תרגול מס' 13 – אלגוריתמים הסתברותיים

עד עתה דיברנו רק על אלגוריתמים דטרמיניסטיים. בהינתן קלט קבוע לאלגוריתם דטרמיניסטי, כל הרצה שלו תחזיר את אותו הפלט. הוכחת נכונות של אלגוריתם דטרמיניסטי דורשת שעבור כל קלט האלגוריתם יתן תוצאה נכונה. אלגוריתם הסתברותי הינו אלגוריתם שבו חלק הפעולות הינן הגרלת ערכים למשתנים (לפעמים נאמר שהאלגוריתם "מטיל מטבעות"), ופעולת האלגוריתם תלויה בתוצאת ההגרלה. על כן, ייתכן ששתי הרצות של אלגוריתם הסתברותי על אותו קלט יסתיימו בפלט שונה. כשנדבר על נכונות אלגוריתם הסתברותי, נדרוש שלכל קלט ההסתברות שהאלגוריתם יחזיר תשובה נכונה היא גדולה. היתרון של האלגוריתמים ההסתברותיים שנראה הוא שהם יעילים יותר מהאלגוריתם הדטרמיניסטי לפתרון אותה הבעיה.

בעיית Matrix Identification:

מופע: 3 מטריצות מסדר $n \times n$ A, B, ו-C

צ"ל: האם $AB \neq C$?
שימו לב! הבעיה לעיל הינה בעיית הכרעה.

פתרון טריוויאלי: לבצע הכפלת מטריצות ולהשוות. זמן ריצה: $O(n^3)$.

היות ואנו מעוניינים בהכרעה בלבד - האם $AB \neq C$, נראה כי העלות לעיל היא יותר מהנדרש.
נראה כי ניתן לשפר את זמן הריצה באופן משמעותי אם נסכים לטעות חד-צדדית זניחה.

מה הכוונה טעות חד צדדית כאן?
אם AB = C, **תמיד** נענה "שווים".
אם AB ≠ C נענה "שונים" בהסתברות גבוהה מאד.

הערה: יתכן ש- AB ≠ C אבל נשיב "שווים" (כלומר נחשוב ש AB=C), אך **בהסתברות נמוכה מאד** (ולכן נטעה בהסתברות זניחה).

שלבי התכנון:
בשלב הראשון נייצר אלגוריתם הרץ ב-$O(n^2)$, וטועה (חד – צדדית) בהסתברות ½.
בשלב השני נתאר אלגוריתם אשר יקטין את הסתברות השגיאה.

שלב ראשון: אלגוריתם אקראי עם הסתברות לטעות חד-כיוונית 1/2

הסבר כללי:
האלגוריתם מבוסס על העובדה הבאה:
אם AB=C, אזי לכל וקטור v באורך n מתקיים $AB\vec{v} = C\vec{v}$.
אם נמצא וקטור שעבורו $AB\vec{v} \neq C\vec{v}$ נוכל להיות בטוחים כי AB ≠ C. לכן אנו נגריל וקטור, ואז נבדוק האם הוא מקיים את השוויון. בהמשך נראה כי קיימת שיטה יעילה לבדיקת השוויון וכי כמות הוקטורים שמובילים לתוצאה נכונה היא גדולה (כלומר ההסתברות להגריל וקטור כזה היא גבוהה).

האלגוריתם:

קלט: מטריצות מסדר $n \times n$ A, B ,C-ו
פלט: אם AB=C: יוחזר "שווים" בהסתברות 1.
אם AB≠ C: יוחזר "שונים" בהסתברות ≤ 1/2.

1. הגרל (בהסתברות אחידה) וקטור בוליאני $\vec{v} \in \{0,1\}^n$.
2. אם $AB\vec{v} \neq C\vec{v}$ החזר "שונים", אחרת החזר "שווים".

ניתוח זמן:
1. O(n)
2. נחלק לפעולות:
   a. $C\vec{v}$ – הכפלת $(n \times n) \cdot (n \times 1) \Leftarrow O(n^2)$.
   b. $B\vec{v}$ – הכפלת $(n \times n) \cdot (n \times 1) \Leftarrow O(n^2)$.
   c. $A(\overrightarrow{Bv})$ - הכפלת $(n \times n) \cdot (n \times 1) \Leftarrow O(n^2)$.
   d. $\overrightarrow{A(Bv)} - \overrightarrow{Cv}$ - חיסור שני וקטורים $\Leftarrow O(n)$
3. O(n) - השוואה בין שני ווקטורים.

סה"כ - $O(n^2)$ כנדרש

ניתוח הסתברות השגיאה של MatrixID

נגדיר: D = AB – C. נשים לב כי $AB = C$ אם ורק אם $D = \vec{0}$.

אם AB = C, אזי D = 0, ולכן לכל וקטור $\vec{v}$: $D\vec{v} = \vec{0}$, אזי תמיד נענה "שווים".
אם AB ≠ C, אזי D היא איננה מטריצת ה-0. נראה שבמקרה זה בהסתברות גדולה לפחות מחצי נענה "שונים".

<u>הניתוח הבא יראה כי הסיכוי לטעות במקרה בו AB ≠ C הוא ≤ ½:</u>

הגדרות: עבור $D \neq \vec{0}$
וקטור "שקרן" הוא וקטור בוליאני $\vec{v} \in \{0,1\}^n$ המקיים $D\vec{v} = 0$. המשמעות היא שניסינו להגריל וקטור שבאמצעותו נוכל להוכיח כי - $AB \neq C$ אך יצא לנו וקטור שעבורו $AB\vec{v} = C\vec{v}$.
וקטור "הגון" הוא וקטור $\vec{v} \in \{0,1\}^n$ המקיים $D\vec{v} \neq \vec{0}$ (הוא למעשה **עד** לכך ש-$AB \neq C$).

אבחנה:
אם $AB \neq C$, אזי ההסתברות לטעות שקולה להסתברות שנגריל וקטור שקרן.

אבחנה:
באופן דומה, הסתברות לא לטעות שווה להסתברות להגריל וקטור "הגון".

דוגמא:

נניח שנתונות המטריצות הבאות:

$$A=\begin{pmatrix}1&2&3\\1&2&3\\1&2&3\end{pmatrix}\quad B=\begin{pmatrix}1&2&3\\1&2&3\\1&2&3\end{pmatrix}\quad C=\begin{pmatrix}3&6&99\\3&6&99\\3&6&99\end{pmatrix}$$

ונניח שהוגרל הווקטור (השקרן) הבא:

$$v=\begin{pmatrix}1\\1\\0\end{pmatrix}$$

נשים לב ש:

$$AB=\begin{pmatrix}6&12&18\\6&12&18\\6&12&18\end{pmatrix}$$

כשנצבע את ההכפלה $A(Bv)$ נקבל:

$$A(Bv)=\begin{pmatrix}1&2&3\\1&2&3\\1&2&3\end{pmatrix}*\begin{pmatrix}1&2&3\\1&2&3\\1&2&3\end{pmatrix}*\begin{pmatrix}1\\1\\0\end{pmatrix}$$

$$A(Bv)=\begin{pmatrix}1&2&3\\1&2&3\\1&2&3\end{pmatrix}*\begin{pmatrix}3\\3\\3\end{pmatrix}$$

$$A(Bv)=\begin{pmatrix}9\\9\\9\end{pmatrix}$$

מצד שני כשנבצע את ההכפלה Cv נקבל:

$$Cv=\begin{pmatrix}3&6&99\\3&6&99\\3&6&99\end{pmatrix}*\begin{pmatrix}1\\1\\0\end{pmatrix}=\begin{pmatrix}9\\9\\9\end{pmatrix}$$

ולכן נסיק בטעות ש-$AB=C$, אך מנגד קיים $v'=\begin{pmatrix}1\\1\\1\end{pmatrix}$ הוא לא וקטור שקרן.

נשים לב שבמקרה זה $D=\begin{pmatrix}3&6&-81\\3&6&-81\\3&6&-81\end{pmatrix}$ ובפרט $D\neq\vec{0}$. קיבלנו את הוקטור ההוגן v' על-ידי היפוך הקוארדינטה השלישית בוקטור v.

טענה:

תהי D מטריצה $n\times n$ בוליאניות כך ש-$D\neq\vec{0}$. אזי עבור v ווקטור שונה מ-0 הנבחר אקראית בצורה אחידה מ $\{0,1\}^n$ מתקיים כי: $\Pr[D\vec{v}=\vec{0}]\leq\frac{1}{2}$ (זו ההסתברות לטעות).

הוכחה:

נראה כי אם D היא אינה מטריצת ה-0 אזי מספר הוקטורים השקרנים קטן שווה למספר הוקטורים ה"הגונים", ולכן צפיפות הוקטורים הרצויים ("הגונים") היא גבוהה מאוד (לפחות 50%).

היות וידוע כי $AB\neq C$ אזי קיים איבר ב D אשר שונה מ 0. נניח כי $D_{ij}\neq 0$.

תהי d השורה ב D_{ij} מופיע. נשים לב כי אם $dv \neq 0$ אזי $Dv \neq 0$. נראה אם כך, כי כמות הווקטורים המקיימים תכונה זו הינה לפחות 2^{n-1}. כעת, נבחר ווקטור $v = (v_1, v_2, \dots, v_n)$. וקטור v הוא רע אם:

$$(1) \quad d_{i1}v_1 + \cdots + d_{ij}v_j + \cdots + d_{in}v_n = 0$$

נשים לב כי אם נהפוך את הערך של v_i אז נקבל וקטור חדש שהוא "הוגן", כלומר על כל וקטור "שקרן" יש לנו וקטור "הוגן", כמו כן פעולת ההפיכה הזאת היא פעולה חח"ע ועל כן כמות הווקטורים "ההוגנים" גדולה שווה לכמות הווקטורים "השקרנים".

$$Number\ of\ lieing\ verctors \leq Number\ of\ fair\ verctors$$

סה"כ:

$$\Pr[The\ algorithm\ fails] = \frac{Number\ of\ lieing\ vectors}{Number\ of\ lieing\ verctors + Number\ of\ fair\ verctor}$$

$$\leq \frac{Number\ of\ lieing\ vectors}{2 * Number\ of\ lieing\ verctors} = \frac{1}{2}$$

מש"ל.

(**שאלה לסטודנטים**: *האם יתכן כי ההסתברות תהיה קטנה ממש מחצי?*)

שלב שני: שיפור ההסתברות

הוכחנו כי:

1. אם AB=C נזהה זאת בהסתברות 1.
2. אחרת, אם $AB \neq C$ נזהה זאת בהסתברות לפחות 1/2.

1/2 איננה הסתברות שמספקת אותנו. כיצד ניתן לשפרה?

<u>דרך 1:</u>

למשל נריץ 3 פעמים את האלגוריתם באופן בת"ל.
אם בכל 3 הריצות נקבל false נענה false. (במשמעות של "כנראה לא שונים")
אחרת נענה true (במשמעות של "שונה בוודאות" – כי מצאנו עד לשונות).

ומה הסיכוי לטעות?

1. אם AB = C, כל הריצות יענו "לא שונים" ולכן נענה "לא שונים" ונצדק
2. אם AB ≠ C, נטעה אם כל הריצות טעו. ההסתברות **שכל** הריצות יטעו היא לכל היותר ½ * ½ * ½ = 1/8 . אנו נטעה (ונגיד "לא שונים") רק אם **כל** הריצות יטעו.

במקרה כללי של k ריצות:
אם בכל k הריצות נקבל "שווים" נענה שווים.
אחרת נענה "שונים".

ומה הסיכוי לטעות?

1. אם $AB = C$, כל הרצות יחזירו "שווים" ולכן נענה "שווים" בצדק.
2. אם $AB \neq C$, ההסתברות שבכל הריצות נקבל "שווים" לכל היותר $\frac{1}{2^k}$ ולכן נטעה רק בהסתברות לכל היותר $\frac{1}{2^k}$.

במה תלוי זמן הריצה של האלגוריתם שנייצר?

- בגודל הקלט (כמובן).

- במידת הטעות לה אנו שואפים.

במקרה שלנו נקבל זמן ריצה של $O(k * n^2)$ עבור אלגוריתם בעל טעות חד צדדית בהסתברות $1/(2^k)$.
(עבור $k = 70$ זה יוצא טעות נדירה מאד!!)

<u>דרך 2:</u>
נגריל את הווקטורים מקבוצה גדולה יותר $\{0,1,\ldots,j\}$ וכעת ההסתברות לטעות תקטן ל $1/j$. אך כעת ייצוג המספרים ידרוש $\log j$ ביטים ופעולות המכפלה ידרשו יותר זמן.