Metropolis-Hastings: la formalización del algoritmo

¿Cómo interpretamos la salida del algoritmo?

- Vamos a demostrar que para cada $j \in \{1, \ldots, N\}$, se tiene que a_j es generado según la distribución de probabilidades $Q^j \vec{x}_0$, donde $\vec{x}_0$ es un vector tal que $\vec{x}_0[a_0] = 1$ y $\vec{x}_0[b] = 0$ para todo $b \neq a_0$

En este contexto, Q es la matriz que describimos antes, con distribución estacionaria $\vec{\pi}$

- Así, el algoritmo toma un vector inicial $\vec{x}_0$ y realiza N transiciones según la matriz de transición Q

Markov Chain Monte Carlo: algunos comentarios

El algoritmo de Metropolis-Hastings nos entrega una estrategia general para muestrear de acuerdo a una distribución de interés $\vec{\pi}$

- Si N es suficientemente grande, entonces a_N será generado de acuerdo a una distribución similar a la estacionaria

Sin embargo, el algoritmo no entrega a priori ningún criterio respecto a qué se entiende por suficientemente grande.

- ¿Cuál debe ser el valor de N para que la cadena de Markov se acerque lo suficiente a la distribución estacionaria?

Markov Chain Monte Carlo: algunos comentarios

Por ejemplo, para aplicar el algoritmo a problemas de conteo necesitamos un generador casi uniforme.

- Eso significa que $\vec{\pi}$ es la distribución uniforme, y que queremos estar muy cerca de la distribución estacionaria
- A la vez, no queremos que N sea demasiado grande ya que queremos obtener un algoritmo de tiempo polinomial

Markov Chain Monte Carlo: algunos comentarios

Otro tema importante es el de la convergencia de la cadena de Markov a la distribución $\vec{\pi}$

- El algoritmo de Metropolis-Hastings nos asegura que $\vec{\pi}$ es distribución estacionaria, pero no nos asegura que sea la única
- Aún más, el algoritmo no nos asegura que la cadena de Markov converja a $\vec{\pi}$ independientemente del vector $\vec{x}_0$ inicial

Para conseguir estas garantías se debe exigir más condiciones a la cadena de Markov caracterizada por Q

- Más adelante vamos a analizar en detalle cómo conseguir estas garantías manipulando la matriz P

Markov Chain Monte Carlo: algunos comentarios

Cómo mencionamos antes, en los problemas de conteo consideramos $\vec{\pi}$ como la distribución uniforme.

- Entonces, dado que $\vec{\pi}[a] = \vec{\pi}[b]$ para todo a, b, la condición de reversibilidad se reduce a que $P[a, b] = P[b, a]$

Por lo tanto, no siempre será necesario recurrir al algoritmo de Metropolis-Hastings en este contexto.

- Nos basta con definir una cadena de Markov cuya matriz de transición sea simétrica, y que además sea irreducible y aperiódica
- Esto puede lograrse en muchos casos. En otros, sin embargo, es necesario recurrir al algoritmo de Metropolis-Hastings

Metropolis-Hastings: la correctitud del algoritmo

Proposición

Sea Ω un conjunto finito, $\vec{\pi}$ una distribución sobre Ω y P la matriz de transición de una cadena de Markov con conjunto de estados Ω. Además, sea Q una matriz tal que:

$$Q[a,b] \quad = \quad \begin{cases} P[a,b] \cdot \alpha[a,b] + r(b) & \text{si } a = b \\ P[a,b] \cdot \alpha[a,b] & \text{si } a \neq b \end{cases}$$

para todo $a, b \in \Omega$, donde $r(b) = 1 - \sum_{x \in \Omega} P[x,b] \cdot \alpha[x,b]$ y

$$\alpha[a,b] \quad = \quad \begin{cases} \text{mín} \left\{ \dfrac{P[b,a] \cdot \vec{\pi}[a]}{P[a,b] \cdot \vec{\pi}[b]}, 1 \right\} & \text{si } P[a,b] \cdot \vec{\pi}[b] > 0 \\ 1 & \text{en otro caso} \end{cases}$$

Entonces Q es la matriz de transición de una cadena de Markov y $Q[d,c] \cdot \vec{\pi}[c] = Q[c,d] \cdot \vec{\pi}[d]$ para todo $c, d \in \Omega$

Metropolis-Hastings: la correctitud del algoritmo

Ejercicio

Demuestre la proposición anterior.

Metropolis-Hastings: la correctitud del algoritmo

Ahora, solo falta demostrar que el algoritmo de Metropolis-Hastings efectúa las transiciones de acuerdo a las probabilidades especificadas por la matriz Q

En el siguiente teorema, consideramos toda la notación dada en la proposición anterior y en la descripción del algoritmo de Metropolis-Hastings.

Metropolis-Hastings: la correctitud del algoritmo

Teorema

Para todo $c \in \Omega$ *y* $j \in \{0, 1, \ldots, N\}$ *se tiene que:*

$$\mathbf{Pr}(a_j = c) \quad = \quad (Q^j \vec{x}_0)[c]$$

Demostraremos el teorema sin requerir que $\vec{x}_0$ sea un vector canónico.

- $\vec{x}_0$ representa entonces la distribución de probabilidades de acuerdo a la cual se eligió a_0

La demostración del teorema

Haremos la demostración por inducción, partiendo por el caso base $j = 0$

La demostración del caso base es trivial, dado que $Q^0 = I$ y, por hipótesis, a_0 se elige de acuerdo a la distribución de probabilidades representada por $\vec{x}_0$, o sea, para todo $c \in \Omega$ se tiene:

$$\mathbf{Pr}(a_0 = c) \;=\; \vec{x}_0[c] \;=\; (Q^0 \vec{x}_0)[c]$$

La demostración del teorema

Ahora, suponemos que el resultado es cierto para $j = k$, y consideramos $j = k + 1$

Sea $c \in \Omega$. Tenemos entonces que:

$$\mathbf{Pr}(a_{k+1} = c) = \mathbf{Pr}(a_{k+1} = c \wedge u \leq \alpha[b, a_k]) + \mathbf{Pr}(a_{k+1} = c \wedge u > \alpha[b, a_k])$$

Denotamos el primer término como p_1 y el segundo como p_2

- Estudiamos a continuación cada uno de estos términos

La demostración del teorema

$$
\begin{aligned}
p_1 &= \mathbf{Pr}(a_{k+1} = c \wedge u \leq \alpha[b, a_k]) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(a_{k+1} = c \wedge u \leq \alpha[b, a_k] \wedge b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(a_{k+1} = c \mid u \leq \alpha[b, a_k] \wedge b = y \wedge a_k = x) \cdot \\
&\qquad\qquad \mathbf{Pr}(u \leq \alpha[b, a_k] \wedge b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(y = c) \cdot \mathbf{Pr}(u \leq \alpha[b, a_k] \mid b = y \wedge a_k = x) \cdot \\
&\qquad\qquad \mathbf{Pr}(b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(y = c) \cdot \alpha[y, x] \cdot \mathbf{Pr}(b = y \mid a_k = x) \cdot \mathbf{Pr}(a_k = x)
\end{aligned}
$$

La demostración del teorema

Por lo tanto, tenemos por hipótesis de inducción que:

$$
\begin{aligned}
p_1 &= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(y = c) \cdot \alpha[y, x] \cdot P[y, x] \cdot \mathbf{Pr}(a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(y = c) \cdot \alpha[y, x] \cdot P[y, x] \cdot (Q^k \vec{x}_0)[x] \\
&= \sum_{x \in \Omega} \alpha[c, x] \cdot P[c, x] \cdot (Q^k \vec{x}_0)[x]
\end{aligned}
$$

La demostración del teorema

Por otra parte, tenemos que:

$$
\begin{aligned}
p_2 &= \mathbf{Pr}(a_{k+1} = c \wedge u > \alpha[b, a_k]) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(a_{k+1} = c \wedge u > \alpha[b, a_k] \wedge b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(a_{k+1} = c \mid u > \alpha[b, a_k] \wedge b = y \wedge a_k = x) \cdot \\
&\qquad\qquad \mathbf{Pr}(u > \alpha[b, a_k] \wedge b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(x = c) \cdot \mathbf{Pr}(u > \alpha[b, a_k] \mid b = y \wedge a_k = x) \cdot \\
&\qquad\qquad \mathbf{Pr}(b = y \wedge a_k = x) \\
&= \sum_{x \in \Omega} \sum_{y \in \Omega} \mathbf{Pr}(x = c) \cdot (1 - \alpha[y, x]) \cdot \mathbf{Pr}(b = y \mid a_k = x) \cdot \mathbf{Pr}(a_k = x)
\end{aligned}
$$

La demostración del teorema

Por lo tanto, tenemos por hipótesis de inducción que:

$$
\begin{aligned}
p_2 &= \sum_{x\in\Omega}\sum_{y\in\Omega} \mathbf{Pr}(x=c)\cdot(1-\alpha[y,x])\cdot P[y,x]\cdot \mathbf{Pr}(a_k=x) \\
&= \sum_{x\in\Omega}\sum_{y\in\Omega} \mathbf{Pr}(x=c)\cdot(1-\alpha[y,x])\cdot P[y,x]\cdot (Q^k\vec{x}_0)[x] \\
&= \sum_{y\in\Omega}(1-\alpha[y,c])\cdot P[y,c]\cdot (Q^k\vec{x}_0)[c] \\
&= (Q^k\vec{x}_0)[c]\cdot\sum_{y\in\Omega}(1-\alpha[y,c])\cdot P[y,c] \\
&= (Q^k\vec{x}_0)[c]\cdot\left(\sum_{y\in\Omega} P[y,c]-\sum_{y\in\Omega}\alpha[y,c]\cdot P[y,c]\right) \\
&= (Q^k\vec{x}_0)[c]\cdot\left(1-\sum_{y\in\Omega}\alpha[y,c]\cdot P[y,c]\right) \\
&= (Q^k\vec{x}_0)[c]\cdot r(c)
\end{aligned}
$$

La demostración del teorema

Tenemos entonces que:

$$
\begin{aligned}
\mathbf{Pr}(a_{k+1} = c) &= p_1 + p_2 \\
&= \left(\sum_{x \in \Omega} \alpha[c, x] \cdot P[c, x] \cdot (Q^k \vec{x}_0)[x] \right) + (Q^k \vec{x}_0)[c] \cdot r(c) \\
&= (\alpha[c, c] \cdot P[c, c] \cdot (Q^k \vec{x}_0)[c] + r(c) \cdot (Q^k \vec{x}_0)[c]) \; + \\
&\qquad \sum_{x \in \Omega \,:\, x \neq c} \alpha[c, x] \cdot P[c, x] \cdot (Q^k \vec{x}_0)[x] \\
&= Q[c, c] \cdot (Q^k \vec{x}_0)[c] + \sum_{x \in \Omega \,:\, x \neq c} Q[c, x] \cdot (Q^k \vec{x}_0)[x] \\
&= \sum_{x \in \Omega} Q[c, x] \cdot (Q^k \vec{x}_0)[x] \\
&= (Q^{k+1} \vec{x}_0)[c]
\end{aligned}
$$

La demostración del teorema

Con esto queda demostrado el caso inductivo.

Así, queda demostrado el teorema. □

$\vec{\pi}$ como la única distribución estacionaria del algoritmo de Metropolis-Hastings

¿Cómo podemos asegurar que $\vec{\pi}$ es la única distribución estacionaria de la cadena de Markov con matriz de transición Q?

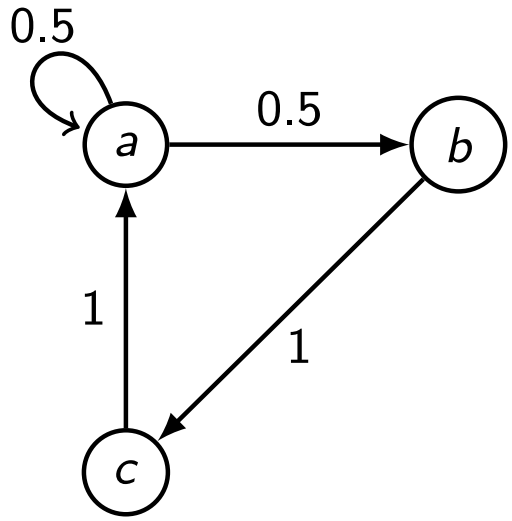
- ¿Cómo podemos asegurar que la cadena de Markov converge a $\vec{\pi}$ independientemente del vector $\vec{x}_0$ inicial?

Para tener las condiciones anteriores la cadena de Markov debe ser irreducible y aperiódica.

- ¿Para lograr esto basta con que la cadena de Markov con matriz de transición P sea irreducible y aperiódica?

No basta partir de una cadena irreducible y aperiódica

Considere la siguiente cadena de Markov:



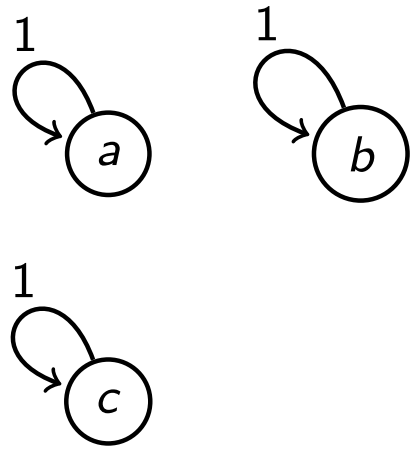
Esta cadena de Markov es irreducible y aperiódica, y su única distribución estacionaria es:

$$\begin{pmatrix} 0.5 \\ 0.25 \\ 0.25 \end{pmatrix}$$

No basta partir de una cadena irreducible y aperiódica

Queremos utilizar el algoritmo de Metropolis-Hastings para obtener a partir de P la distribución uniforme $\vec{\pi}[a] = \vec{\pi}[b] = \vec{\pi}[c] = \frac{1}{3}$ como distribución estacionaria de una cadena de Markov.

En este caso tenemos que $P[a, b] = P[b, c] = P[c, a] = 0$ y $\alpha[b, a] = \alpha[c, b] = \alpha[a, c] = 0$, por lo que el algoritmo genera la siguiente cadena de Markov:



No basta partir de una cadena irreducible y aperiódica

Vale decir, tenemos que: $Q = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$

La cadena de Markov generada por el algoritmo de Metropolis-Hastings no es irreducible ni tiene una única distribución estacionaria.

- Y además converge a distintas distribuciones dependiendo de la distribución de partida

¿Cómo podemos solucionar estos problemas?

Una solución simple y general

Suponga que tenemos una cadena de Markov con conjunto de estados Ω y matriz de transición P

En el algoritmo de Metropolis-Hastings suponemos que es posible hacer muestreo de acuerdo con P

- Vale decir, dado $a \in \Omega$, suponemos que es posible generar de manera aleatoria $b \in \Omega$ de acuerdo a la distribución probabilidad $P[\cdot, a]$
- Más aun, suponemos que b puede ser generado de manera eficiente

Para solucionar los problemas mencionados en las transparencias anteriores vamos a modificar P

- Lo importante es que debe ser posible hacer muestreo de acuerdo con la matriz modificada de manera eficiente

Una solución simple y general

Sea $\theta \in (0, 1)$ y U una matriz de $|\Omega| \times |\Omega|$ tal que

$$U[a, b] \quad = \quad \frac{1}{|\Omega|}$$

Vale decir, U representa una cadena de Markov en la cual es posible moverse de un estado a cualquier otro con distribución uniforme.

Considerando P, θ y U defina:

$$P_\theta \quad = \quad \theta \cdot P + (1 - \theta) \cdot U$$

Una solución simple y general

Tenemos que:

- P_θ es irreducible y aperiodica
- Es posible hacer muestreo de manera eficiente de acuerdo a P_θ si es posible hacer muestreo de manera eficiente de acuerdo a P
- La matriz Q_θ generada por el algoritmo de Metropolis-Hastings es irreducible y aperiódica
  - Por lo tanto, utilizando P_θ solucionamos los problemas mencionados en las transparencias anteriores

Ejercicio

Demuestre las propiedades mencionadas arriba.

Una respuesta a la segunda pregunta: velocidad de convergencia

¿Cómo podemos asegurar que una cadena de Markov convergerá a la distribución estacionaria en un tiempo razonable?

Para responder esta pregunta, primero necesitamos definir una métrica que indique la distancia que hay entre la distribución estacionaria y la distribución de probabilidad de la cadena en un tiempo $t \geq 0$

Distancia de variación

Sea $\{X_t\}_{t \in \mathbb{N}}$ una cadena de Markov con conjunto de estados Ω, matriz de transición P y distribución estacionaria $\vec{\pi}$

Definición

Dado $x \in \Omega$, la distancia de variación en tiempo $t \geq 0$ con respecto a un estado inicial x se define como:

$$\Delta_x(t) \quad = \quad \frac{1}{2} \sum_{y \in \Omega} \left| P^t[y, x] - \vec{\pi}[y] \right|$$

Mixing time de una cadena de Markov

Definición

Dada una cadena de Markov $\{X_t\}_{t\in\mathbb{N}}$ *y* $\varepsilon \in (0,1)$*, el mixing time de* $\{X_t\}_{t\in\mathbb{N}}$ *a partir de un estado inicial* x *se define como:*

$$\tau_x(\varepsilon) \quad = \quad \text{mín}\{t \in \mathbb{N} \mid \forall t' \geq t \,:\, \Delta_x(t') \leq \varepsilon\}$$

¿Cómo podemos acotar el mixing time de una cadena de Markov?

La técnica de los caminos canónicos

Desde ahora en adelante, asumiremos una notación estándar para los elementos de una cadena de Markov $\{X_t\}_{t\in\mathbb{N}}$:

- Ω es su conjunto de estados
- P es su matriz de transición
- $\vec{\pi}$ es su distribución estacionaria

Suponiendo que $\{X_t\}_{t\in\mathbb{N}}$ es reversible, para cada $x, y \in \Omega$ definimos

$$Q(\{x, y\}) \;=\; P[x, y] \cdot \vec{\pi}[y] \;=\; P[y, x] \cdot \vec{\pi}[x]$$

Intuitivamente, $\{x, y\}$ representa a un arco en un grafo no dirigido.

La técnica de los caminos canónicos

Sea $\{X_t\}_{t\in\mathbb{N}}$ una cadena de Markov reversible, irreducible y aperiódica

Asociamos a $\{X_t\}_{t\in\mathbb{N}}$ un grafo $H_{P,\vec{\pi}} = (\Omega, E_{P,\vec{\pi}})$ no dirigido, donde:

$$E_{P,\vec{\pi}} \quad = \quad \{\{x, y\} \in 2^{\Omega} \mid Q(\{x, y\}) > 0\}$$

Además, dados $x, y \in \Omega$, un camino canónico de x a y se define como un camino $\gamma_{x,y} = e_1, \ldots, e_r$ desde x a y en $H_{P,\vec{\pi}}$

- Suponemos que $r \geq 1$, y denotamos al largo r del camino como $|\gamma_{x,y}|$
- Utilizamos la notación $e \in \gamma_{x,y}$ para indicar que e es un arco (no dirigido) en el camino $\gamma_{x,y}$

La técnica de los caminos canónicos

Suponga que por cada par $x, y \in \Omega$ escogemos un camino canónico $\gamma_{x,y}$, y definimos Γ como el conjunto de estos caminos.

Definición

La máxima carga de un arco se define como

$$\rho(\Gamma) = \max_{e \in E_{P,\vec{\pi}}} \left(\frac{1}{Q(e)} \sum_{\gamma_{x,y} \in \Gamma \,:\, e \in \gamma_{x,y}} \vec{\pi}[x] \cdot \vec{\pi}[y] \cdot |\gamma_{x,y}| \right)$$

Intuitivamente, queremos que no haya arcos con mucha carga para no formar *cuellos de botella*.

La técnica de los caminos canónicos: un resultado fundamental

Teorema

Sea $\{X_t\}_{t\in\mathbb{N}}$ *una cadena de Markov reversible, irreducible y aperiódica, tal que* $P[x,x] \geq \frac{1}{2}$ *para todo* $x \in \Omega$. *Además, sea* Γ *un conjunto de caminos canónicos para cada par de estados en* Ω.

Para cada $x \in \Omega$ *y* $\varepsilon \in (0,1)$, *se tiene que:*

$$\tau_x(\varepsilon) \quad \leq \quad \rho(\Gamma) \cdot \left(\ln \frac{1}{\vec{\pi}[x]} + \ln \frac{1}{\varepsilon} \right)$$

Un primer ejemplo: generación aleatoria de strings binarios

Consideremos el problema de generar con distribución uniforme strings binarios de largo $n \geq 1$ a través de una cadena de Markov.

- Usaremos el conjunto de estados $\Omega = \{0, 1\}^n$

Para movernos entre estados invertiremos a lo más un bit del string.

- Con probabilidad $\frac{1}{2}$ no nos movemos de estado
- Si nos movemos, se invierte uno de los n bits con distribución uniforme

Un primer ejemplo: generación aleatoria de strings binarios

Sea $\vec{\pi} : \Omega \to \mathbb{R}$ la distribución uniforme sobre los estados.

Suponiendo que $d : \Omega \times \Omega \to \mathbb{N}$ es la distancia de Hamming entre dos strings binarios, tenemos que:

$$P[x, y] = \begin{cases} \frac{1}{2} & \text{si } d(x, y) = 0 \\ \frac{1}{2 \cdot n} & \text{si } d(x, y) = 1 \\ 0 & \text{si } d(x, y) \geq 2 \end{cases}$$

De esto se concluye que $\vec{\pi}$ es la (única) distribución estacionaria de la cadena de Markov.

- ¿Por qué?

Un primer ejemplo: generación aleatoria de strings binarios

La cadena de Markov definida satisface las condiciones del teorema.

- La cadena es reversible, irreducible, aperiódica y $P[x, x] \geq \frac{1}{2}$ para cada $x \in \Omega$

Podemos usar entonces la técnica de los caminos canónicos para acotar el mixing time.

- La clave es escoger un conjunto Γ de caminos canónicos tal que $\rho(\Gamma)$ sea polinomial en n, que en este caso es el tamaño de la entrada

Utilizamos como primer ejemplo a la generación aleatoria de strings binarios porque es simple escoger un conjunto de caminos Γ tal que $\rho(\Gamma)$ es polinomial en n

Un primer ejemplo: generación aleatoria de strings binarios

Dado $x = x_1 \cdots x_n$ e $y = y_1 \cdots y_n$, consideramos el camino canónico $\gamma_{x,y} = e_1 \cdots e_n$, donde para $i \in \{1, \ldots, n\}$:

$$e_i \quad = \quad \left\{ y_1 \cdots y_{i-1} x_i x_{i+1} \cdots x_n,\ y_1 \cdots y_{i-1} y_i x_{i+1} \cdots x_n \right\}$$

Vale decir, el camino de x a y corresponde a cambiar uno a uno de izquierda a derecha los bits de x por los de y

- Si los strings tienen un bit en común, utilizamos un loop
- Nótese que $|\gamma_{xy}| = n$

Acotando el número de caminos

Necesitamos obtener una cota superior para la cantidad de caminos que pasan por cada arco.

- Consideramos un arco e arbitrario

Para algún $i \in \{1, \ldots, n\}$, tenemos que:

$$e = \left\{ y_1 \cdots y_{i-1} x_i x_{i+1} \cdots x_n,\ y_1 \cdots y_{i-1} y_i x_{i+1} \cdots x_n \right\}$$

Acotando el número de caminos

Dados $x, y \in \Omega$, si $e \in \gamma_{x,y}$, entonces:

- Los bits $x_i, \ldots, x_n$ de x están fijos, y tenemos 2^{i-1} opciones para los otros bits de x
- Los bits $y_1, \ldots, y_i$ de y están fijos, y tenemos 2^{n-i} opciones para los otros bits de y

Concluimos que la cantidad total de caminos que pueden contener a e es

$$2^{i-1} \cdot 2^{n-i} \quad = \quad 2^{n-1}$$

Acotando superiormente $\rho(\Gamma)$

Recordemos que la carga máxima de un arco es

$$\rho(\Gamma) \quad = \quad \max_{e \in E_{P,\vec{\pi}}} \left(\frac{1}{Q(e)} \sum_{\gamma_{x,y} \in \Gamma \,:\, e \in \gamma_{x,y}} \vec{\pi}[x] \cdot \vec{\pi}[y] \cdot |\gamma_{x,y}| \right)$$

Tenemos que:

- $\forall x \in \Omega : \vec{\pi}[x] = \frac{1}{2^n}$
- $\forall e \in E_{P,\vec{\pi}}$ tal que $e = \{x, y\} : Q(e) = P[x, y] \cdot \vec{\pi}[y] \geq \frac{1}{2 \cdot n} \cdot \frac{1}{2^n}$
- $\forall e \in E_{P,\vec{\pi}} : |\{\gamma_{x,y} \in \Gamma \mid e \in \gamma_{x,y}\}| = 2^{n-1}$
- $\forall x, y \in \Omega : |\gamma_{x,y}| = n$

Acotando superiormente $\rho(\Gamma)$

Desarrollando la expresión obtenemos:

$$
\begin{aligned}
\rho(\Gamma) &= \max_{e \in E_{P,\vec{\pi}}} \left(\frac{1}{Q(e)} \sum_{\gamma_{x,y} \in \Gamma \,:\, e \in \gamma_{x,y}} \vec{\pi}[x] \cdot \vec{\pi}[y] \cdot |\gamma_{x,y}| \right) \\
&\leq \max_{e \in E_{P,\vec{\pi}}} \left(2 \cdot n \cdot 2^n \sum_{\gamma_{x,y} \in \Gamma \,:\, e \in \gamma_{x,y}} \frac{1}{2^n} \cdot \frac{1}{2^n} \cdot n \right) \\
&= \max_{e \in E_{P,\vec{\pi}}} \left(2 \cdot n \cdot 2^n \cdot \frac{1}{2^n} \cdot \frac{1}{2^n} \cdot n \cdot 2^{n-1} \right) \\
&= \max_{e \in E_{P,\vec{\pi}}} n^2 \\
&= n^2
\end{aligned}
$$

Acotando el mixing time en la generación de strings binarios

Recordemos que para $x \in \Omega$ y $\varepsilon \in (0, 1)$:

$$\tau_x(\varepsilon) \quad \leq \quad \rho(\Gamma) \cdot \left(\ln \frac{1}{\vec{\pi}[x]} + \ln \frac{1}{\varepsilon} \right)$$

Por lo tanto, tenemos que:

$$\tau_x(\varepsilon) \ \leq \ n^2 \cdot \left(\ln \frac{1}{\vec{\pi}[x]} + \ln \frac{1}{\varepsilon} \right) \ =$$

$$n^2 \cdot \left(\ln 2^n + \ln \frac{1}{\varepsilon} \right) \ = \ n^2 \cdot \left(n \cdot \ln 2 + \ln \frac{1}{\varepsilon} \right)$$

Concluimos que la cadena converge rápido desde cualquier estado inicial.

Un FPAUG para strings binarios

Ejercicio

Utilizando el resultado anterior construya un FPAUG para el problema de generar string binarios con distribución uniforme.

- Su algoritmo debe realizar a lo más la siguiente cantidad de pasos en la cadena de Markov usada:

$$\left\lceil n^2 \cdot \left(n \cdot \ln 2 + \ln \frac{2}{\varepsilon} \right) \right\rceil$$