13.6.04

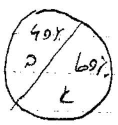
1

סבוכיות - שיעור אחרון

גליה קמפ

<u>חסם צ'רנוף</u> - (דמיין מדינה בו מתמודדים בבחירות בין 2 מועמדים לנשיאות. רוצים לבצע סקר, ורוצים לדעת תוצאות בדיוק של 1%. אולי גם (שאלה) את כולם, ויתכן וטעינו. לכן, אנחנו יכולים רק לבקש דיוק של $1\% \pm$ עם ודאות מסויימת (למשל 90%). את הדיוק נסמן ב-δ ואת הודאות ב $(1-\alpha)$.

דוגמא



$X_i=1$ (ג) with 60%

$X_i=0$ (ב) with 40%

$E[X]=0.6$

<u>משפט צ'רנוף</u> - עבור מ"מ $X_1,\ldots,X_k\in[0,1]$ ב"ת (האנשים שאנו שואלים), נסמן $X=\frac{1}{k}\sum_{i=1}^{k}X_i$ (הממוצע).

ע"פ משפט צ'רנוף:

$$\Pr\left[\left|\frac{1}{k}\sum X_i-E[X]\right|>\delta\right]\le 2\cdot e^{-2\delta^2 k}$$

כלומר, הסיכוי שההפרש יהיה גדול מ-δ הוא בסיכוי הקטן מ- $2\cdot e^{-2\delta^2 k}$ כלומר, כדי לקיים ודאות של $1-\alpha$ נרצה כי $2e^{-2\delta^2 k}=\varepsilon$ $\Leftarrow$ $\ln 2\cdot e^{-2\delta^2 k}=\ln\varepsilon$

$\Leftarrow$ $O(-2\delta^2 k)=\ln\varepsilon$ $\Leftarrow$ $O(2\delta^2 k)=\ln\frac{1}{\varepsilon}$ $\Leftarrow$ $k=O(\ln 1/\varepsilon)\cdot\frac{1}{\delta^2}$.

כלומר, בדוגמה שראינו עלינו לשאול 10,000 אנשים, נשים לב שהכמות אינה תלויה בגודל האוכלוסיה ולכן המדינה יכולה להיות גדולה/קטנה יותר ועדיין עבור אותו דיוק וודאות נצטרך אותה כמות של אנשים.

היום נראה 2 אלגוריתמים באמצעות LP ו- SDP ועיגול, ההבדל מעיגול שראינו בעבר הוא בכך שהעיגול כאן יהיה רנדומי.

נתחיל עם <u>Set-Cover</u>:

<u>קלט:</u> $C=\{S_1,S_2,\ldots,S_k\}$ כאשר $S_i\subseteq\{1,\ldots,n\}$

<u>פלט:</u> $S=\{S_{i_1},\ldots,S_{i_L}\}$ כך ש $\forall i\in\{1,\ldots,n\}.\exists S_j\in S.\ i\in S_j$ ורוצים למזער את S בגודל המינימלי! (כלומר מספר הקבוצות הקטן ביותר שאפשר)

<u>אלגוריתם:</u> לכל קבוצה S_i: נגדיר משתנה x_i כך ש $x_i=1$ אמ"מ $S_i\in S$ ו $x_i=0$ אמ"מ $S_i\notin S$.

אנחנו ננסה להביא למינימום את הסכום של x_i כך שכל i יהיה בקבוצה שאליה היא שייך מבין הקבוצות ב-S

<u>פורמלית:</u>

$$\begin{cases}\min\sum x_i \quad s.t.\ \forall j\in\{1,\ldots,n\}.\ \sum_{i:j\in S_i}x_i\ge 1\\ x_j\in\{0,1\}\end{cases}$$

זו בעיית Integer Programin (IP) אין אלג' יעיל הנ"פ שלמה.

כדי להפוך את הבעיה לבעיית LP (Linear Programing) עלינו לשנות את הדרישה $x_j\in\{0,1\}$ כיוון שזה אינו בתחום לינארי. לכן נשנה את הדרישה לכך $0\le x_j\le 1$. ל-LP יש אלגוריתמים יעילים הפותרים את הבעיה.

נסמן את הפתרון של האלג' הלינארי ב- $(P_1,P_2,\ldots,P_k)$ עם $OPT=\sum_{i=1}^{k}P_i\le$ Set-Cover.

15.6.07
2

סבוכיות - שיעור אחרון

אנחנו רוצים לבצע עיגול של הערכים כדי למצוא Set-Cover טוב לבעיה.

כלומר ניקח את הפתרון $(p_1,\dots,p_k)$ ונהפוך אותו ל Set-Cover עם t קבוצות. זה אולי לא ה SC האופטימלי (מיני) אבל אנחנו רוצים שיתקיים כי $SC \le t \le O(\log n)\cdot SC$ (כאשר SC הוא הכיסוי האופטימלי).

<u>אלגוריתם העיגול:</u>

אנחנו מבצעים מספר "סיבובים":

- עבור כל i ניקח את הקבוצה S_i בהסתברות p_i (זה) (כאן מרכיב הרנדומיות).

נסתכל על התוחלת של גודל ה- SC, היא $E[\#\text{Set-cover}]=\sum_{i=1}^{k} p_i = OPT$.

אנו מבצעים את $d\cdot\log n$ (d-קבוע) ונכניס ל S יהיה איחוד של כל הקבוצות שהחלטנו לקחת בכל הסיבובים.

נשים לב שעבור תוחלת מספר הקבוצות ב-S מתקיים $E[\#\text{set in } S] \le d\cdot\log\cdot OPT$.
(הסיבה שזה $\le$ היא כיוון שייתכן ובחרנו את אותה הקבוצה פעמיים).
נניח כי האיבר j נמצא ב- f קבוצות $S_{i_1},\dots,S_{i_f}$.

מה הסיכוי ש- j לא יהיה בבחירה (לאחר סיבוב אחד)?

$$Pr[j \text{ not covered in } \underline{\text{one round}}] = \prod_{l=1}^{f}(1-p_{i_l}) \underset{\uparrow \; 1-x\le e^{-x}}{\le} \prod_{l=1}^{f} e^{-p_{i_l}} \underset{\uparrow \text{ הנחה}}{=} e^{-\sum_{l=1}^{f} p_{i_l}}$$

$$= e^{-\sum_{l: j\in S_l} p_{i_l}} \le \frac{1}{e} \rightarrow \text{וזהו קבוע}$$

מכאן, $\forall j.\ Pr[j \text{ covered in 1 round}] \ge 1-\frac{1}{e} \approx 60\%$. נבחר d כך שיתקיים $\left(\frac{1}{e}\right)^{d\log n} \le \frac{1}{4n}$

Choose d s.t. $\left(\frac{1}{e}\right)^{d\log n} \le \frac{1}{4n}$

ונעבור עתה לסיכוי ש-j לא יכוסה <u>בכל</u> הסיבובים:

$$\forall j\in\{1,\dots,n\}.\ Pr[j \text{ not in all rounds}] \le \left(\frac{1}{e}\right)^{d\log n} \le \frac{1}{4n}$$

עבור כל האיברים:

<u>Union bound</u>: $Pr[\exists j\in\{1,\dots,n\}: j \text{ not coverd}] \le \sum_{j=1}^{n}\frac{1}{4n} = \frac{1}{4}$

מכאן, כי יש סיכוי $\ge \frac{3}{4}$ כך ש- S שקיבלנו הוא Set-Cover.

<u>SDP</u>

ה- LP השתמשנו בתבנית:

$$\min_{x_1,\dots,x_n} \sum_j A_j\cdot x_j$$

$$\text{s.t. } \sum_{j=1}^{n} B_j^i\cdot x_j$$

$$\vdots$$

$$x_1\ge 0,\ x_2\ge 0,\dots$$

סבוכיות - שיעור אחרון

תזכורת

$v\cdot w = \sum_{i=1}^{n} v_i \cdot w_i$

$\|v\| = \sqrt{v\cdot v}$

SDP היא צורה כללית. במחלקה זו אנו מדברים על וקטורים $v_1,\dots,v_n$ מעל $\mathbb{R}^n$.

$$v_1,\dots,v_n \in \mathbb{R}^n$$

$$\min \sum_{i,j=1}^{n} A_{i,j}\cdot v_i \cdot v_j$$

$$\text{s.t.} \quad \sum_{i,j=1}^{n} B^1_{i,j}\cdot v_i\cdot v_j \ \overset{=}{\geq}\ b^1$$

$$\vdots$$

$$\sum_{i,j=1}^{n} B^m_{i,j}\cdot v_i\cdot v_j \geq b^m$$

SDP זה ר"ת של Semidefinite Programing!

קיימים אלגוריתמים יעילים הפותרים SDP (אפשר לעשות זאת אפילו עם האלג' של LP).

אלגוריתמים אלו גם נותנים לנו את $v_1,\dots,v_n$.

דוגמא:

$$\text{Ex:}\quad B^1_{ij} = \begin{cases} 1 & i=j=1 \\ 0 & (i,j)\neq(1,1)\end{cases}$$

$$\sum_{i,j}^{n} B^1_{ij}\cdot v_i\cdot v_j = B_{11}\, v_1\cdot v_1 = \|v_1\|^2$$

בעיית MAX-CUT

זו בעיה NP-שלמה ואנו רוצים למצוא לה קירוב.

הקלט: גרף $G=(V,E)$ לא מכוון.

פלט: תת קבוצה $S\subseteq V$ כך ש-$|E(S,\bar{S})|$ יהיה מקסימלי.

קיים אלגוריתם מפורסם של SDP הנותן קירוב לבעיה עם פקטור של $0.878\ldots$ כלומר, האלגוריתם מחזיר S' כך ש-$|E(S',\bar{S'})| \geq 0.878\ldots|E(S,\bar{S})|$.

אלגוריתם:

נגדיר משתני אינדקציה x_i לכל צומת בגרף שיקבלו 1 אם צומת i ב-S ו-(-1) אחרת.

$$\text{Variables: } \forall i\in V.\ x_i \quad \begin{pmatrix} x_i = 1 \Leftrightarrow i\in S \\ x_i=-1 \Leftrightarrow i\notin S\end{pmatrix}$$

$$i,j\in V \quad x_i\cdot x_j = 1 \rightarrow i,j\in S \text{ or } i,j\in\bar{S} \rightarrow (i,j)\notin E(S,\bar{S})$$

$$x_i\cdot x_j = -1 \rightarrow i\in S \,\&\, j\notin S \text{ or } i\notin S \,\&\, j\in S \rightarrow (i,j)\in E(S,\bar{S})$$

נרצה לקבל 1 כשהקשת בחתך ו-0 אחרת, לכן נסתכל על הביטוי הבא:

$$\frac{1-x_i\cdot x_j}{2} = \begin{cases} 1 & \text{if } (i,j)\in E(S,\bar{S}) \\ 0 & \text{if } (i,j)\notin E(S,\bar{S})\end{cases}$$

$$\max \sum_{(i,j)\in E} \frac{1-x_i\cdot x_j}{2} \quad \text{s.t.} \quad x_i\in\{-1,1\}\ \forall i\in V$$

שוב קיבלנו IP. לכן, נהפוך את הבעיה לבעיית SDP:

$$v_1,\dots,v_n\in\mathbb{R}^n$$

$$\max \sum_{(i,j)\in E} \frac{1-v_i\cdot v_j}{2} \quad \text{s.t.} \quad i\in V \quad v_i\cdot v_i = 1 \Leftrightarrow \|v_i\|=1$$

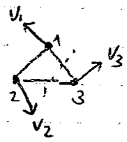
$$\max \sum_{(i,j)\in E}\frac{1}{2} - \sum_{(i,j)}\frac{v_i\cdot v_j}{2} = \frac{|E|}{2} - \min\sum_{(i,j)}\frac{v_i\cdot v_j}{2}$$

15.6.09

4

סבוכיות - שיעור אחרון

<u>דוגמא:</u> K_3



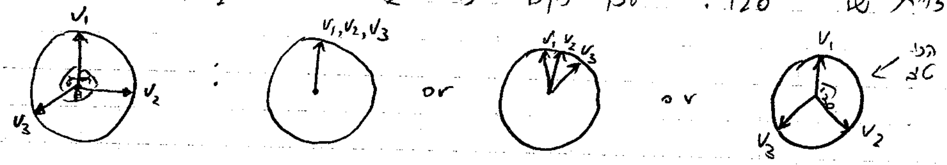
$(MAX\text{-}CUT = 2)$

לכל קודקוד התאמנו וקטור, כדי למקסם את הסכום שלנו להביא למינימום את סכום מכפלות הוקטורים. לכן נרצה שיהיה כמה שיותר זוגות של וקטורים עם כיוון הפוך.

בדוגמא שלנו:

$$\max \frac{1}{2}(1-V_1\cdot V_2+1-V_2\cdot V_3+1-V_1\cdot V_3)=\frac{3}{2}-\min\left(\frac{V_1\cdot V_2+V_2\cdot V_3+V_3\cdot V_1}{2}\right) =$$

כלומר מבין האפשרויות השונות הכי טוב הוא כאשר בין 2 הוקטורים יש זווית של $120°$. לכן נקבל כי $V_1\cdot V_2=\cos 120=-\frac{1}{2}$.



$$= \frac{3}{2}+\frac{\frac{1}{2}+\frac{1}{2}+\frac{1}{2}}{2}=\frac{6}{4}+\frac{3}{4}=\frac{9}{4}\ge 2$$

<u>אלגוריתם:</u> האלגוריתם ע"ש goemans-Williamson '94 ונקרא Hyperplane rounding

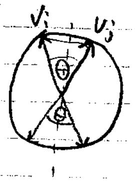
$$\text{Solve SDP} \rightarrow v_1,\ldots,v_n \qquad OPT=\sum_{(i,j)\in E}\frac{1-v_i\cdot v_j}{2}$$

אפשר לחשוב על תוצאת SDP כוקטורים בספרה. אם יש בין 2 צמתים קשת, קיים סיכוי גדול שהם רחוקים אחד מהשני (כדי למקסם באמצעות SDP). לכן, ניתן לחתוך את הספרה בצורה רנדומאלית. הסיכוי הוא של 2 צמתים עם קשת יהיו ב-2 קבוצות שונות (כדי הם נראה רחוקים אחד מהשני).

כדי לבצע את החתך אנו נעזרים ב Hyper plane (על-מישור). הכוונה היא למישור ממימד $n-1$ בתוך מרחב ממימד n.

ניתן לתאר מישור באמצעות הנורמל של המישור (כל הוקטורים על המישור הם במכפלה פנימית איתו של 0) $Plane=\{v: n\cdot v=0\}$

לכן, כדי לחתוך את המרחב, נבחר וקטור רנדומלי מהספרה ונגדיר אותו להיות הנורמל של המישור שלנו (נסמן את הנורמל ב- n. $\|n\|=1$, $S=\{i: v_i\cdot n\ge 0\}$



מה הסיכוי שקשת (i,j) תחתך בחתך הרנדומלי?

$$Pr[(i,j)\in E \text{ is cut}]=\frac{2\theta}{2\pi}=\frac{\theta}{\pi}$$

<u>טענה:</u> $Pr[(i,j) \text{ is cut}]\ge 0.878\ldots\frac{1-v_i\cdot v_j}{2}$.

<u>מסקנה:</u>

$$E[|E(S,\bar{S})|]=\sum_{(i,j)\in E}Pr[(i,j)\in E \text{ is cut}]\ge 0.878\ldots\underbrace{\sum_{(i,j)\in E}\frac{1-v_i\cdot v_j}{2}}_{OPT}\ge 0.878\ldots MAX\text{-}CUT$$

15.6.09

5

סבוכיות - שיעור אחרון

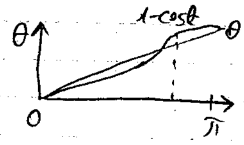
הוכחת הטענה:

$$\frac{\Pr[(i,j) \in \text{is cut}]}{1-\frac{v_i \cdot v_j}{2}} = \frac{\frac{\theta}{\pi}}{1-\frac{v_i \cdot v_j}{2}} \underset{v_i \cdot v_j = \cos\theta}{=} \frac{\frac{\theta}{\pi}}{1-\frac{\cos\theta}{2}} = \frac{2}{\pi} \cdot \frac{\theta}{1-\cos\theta} \geq 0.878\ldots$$

אי-שוויון

לפתור אנליטית, מחשב וכו' את החסם התחתון לערך.

$$\min \frac{2}{\pi} \frac{\theta}{1-\cos\theta} = 0.878\ldots$$



זהו אלגוריתם מקרב מפורסם ומשתמשים בזה הרבה מאוד.