

Laboratorio 3: Algoritmos estocásticos usando cadenas de Markov

Problema 1: Modelo de Ising en \mathbb{Z}^2

Queremos modelar el ferromagnetismo en una placa metálica plana idealizada. Específicamente para $N \in \mathbb{N}$ (grande), las moléculas de la placa se ubican en la grilla 2-dimensional $\Lambda = \Lambda_N = \{-N, \dots, N\}^2 \subseteq \mathbb{Z}^2$, y cada molécula posee un momento magnético o *spin*, el cual puede estar orientado hacia arriba o hacia abajo. Por lo tanto, el conjunto de posibles configuraciones es $E_N = \{-1, 1\}^\Lambda$. Dado $x \in E_N$, y $m \in \Lambda$, denotamos $x(m) \in \{-1, 1\}$ el spin del sitio m en la configuración x . Trabajaremos en el espacio E de configuraciones con spin fijo hacia arriba en el borde:

$$E = \{x \in E_N : x(m) = 1, \forall m \in \partial\Lambda\}, \quad \text{donde } \partial\Lambda = \Lambda_N \setminus \Lambda_{N-1}.$$

En un material ferromagnético, los spins de sitios cercanos tienen tendencia a alinearse; es decir, spins iguales en sitios contiguos tienen asociada menor energía. Específicamente, la energía de una configuración $x \in E_N$ viene dada por

$$H(x) = \sum_{m \sim m'} [x(m) - x(m')]^2,$$

donde $m \sim m'$ denota que m y m' son vecinos en la grilla, es decir, que están a distancia 1 (cada par $m, m' \in \Lambda$ aparece una sola vez en la sumatoria). Supondremos que la probabilidad de que el sistema se encuentre en la configuración $x \in E$ está dada por

$$\pi_x = \frac{e^{-\beta H(x)}}{Z(\beta)},$$

para $\beta > 0$ dado ($1/\beta$ es la temperatura), y $Z(\beta) = \sum_{y \in E} e^{-\beta H(y)}$ es la constante de normalización. Observe que lo anterior hace improbables las configuraciones con mayor energía, lo cual es consecuente con lo que ocurre en los sistemas físicos. Se desea simular realizaciones de la distribución $\pi \in \mathcal{P}(E)$.

1. Calcule $|E|$. Argumente por qué es imposible en la práctica calcular $Z(\beta)$ explícitamente, incluso para N pequeño (por ejemplo, $N = 10$).

Debido a lo anterior, no es posible simular π directamente. Por este motivo, utilizaremos un algoritmo tipo *Markov chain Monte Carlo* (MCMC) para realizar simulaciones aproximadas de π , según lo descrito en cátedra. Para esto, siga los siguientes pasos.

2. Consideremos el grafo G sobre E en el cual xy es una arista de G si y sólo si x y y difieren en exactamente un sitio, es decir, si y sólo si

$$\exists m_0 \in \Lambda \setminus \partial\Lambda, \text{ tal que } x(m_0) = -y(m_0), \text{ y además } x(m) = y(m), \forall m \neq m_0.$$

Escriba la matriz estocástica R asociada a este grafo para el algoritmo de Metropolis y el de Gibbs. Obtenga una expresión explícita simple para $\frac{\pi_y R_{yx}}{\pi_x R_{xy}}$ en ambos casos ¿Cuál algoritmo es más conveniente usar y por qué?

3. Describa el algoritmo MCMC correspondiente a este caso usando pseudo-código. En base a él, programe un método $X = \text{Ising}(N, \beta, n_f)$ que simule n_f pasos de la cadena, grafique su estado cada cierta cantidad de pasos (mostrando la grilla Λ y asociando un color al spin -1 y otro a 1), y retorne el estado final X , en caso que n_f sea finito. Escoja los spins iniciales independientes con ley $2 \cdot \text{Bernoulli}(p) - 1$, para algún $p \in (1/2, 1)$ (por ejemplo $p = 2/3$ ó $p = 3/4$). Haga esto para 2 valores de p distintivos. Comente la diferencia o similitud observadas entre ambos casos y la relevancia de dicha elección. Fije un valor para todo lo que viene.
4. En lo que sigue, fije $N \in \{50, \dots, 200\}$ (lo más grande que se pueda, mientras la simulación sea fluida). Observe el comportamiento de la cadena en el tiempo largo para un β pequeño y otro grande (digamos, escoja un $\beta < 0,1$ y otro $\beta > 5$). Grafique y comente.

Lo observado en el punto anterior se conoce como *transición de fase*: existe un valor crítico $\beta_C > 0$ tal que para cualquier $\beta > \beta_C$ y N grande, π asigna probabilidad casi 1 a configuraciones con spin hacia arriba en la gran mayoría de los sitios (con la condición de borde que usamos), mientras que cuando $\beta < \beta_C$ se observa coexistencia de ambos spins. A continuación estudiaremos este fenómeno.

5. Fije n_f grande (en el orden de los millones), de modo que el algoritmo tenga tiempo de acercarse a π . Fije una malla del intervalo $[0, 1]$ de distintos valores de β . Para cada β en la malla, obtenga el estado X de la cadena luego de n_f pasos, y calcule el *spin medio* $s = \frac{1}{|\Lambda|} \sum_{m \in \Lambda} X(m)$. Grafique s en función de β , y estime visualmente el valor crítico β_C .
6. Repita lo anterior en un intervalo más pequeño centrado en su estimación de β_C , y con n_f más grande aún, de modo de obtener una estimación más fina. Grafique. Averigüe el valor exacto de β_C y compárelo con su estimación.

Problema 2: Problema del vendedor viajero

Considere un conjunto $\{1, 2, \dots, N\}$ de ciudades en el dominio plano $[0, 1]^2$. El problema del vendedor viajero consiste en encontrar un ciclo que recorra todas las ciudades una y solo una vez, partiendo y terminando en la primera, que minimice la distancia recorrida.

Sea $E = \{\sigma \in S_N : \sigma(1) = 1\}$ el conjunto de todas las posibles rutas que empiezan en 1. Es fácil verificar que $|E| = (N - 1)!$, por lo que si consideramos el problema con 15 ciudades, la cantidad de rutas posibles es 87.178.291.200, por lo cual es claro que es prácticamente imposible recorrer todas las posibilidades. La idea es construir un algoritmo estocástico, denominado *recocido simulado* (*simulated annealing*), para minimizar la función de distancia total recorrida

$$w(\sigma) = \sum_{i=1}^N d(\sigma(i), \sigma(i+1))$$

en donde $d : \{1, \dots, N\}^2 \rightarrow \mathbb{R}_+$ es la distancia euclidiana usual entre dos ciudades, y se usa la convención $\sigma(N+1) = 1$ para $\sigma \in E$. Definimos el grafo G sobre E dado por la siguiente relación de adyacencia: $\sigma \sim \tau$ si y sólo si τ se obtiene permutando exactamente 2 ciudades de σ .

1. Programe una función que genere N ciudades uniformemente distribuidas en $[0, 1]^2$, y que genere luego la matriz $D = (D_{ij})_{i,j=1}^N$ de distancias, donde D_{ij} es la distancia entre la ciudad i y la ciudad j .
2. Programe una función que dado un camino que recorre las N ciudades, en el orden dado por la permutación σ , grafique dicho camino.

3. Se define $osc_K(w) = \max\{w(\tau) - w(\sigma) : \sigma \sim \tau\}$. Dé una cota superior para $osc_K(w)$ para cada N , que no dependa de la posición de las ciudades.
4. Considere una sucesión de temperaturas inversas $\beta_n = \frac{1}{C} \ln(n+e)$ con $C > (N-1)osc_K(w)$ y una cadena de Markov (X_n) no homogénea tal que, en el tiempo n , su matriz de transición está dada por la matriz de la cadena $(X_k^{\beta_n})$, según el método visto en cátedra. Es decir: dado $X_n = \sigma$, se escoge un vecino $\tau \sim \sigma$ uniformemente al azar, y con probabilidad $e^{-\beta_n(w(\tau)-w(\sigma))} \wedge 1$ se define $X_{n+1} = \tau$; si no, se mantiene $X_{n+1} = \sigma$. Se puede probar que dicha cadena converge en probabilidad a una variable aleatoria distribuida uniformemente en el conjunto de mínimos globales de la función w . Implemente un método que simule esta cadena para un estado inicial que usted escoja.
5. Para $N = 20$ ciudades fijas, encuentre un mínimo global aproximado de la función w . Pruebe con sucesiones β_n de distintas formas, por ejemplo lineal, cuadrática, exponencial, etc. Grafique en cada caso la evolución de la función w evaluada en el estado de la cadena, durante el tiempo de ejecución del algoritmo (el que usted deberá determinar dependiendo de la sucesión β_n escogida). Grafique para algunos tiempos representativos los caminos respectivos.