Sampling: intro to MC & MD

Reminder: Want to compute $\langle A \rangle = \int dx\, A(x) P(x)$, but cannot do $\langle A \rangle \approx \sum_{i=1}^{N} A(x_i) P(x_i) \Delta x$ in high dimensions

Idea of both MC & MD: rather than generate x_i on a grid w/ prob 1, generate $x_i \propto P(x)$ somehow

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

One way to do this is to generate a "Markov Chain", rule $x_i \to x_{i+1}$ that only depends on x_i (not $x_1 \ldots x_{i-1}$)

If we also satisfy detailed balance

$$P(x_i) P(x_i \to x_{i+1}) = P(x_{i+1}) P(x_{i+1} \to x_i)$$

$P(x_i \to x_{i+1})$ has 2 parts

$$= P_{gen}(x_i \to x_{i+1})\, P_{acc}(x_i \to x_{i+1})$$

$$P_{acc}(x_i \to x_{i+1}) = \underbrace{\frac{P(x_{i+1})}{P(x_i)} \frac{P_{gen}(x_{i+1} \to x_i)}{P_{gen}(x_i \to x_{i+1})}}_{r(x_i \to x_{i+1})} P_{acc}(x_{i+1} \to x_i)$$

How can we do this? One way,

Metropolis rule $P_{acc}(x_i \to x_{i+1}) = \min(1, r(x_i \to x_{i+1}))$

Algorithm Start @ x_i

- propose x_{i+1} with prop $P_{gen}(x_i \to x_{i+1})$
- gen $r \in (0,1)$
- if $r < P_{acc}(x \to x_{i+1})$, move to x_{i+1}
- if not, keep x_i (for statistics) as x_{i+1}

Tune P_{acc} to get acept prob. Often move is uniform $x_{i+1} = x_i + \xi \cdot \underset{(0,1)}{r_2}$

For canonical, if uniform gen

$$r(x_i \to x_{i+1}) = P(x_{i+1})/P(x_i) = e^{-\beta(E(x_{i+1}) - E(x_i))}$$

Another more advanced example:

Turns out, const pressure

$$\Delta(N,P,T) = \frac{1}{V_0}\int_0^\infty dV\, e^{-\beta PV}\, Q(N,V,T)$$

(like how $Q = \frac{1}{\varepsilon_0}\int_0^\infty d\varepsilon\, e^{-\beta\varepsilon}\, \Omega(N,V,\varepsilon)$)

What if we want to do MC & keep const pressure, have to adjust the volume

$$\Delta(N,P,T) \propto \int_0^\infty dV e^{-\beta PV} \int dq^{3N} e^{-\beta u(\vec{q})}$$

$$= \int_0^\infty dV e^{-\beta PV} \cdot V^N \int ds^{3N} e^{-\beta u(\vec{s} V^{1/3})}$$

note: $V^N = e^{N\log V}$

$$= \int_0^\infty dV \int_0^1 ds^{3N} \cdot e^{-\beta(PV + u(sV^{1/3}) - \frac{N}{\beta}\log V)}$$

so now $P(\text{"}x\text{"}) = \exp[-\beta(PV + u - \frac{N}{\beta}\log V)] / \Delta(N,P,T)$

So can make Volume moves,
$V_{i+1} = V_i + \xi$, $\xi \in (-\delta, \delta)$

$$A(V_i \to V_{i+1}) = \min[1, \exp[-\beta P(V_{i+1} - V_i) + N\log(V_{i+1}/V_i) - \beta(\varepsilon_{i+1} - \varepsilon_i)]]$$

$$\varepsilon_{i+1} = u(\vec{s}\, V_{i+1}^{1/3}), \quad \varepsilon_i = u(\vec{s}, V_i^{1/3})$$

This is probably an expensive calculation unless
$u(\lambda\vec{q}) = \lambda^n u(\vec{q})$

Conclusion: Monte Carlo is a very powerful (Easy) and general method for sampling from a distribution, not just in chemistry. Useful in statistics, data science, machine learning ...

Downsides: (1) only one thing happens at a time sometimes many moves rejected (inefficient)

(2) Generally good for <u>static</u> properties, not likely any connection to real time, so not good if you want info about kinetics

$\Rightarrow$ Molecular dynamics is an alternative idea

Solve newton's equations approximately, with the same idea of computing $\langle A \rangle = \int dx\, P(x) A(x)$

We know from before, if we have $\{\vec{q}(0), \vec{p}(0)\}$ and $\mathcal{H}$, then we can generate $\{\vec{q}(t), \vec{p}(t)\}$ at any time t using $\vec{F} = m\ddot{\vec{q}}$, $\vec{F}_i = -\partial U / \partial \vec{q}_i$

or alternatively $\partial \mathcal{H} / \partial q_i = -\dot{p}_i$ $\quad \partial \mathcal{H} / \partial p_i = \dot{q}_i$

If the system is "ergodic", then as $t \to \infty$ will sample all configurations $\{p(t), q(t)\}$ st $\mathcal{H}(p(t), q(t)) = \mathcal{E}$ with equal prob, ie $P(\vec{X}) = P(\vec{p}, \vec{q}) = 1/\Omega(N,V,\mathcal{E})$

so $\langle A \rangle = C \int d\vec{p} \int d\vec{q}\, A(p,q)\, \delta(\mathcal{H}(p,q) - \mathcal{E}) / \Omega(N,V,\mathcal{E})$

$$\Omega(N,V,\mathcal{E}) = C \int d\vec{p} \int d\vec{q}\, \delta(\mathcal{H}(p,q) - \mathcal{E})$$

& if ergodic $\langle A \rangle = \lim_{T \to \infty} \frac{1}{T} \int_0^T dt\, A[p(t), q(t)]$

In practice, need: 1) initial starting cfg (gen vel from Boltzmann dist)
2) interaction energy

Remember we previously said $\frac{dA}{dt} = \{A, H\} = \sum_{i=1}^{N} \left(\frac{\partial A}{\partial q_i} \underbrace{\frac{dq_i}{dt}}_{= \partial H/\partial p_i} + \frac{\partial A}{\partial p_i} \underbrace{\frac{dp_i}{dt}}_{= -\partial H/\partial q_i} \right)$

and defined $i\mathcal{L} A = \{H, A\}$, $i\mathcal{L} = \{H, \_\}$

so $dA/dt = -i\mathcal{L} A$ so

formally $A(t) = e^{-i\mathcal{L}t} A(0)$, but we cannot solve this for almost any problem, and so we can use a computer to solve these equations approximately

First, look at the way we can do this by looking at Newtonian dynamics as a taylor series in position at small time

(1) $\vec{q}(\tau+d\tau) \approx \vec{q}(\tau) + d\tau \left.\frac{d\vec{q}}{dt}\right|_{t=\tau} + \frac{d\tau^2}{2} \left.\frac{d^2\vec{q}}{dt^2}\right|_{t=\tau} + O(d\tau^3)$

$\approx \vec{q}(\tau) + d\tau\, \vec{v}(\tau) + d\tau^2/_2\, \vec{a}(t)$ [Remember: $d = vt + \frac{1}{2}at$

Remember $a_i(t) = -\frac{\partial}{\partial q_i} U(q(t)) \cdot \frac{1}{m_i} \doteq F_i/m$

Also would need $v(\tau + d\tau)$, can do by finite diff

[$\vec{v} = d\vec{q}/d\tau \approx \frac{\vec{q}(t+d\tau) - \vec{q}(t)}{d\tau}$] or by expanding

$\vec{v}(t+d\tau) \approx \vec{v}(t) + d\tau\, \vec{a}(t) + O(d\tau^2)$, but people came up with schemes that are better.
