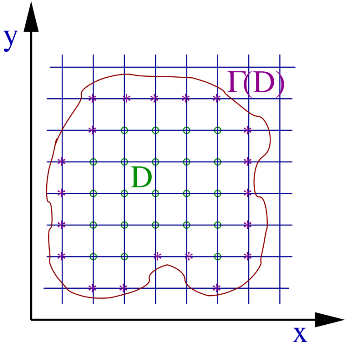
$$u(x_1, x_2, ..., x_k) = f(x_1, x_2, ..., x_k) \in \Gamma(D)$$

⇛ Now I am lazy so I put $k = 2$ but it's the same for all k!

Laplace equation, Dirichlet boundary conditions

⇛ We will put the Dirichlet boundary condition as a discrete condition:

- The domain D we put a lattice with distance h.
- Some points we treat as inside (denoted with circles). Their form a set denoted D^*.
- The other points we consider as the boundary points and they form a set $\Gamma(D)$.



⇛ We express the second derivatives with the discrete form:

$$\frac{\frac{u(x+h)-u(x)}{h}-\frac{u(x)-u(x-h)}{h}}{h}=\frac{u(x+h)-2u(x)+u(x-h)}{h^2}$$

⇛ Now we choose the units so $h=1$.

Laplace equation, Dirichlet boundary conditions

The Dirichlet condition in the discrete form:

Find the u^* function which obeys the differential equation:

$$U^*(x,y) = \frac{1}{4}\left[u^*(x-1,y) + u^*(x+1,y) + u^*(x,y-1) + u^*(x,y+1)\right]$$

in all points $(x,y) \in D^*$ with the condition:

$$u^*(x,y) = f^*(x,y), \quad (x,y) \in \Gamma(D^*)$$

where $f^*(x,y)$ is the discrete equivalent of $f(x,y)$ function.

$\Rrightarrow$ We consider a random walk over the lattice $D^* \cup \Gamma(D^*)$.

- In the $t=0$ we are in some point $(\xi,\eta) \in D^*)$
- If at the t the particle is in (x,y) then at $t+1$ it can go with equal probability to any of the four neighbour lattices: $(x-1,y)$, $(x+1,y)$, $(x,y-1)$, $(x,y+1)$.
- If the particle at some moment gets to the edge $\Gamma(D^*$ then the walk is terminated.
- For the particle trajectory we assign the value of: $\nu(\xi,\eta) = f^*(x,y)$, where $(x,y) \in \Gamma(D^*)$.

Laplace equation, Dirichlet boundary conditions

⇛ Let $p_{\xi,\eta}(x,y)$ be the probability of particle walk that starting in (ξ,η) to end the walk in (x,y).

⇛ The possibilities:

1. The point $(\xi,\eta) \in \Gamma(D^*)$. Then:

$$p_{\xi,\eta}(x,y) = \begin{cases} 1, & (x,y) = \xi,\eta) \\ 0, & (x,y) \neq \xi,\eta) \end{cases} \tag{1}$$

2. The point $(\xi,\eta) \in D^*$:

$$p_{\xi,\eta}(x,y) = \frac{1}{4}\left[p_{\xi-1,\eta}(x,y) + p_{\xi+1,\eta}(x,y) + p_{\xi,\eta-1}(x,y) + p_{\xi,\eta+1}(x,y)\right] \tag{2}$$

this is because to get to (x,y) the particle has to walk through one of the neighbours: $(x-1,y)$, $(x+1,y)$, $(x,y-1)$, $(x,y+1)$.

⇛ The expected value of the $\nu(\xi,\eta)$ is given by equation:

$$E(\xi,\eta) = \sum_{(x,y)\in\Gamma^*} p_{\xi,\eta}(x,y) f^*(x,y) \tag{3}$$

where the summing is over all boundary points

Laplace equation, Dirichlet boundary conditions

⇛ Now multiplying the 2 by $f^*(x, y)$ and summing over all edge points (x, y):

$$E(\xi, \eta) = \frac{1}{4}\left[E(\xi - 1, \eta) + E(\xi + 1, \eta) + E(\xi, \eta - 1) + E(\xi, \eta + 1)\right]$$

⇛ Putting now 1 to 3 one gets:

$$E(x, y) = f^*(x, y), \quad (\xi, \eta) \in \Gamma(D^*)$$

⇛ Now the expected value solves identical equation as our $u^*(x, y)$ function. From this we conclude:

$$E(x, y) = u^*(x, y)$$

⇛ The algorithm:

- We put a particle in (x, y).
- We observe it's walk up to the moment when it's on the edge $\Gamma(D^*)$.
- We calculate the value of f^* function in the point where the particle stops.
- Repeat the walk N times taking the average afterwards.

Important:

One can show the the error does not depend on the dimensions!

Backup