EXACTAS UBA

INGENIERIA INDUSTRIAL UNIVERSIDAD DE CHILE

Optimización en línea con información estocástica y las Desigualdades del Profeta

Profesor: José Correa
Auxiliares: Javier Marinkovic y Matías Ortiz

Tarea 2

15 de Octubre de 2024
Entrega: 30 de Octubre de 2024

P1. Sea $G = (V, E)$ un grafo, donde cada arista e tendrá un peso aleatorio "Bernoulli amplificado", X_e, es decir, X_e vale c_e con probabilidad p_e, y 0 con probabilidad $1 - p_e$.

El objetivo de esta pregunta será desarrollar un algoritmo online para el problema de encontrar un matching de peso máximo. En clases demostramos que usando thresholds fijos se puede diseñar un algoritmo que entrega una desigualdad de profeta con factor $1/3$, es decir, que el peso esperado del matching que entrega el algoritmo es al menos

$$\frac{1}{3} \cdot \mathbb{E}\left[\max_{M \text{ matching en } G} \sum_{e \in M} X_e\right] = \frac{1}{3} \cdot \mathbb{E}[OPT].$$

En este problema encontraremos un algoritmo alternativo para este resultado, usando *Online Contention Resolution Schemes (OCRS)*. Para ello, procederemos por los siguientes pasos:

a) Demuestre que la esperanza del peso del matching óptimo obtenido por un profeta, $\mathbb{E}[OPT]$, esta acotado superiormente por la solución del siguiente problema lineal, que llamamos relajación "Ex-Ante":

$$\begin{aligned} \max \sum_e y_e c_e & \\ \text{s.a.} \sum_{e: u \in e} y_e \leq 1 & \quad \forall u \in V \\ 0 \leq y_e \leq p_e & \quad \forall e \in E \end{aligned} \qquad \text{[ExA]}$$

Caracterizamos $OPT = \sum_e X_e \mathbf{1}_{e \in OPT}$. Por linealidad de la esperanza, tenemos:

$$\mathbb{E}[OPT] = \sum_e \mathbb{E}[X_e \mathbf{1}_{e \in OPT}]$$

Aplicando esperanzas iteradas, vemos que:

$$\begin{aligned}\mathbb{E}[X_e \mathbf{1}_{e\in OPT}] &= \mathbb{E}[X_e \mathbf{1}_{e\in OPT} | X_e = c_e]p_e + \mathbb{E}[X_e \mathbf{1}_{e\in OPT} | X_e = 0](1-p_e) \\ &= c_e \mathbb{E}[\mathbf{1}_{e\in OPT} | X_e = c_e]p_e = c_e \mathbb{P}[e \in OPT | X_e = c_e]p_e \\ &= c_e \mathbb{P}[e \in OPT]\end{aligned}$$

Por comodidad, definimos $q_e = \mathbb{P}[e \in OPT]$. Con esto obtenemos que $\mathbb{E}[OPT] = \sum_e q_e c_e$, y para concluir que $q = (q_e)_{e\in E}$ es una solución factible de "Ex-Ante". Notamos que:

$$\begin{aligned} &\forall u \in V \qquad && \sum_{e:u\in e} q_e = \mathbb{P}[u \text{ esta emparejado en } OPT] \leq 1 \\ &\forall e \in E && 0 \leq q_e = \mathbb{P}[e \in OPT | X_e = c_e]p_e \leq p_e \end{aligned}$$

Donde la primera igualdad sale de que si dos aristas comparten un nodo u, los eventos de que pertenezcan a OPT son excluyentes; y la segunda del desarrollo previo. Con esto podemos concluir que q es solución factible de "Ex-Ante", y por lo tanto, que $\mathbb{E}[OPT] \leq [ExA]$.

■

b) Suponga ahora que las aristas llegan en el orden $e_1, ..., e_m$ y que al llegar un arco su peso X_e se revela. Consideremos el algoritmo ALG que al llegar una arista $e_i = e = (u, v)$ con valor c_e, chequea que u y v estén disponibles y en tal caso agrega a e en el matching con probabilidad $\beta_e = \text{mín}\{1, \frac{y_e}{3p_e\alpha_e}\}$, donde $\alpha_e = \Pr[u, v \text{ estén disponibles al llegar } e]$.

En principio, podríamos pensar que ALG está mal definido ya que la ejecición de ALG depende del mismo algorimo a traves de la definición de α_e. Pruebe que lo primero no ocurre, es decir pruebe que ALG está bien definido.

A priori no es obvio que el algoritmo esté bien definido, pues los valores α_e definen el funcionamiento del algoritmo pero el algoritmo define los valores α_e. Podemos dar el siguiente argumento de por que no es el caso:

Primero, $\alpha_{e_1} = 1$ determinísticamente. Luego, α_{e_2} solo depende potencialmente de lo que haya pasado con e_1, si es que e_1 comparte algún vertice con e_2. Es decir, conociendo α_{e_1} y β_{e_1}, podemos calcular α_{e_2}.

Podemos repetir esta lógica para argumentar que conociendo $\alpha_{e_1} \ldots \alpha_{e_i}$, podemos calcular $\alpha_{e_{i+1}}$ y entonces estos valores quedan bien definidos.

■

c) Pruebe ahora que, para todo $e \in E$, $\Pr[e \text{ sea seleccionado por ALG}] \leq y_e/3$.

El algoritmo selecciona e con probabilidad β_e si es que están disponibles sus vértices u, v en el momento que llega, y adicionalmente si $X_e = c_e$. Es decir, podemos calcular:

$$\mathbb{P}[e \text{ sea seleccionado por ALG}] = \alpha_e \beta_e p_e$$

Y luego, vemos que:

$$\alpha_e \beta_e p_e \leq \text{mín}\{\alpha_e p_e, \frac{y_e}{3}\} \leq \frac{y_e}{3}$$

■

d) Usando lo anterior pruebe que $0 \leq \frac{y_e}{3p_e\alpha_e} \leq 1$ (Para esto demuestre que $\alpha_e \geq 1/3$). En conclusion el mínimo en la definición de β no es necesario.

Estudiemos la probabilidad complementaria a α_e:

$$\begin{aligned} 1 - \alpha_e &= \mathbb{P}[\text{alguno de } u, v \text{ no está disponibles al llegar } e] \\ &\leq \mathbb{P}[u \text{ no disponible}] + \mathbb{P}[v \text{ no disponible}] \end{aligned}$$

Analicemos estas probabilidades. Definamos las aristas conflictivas para e respecto al vertice u $C(e,u)$ como:

$$C(e,u) = \{e^{'} \in E | e^{'} \text{ llega antes que } e \wedge u \in e^{'}\}$$

Luego:

$$\mathbb{P}[u \text{ no disponible}] = \mathbb{P}[\cup_{e' \in C(e,u)} e^{'} \in ALG] \leq \sum_{e' \in C(e,u)} \mathbb{P}[e^{'} \in ALG]$$

$$\leq \sum_{e' \in C(e,u)} \frac{y_{e'}}{3} \leq \frac{1}{3}$$

Donde la última cota sale a partir de la primera restricción del problema “Ex-Ante”. Obtenemos entonces que:

$$1 - \alpha_e \leq \frac{1}{3} + \frac{1}{3} = \frac{2}{3} \Rightarrow \alpha_e \geq \frac{1}{3}$$

Concluímos viendo que $\frac{y_e}{3p_e\alpha_e} \leq \frac{y_e}{p_e} \leq 1$, donde utilizamos el hecho de que $y_e \leq p_e$. ■

e) Demuestre finalmente que, para todo $e \in E$, Pr[e sea seleccionado por ALG] $\geq y_e/3$, y concluya que $\mathbb{E}[ALG] \geq [ExA]/3 \geq \mathbb{E}[OPT]/3$.

A partir del punto anterior, sabemos que $\beta_e = \frac{y_e}{3p_e\alpha_e}$. Reutilizando la expresión de c), $\mathbb{P}[e \text{ sea seleccionado por } ALG] = \alpha_e\beta_e p_e = \frac{y_e}{3}$. Para continuar, caracterizamos $ALG = \sum_e X_e \mathbf{1}_{e \in ALG}$. Utilizando linealidad de la esperanza:

$$\mathbb{E}[ALG] = \sum_e \mathbb{E}[X_e \mathbf{1}_{e \in ALG}]$$

Aplicando el mismo procedimiento con esperanzas iteradas de a), obtenemos que $\mathbb{E}[X_e \mathbf{1}_{e \in ALG}] = c_e \mathbb{P}[e \in ALG]$. Luego:

$$\mathbb{E}[ALG] = \sum_e c_e \mathbb{P}[e \in ALG] = \sum_e c_e \frac{y_e}{3} = [ExA]/3 \geq \mathbb{E}[OPT]/3.$$

■

f) Porqué no podemos mejorar este algorimo de forma inmediata? Qué pasaría si ajustamos la probabilidad de aceptar una arista a $\frac{y_e}{2{,}99 \cdot p_e \alpha_e}$?

Imaginemos alteramos el algoritmo, de forma que seleccione aristas cuando pueda con probabilidad $\beta = \frac{y_e}{2{,}99 p_e \alpha_e}$. En este caso, si repetimos el procedimiento de la parte d), obtenemos que $\alpha_e \geq \frac{0{,}99}{2{,}99}$, y luego que:

$$\frac{y_e}{2{,}99\alpha_e p_e} \leq \frac{1}{0{,}99}$$

Por lo que ya no podemos argumentar que $\beta_e = \frac{y_e}{2{,}99\alpha_e p_e}$ siempre. Esto significa que tampoco podemos utilizar el análisis de la parte f) para asegurar una cota, y perdemos la garantía de aproximación. ■

Bono (Este item no tiene puntaje base, pero le puede entregar puntaje adicional) ¿Podemos generalizar este algoritmo para variables aleatorias X_e generales? Comente como podría lograrse eso.

P2. Considere un grafo $G = (V, E)$ con pesos aleatorios independientes en las aristas, dados por X_e (una por cada arista $e \in E$). Las aristas llegan una a una y revelan su peso. En esta pregunta se intentará resolver el problema de encontrar de manera *online* un conjunto de aristas que resulte ser acíclico y de peso máximo. De ahora en adelante denotaremos $\mathcal{I}$ como al conjunto de todas las aristas del grafo que resulten ser un subgrafo acíclico.

Consideremos $X = (X_e)_{e \in E}$ y definimos $OPT(X)$ como la variable aleatoria que dadas las realizaciones de X entrega el árbol (subgrafo acíclico de G) de peso máximo. Del mismo modo, dado un algoritmo online, ALG, definimos $ALG(X)$ como la variable aleatoria que entreg ael output del algoritmo bajo las realizaciones de X.

a) Definamos $p_e = \mathbb{P}(e \in OPT)$ y $t_e = \mathbb{E}[X_e | X_e \geq F^{-1}(1 - p_e)]$. Pruebe que $t_e \geq \mathbb{E}[X_e | H]$ para cualquier evento H de probabilidad p_e

b) Consideremos siguiente polítopo $\mathcal{P}_E = \{p \in \mathbb{R}^E_{\geq 0} | \forall S \in \mathcal{P}(E), \sum_{i \in S} p_i \leq r(S)\}$, donde r es la función rango que se define como $r(S) = \max_{I \in \mathcal{I}}\{|I| : I \subseteq S\}$. Pruebe que $\mathcal{P}_E$ corresponde a las conbinaciones convexas de vectores de incidencia de grafos acíclicos y concluya que $p \in \mathcal{P}_E$.

c) Ahora haremos una reducción del problema. Para cada X_e (una por cada arista $e \in E$), consideremos su símil X'_e variable aleatoria definida como sigue:

$$X'_e = \begin{cases} t_e & \text{con probabilidad } p_e \\ 0 & \text{con probabilidad } 1 - p_e \end{cases}$$

Demuestre que:

$$OPT(X) \leq \sum_{e \in E} p_e t_e$$

d) Demustre también que para todo algoritmo online, ALG, se tiene que

$$ALG(X) \geq ALG(X').$$

Concluya que $ALG(X)/OPT(X) \geq ALG(X')/OPT(X')$, por lo que si podemos demostrar una desigualdad del prophet para X' también se satisface para X.

e) Sea $q \in \frac{1}{4}\mathcal{P}_G$, pruebe que existe una forma de orientar las aristas de E de forma que en cada vértice v la masa total de probabilidad de las aristas entrantes a v es a lo más $\frac{1}{2}$.

f) Sea $q \in \frac{1}{4}\mathcal{P}_G$ y definamos $R_q(S)$ el conjunto aleatorio que contiene a los elementos $e \in S$ con probabilidad q_e. Para e, f arcos (aristas luego de dirigir el grafo), diremos que se enfrentan si apuntan al mismo vértice. Denotaremos como $b_e(S)$ a la probabilidad de que e se enfrente a un arco de $R_q(S \setminus \{i\})$. Considere la orientación de las aristas del inciso anterior, $e \in E$ una arista cualquiera y v el vértice tal que $e \in \delta^-(v)$ (es decir que e apunta a v). Pruebe que $\forall S \subseteq E$

$$b_e(S \setminus \delta^+(v)) \leq \frac{1}{2}$$

g) Considere $q \in \frac{1}{4}\mathcal{P}_G$ y dirija las aristas como en la parte (e). Sea $A \subseteq V$ subconjunto aleatorio de vértices, tal que cada vértice es incluido en A independientemente con probabilidad $1/2$ y sea $B = V \setminus A$. Sea $S = \{e \in E | e \in \delta^+(u) \cap \delta^-(v), u \in A, v \in B\}$, donde $\delta^+(u)$ es el conjunto de aristas salientes de u y $\delta^-(u)$ las entrantes. Pruebe que

$$\mathbb{E}_S\left[\sum_{e \in S} q_e t_e (1 - b_e(S))\right] \geq \frac{1}{8} \sum_{e \in E} q_e t_e$$

h) Pruebe el siguiente algoritmo es una 32−aproximación, es decir :

$$\mathbb{E}[ALG(X)] \geq \frac{1}{32} OPT(X)$$

Algoritmo 1 Algoritmo para grafos con pesos en aristas

Input: Un grafo y variables aleatorias X_e
Output: Conjunto de aristas Acíclico

Calcular p y t como en el inciso c). Definir $q_e = p_e/4$ para todo $e \in E$.
Dirigir el grafo como en el inciso e)
Escoger un corte dirigido S uniformemente aleatorio como en g)
for $e \in E$ **do**
  if $e \in S$ **then** $T_e = t_e$
  if $e \notin S$ **then** $T_e = \infty$
end for
Observar las realizaciones y aceptar las que superen su threshold
return $ALG(X)$
