12.1.04 (1)

פרוייקט תוכנה - שיעור 11

אלגוריתם הרשתות של ניומן

כתב אלגוריתם שמטרתו לחלק את הרשת לקבוצות כך שערך Q של החלוקה יהיה מקסימלי. החלוקה הטריוויאלית (כל הגרף בתוך קבוצה) הוא בעל ערך Q 0. זהו חסם תחתון, כלומר, במקרה הגרוע נשתמש בחלוקה הטריוויאלית.

מימוש האלגוריתם:

A - היא מטריצת השכנויות (איבר בה יש 1 אם בין שני הצמתים יש קשת).

M - כמות סכום הקשתות. מחושב על ידי סכימת כל ה-1 במטריצה.

E_{ij} - מספר הקשתות שאנו מצפים בין i ו-j (נמצא בין 0 ל-1).

<u>למה:</u> $E_{ij} \approx k_i * k_j / M$

כלומר, כדי לחשב את Q נשתמש בנוסחה $Q = \sum_{\substack{ij \\ \text{באותה} \\ \text{קבוצה}}} (A_{ij} - k_i * k_j / M)$

בשלב הראשון, נעסוק בבעיה המצומצמת של חלוקת הרשת ל-2 קבוצות בלבד עם Q מקסימלי.

נייצג את החלוקה באמצעות וקטור S. בוקטור נציין 1 עבור איברים שבקבוצה הראשונה ו-(-1) בקבוצה השנייה.

לכן, הנוסחה $\frac{1}{2}(s_i * s_j + 1)$ תהיה 1 כשהם באותה קבוצה ו-0 אחרת.
לכן, כדי לחשב את Q נכפיל את הנוסחה הקודמת בביטוי הזה וכך נוכל לוודא שאנו סוכמים רק קשתות בין 2 קודקודים שבאותה קבוצה

$$Q = \sum (A_{ij} - k_i * k_j / M) \left[\frac{1}{2}(s_i * s_j + 1)\right]$$

<u>הוכחת האלגוריתם</u> (לא חשוב בשביל הפרוייקט)

B מטריצת סכום כל שורה הוא 0, לכן היא ניתנת ללכסון. לכן יש לה בסיס אורתונורמלי $u_1, \dots, u_n$ כך שאם נכפיל את המטריצה באחד מהם, נקבל את u_i מוכפל בקבוע כלשהו, β_i.

כל ערכי β הם ערכים ממשיים ולכן נוכל לסדר אותם (ממגדל לקטן) כך ש-β_1 הוא הערך הגדול ביותר.

לכן, נוכל לחשב את Q כך:

$$Q = \frac{1}{2} s^T B s = \frac{1}{2} \sum_i \beta_i a_i^2 \quad \text{where} \quad a_i = u_i^T \cdot s \quad (s = \sum_i a_i u_i)$$

המסקנה היא שנרצה לדאוג שה-a_i הגדול ביותר יוכפל ב-β_1 כי הוא הכי גדול.
הבעיה היא ש-u_1 של B הוא לאו דווקא וקטור של 1 ו-(-1). מציאת הוקטור הזה היא בעיה NP-קשה.

הפיתרון שמצא ניומן הוא שימוש באלגוריתם היוריסטי. כלומר אלגוריתם שעובד טוב ברוב המקרים ומוצא את הפתרון האופטימלי או הקרוב לאופטימלי.

כך, כדי להפוך את u_1 לוקטור של 1 ו-(1-), נימן החיובי שעבור כל ערך הגדול מ-0, נעגל ל-1, ונעבור כל ערך שלילי או 0 (נעגל ל-(1-)).

<u>האלגוריתם המלא</u> (אלגוריתם יוריסטי)

<u>קלט:</u> רשת עם n צמתים $(n>1)$

1. קיבלנו רשת ← נחשב מטריצה B
2. מחשבים זוג של וקטור עצמי וע"ע.
3. ל-β_1 יש חסם תחתון 0. סכום כל שורה. לכן הוקטור $(1,\dots,1)$ מתאים לע"ע 0. במקרה הכי גרוע 0 הוא הע"ע הגדול ביותר.
אם עשרה ה-β הן 0 או שליליים, אין פיתרון וזו תשובה מדוייקת.

אם β חיובי, נחשב את s ונבדוק אם שיפרנו. לא בטוח שהוא אופטימלי ויכול להיות שניתן לשפר עוד את Q לפעמים.

נסתכל על דוגמה כזו $n=4$: $\vec{u}_1=(\frac{1}{\sqrt{2}}, 0, -\frac{1}{\sqrt{2}}, 0) \Rightarrow \vec{s}=(1,-1,-1,-1)$

<u>כיצד להרחיב את החלוקה ליותר מ-2 קבוצות?</u>
נפעיל את האלגוריתם בדרך רקורסיבית על כל אחת מהקבוצות שנוצרה.
אנו נמשיך לעצור אם שהחלוקה הנוספת ל-2 תגדיל את Q (ולא תשאיר אותו עם אותו ערך או אפילו עם ערך נמוך יותר).

כדי לחשב את הערך החדש של Q, נוכל לחסר את התרומה של הקבוצה המקורית ל-Q ולהוסיף תרומה של כל אחת מתתי הקבוצות. לערך זה נקרא ΔQ (ההפרש בין התרומה החדשה והישנה).

מדוע $B^{(g)}$ עבור קבוצה g.

1. נגדיר תת מטריצה ל-B שתכלול רק שורות שקשורות ב-g.
2. נסכום כל אחת מהשורות לוקטור $f^{(g)}$.
3. נחסיר את $f^{(g)}$ מהאלכסון. המטריצה החדשה נקראת המטריצה המוכללת.

נשים לב שאם g כולל את כל הצמתים, תת המטריצה הייתה B עצמה כיוון שסכום כל שורה הוא 0.

על המטריצה שקיבלנו נוכל להריץ את האלגוריתם שקודם הפעלנו על המטריצה המקורית.

כדי לשפר את היוריסטיקה, לאחר שמצאנו את s ננסה לשפר זאת באמצעות היוריסטיקה ידועה אחרת.

קלט: קבוצה המחולקת ל-g_1 ו-g_2 כחלוקה ראשונית.

בכל שלב הוא בוחר צומת רק שאם נעביר אותו מקבוצה לשנייה יגדל Q מקסימלי.
כל צומת מעבירים אותו פעם אחת בדיוק. וכך כיסוי חלוקה הפוכה.

בזמן המעברים נשמור מערך של סדר ההעברות ונערך ΔQ עבור כל חלוקה בדרך.

③

פרוייקט תוכנה - שיעור 11

לבסוף נעבור על כל ערכי ΔQ ונבחר את המקסימלי. במקרה הגרוע נבחר את החלוקה המקורית עבורה $\Delta Q=0$ ולכן הערך החדש לא יהיה קטן. אם ΔQ שיפר (כלומר היה חיובי), נבצע את התהליך שוב עד ש ΔQ ישאר 0 לאחר התהליך.

<u>כיצד למדוד [illegible] מוכלל</u>

באמצעות שיטת החזקה.

<u>הנחה</u> מטריצה הניתנת ללכסון:

מסדרים את הערכים לפי ערך מוחלט של λ.

שיטת החזקה שואפת ל $A^m x \sim V_1$. הסיבה לכך ש λ_1 הוא הערך הגבוה מכולם ולכן היא המשפיע ביותר עבור ערכים גדולים.