

## וריאנט גאוס-ג'ורדן

השיטה דומה מאוד לשיטת האלימינציה של גאוס, אבל יש שני הבדלים:

1. נירמול. הופכים את אלמנטי האלכסון ל:1:

$$a_{kj}^{(k+1)} = \frac{a_{kj}^{(k)}}{a_{kk}^{(k)}} \quad b_k^{(k+1)} = \frac{b_k^{(k)}}{a_{kk}^{(k)}} \quad j = k, \dots, n$$

2. אלימינציה גם מעל האלכסון:

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - a_{ik}^{(k)} a_{kj}^{(k+1)} \quad b_j^{(k+1)} = b_j^{(k)} - a_{jk}^{(k)} b_k^{(k+1)}$$

עבור  $k$  השורה  $k$  תיראה כך:

$$0 \quad 0 \quad \dots \quad \boxed{1} \quad a_{k,k+1}^{(k+1)} \quad \dots \quad a_{k,j}^{(k+1)} \quad \dots \quad a_{k,n}^{(k+1)} \quad b_k^{(k+1)}$$

ובשורה  $i > k$

$$0 \quad 0 \quad \dots \quad \boxed{a_{ik}^{(k)}} \quad \dots \quad a_{ij}^{(k)} \quad \dots \quad a_{in}^{(k)} \quad b_j^{(k)}$$

## הערות

- האלגוריתם הופך את  $A^{(n)}$  למטריצת היחידה, ולכן  $b^{(n)}$  שווה לווקטור הנעלמים  $x$ , ומקבלים את התשובה ישירות, בלי צורך לבצע הצבה לאחור.
- הסיכוביות של האלגוריתם היא  $\frac{1}{2}n^3 \approx \frac{n^2(n+1)}{2} - \frac{1}{3}n^3$  בניגוד ל  $\frac{1}{3}n^3$  של השיטה המקורית.

## Pivoting/Scaling עבור הגדלת הדיוק

אם המטריצה  $A$  סינגולרית, בשלב מסויים ניתקל ב0 על האלכסון - מה שיעשה לנו בעיות בחישוב. אבל יכול גם להיות שהמערכת לא סינגולרית, אבל בכל זאת ניתקל באפס על האלכסון בגלל שגיאות קירוב. יכול גם להיות שלא ניתקל ב0 על האלכסון, אבל בגלל שיש איברים קטנים מאוד על האלכסון החישוב משתבש.

## Pivoting חלקי

נרצה שכופלי ה- $\frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$  יהיו קטנים ככל האפשר.

לכן בכל איטרציה נמצא את האלמנט שערכו המוחלט מקסימלי:

$$c_k = \max_{k \leq i \leq n} |a_{ik}^{(k)}|$$

בוחרים את האינדקס  $k \leq i$  עבורו  $c_k = c_i$ , ומחליפים את השורה ה- $i$  עם השורה ה- $k$ . זה לא מפר את שקילות המערכת - אבל צריך לזכור את ההחלפות האלה, כדי לבצע את ההצבה בסוף.

ההחלפה הזאת, שנקראת Pivoting, דואגת לכך שהכופלים בכל שורה לא יהיו גדולים מ-1.

## Pivoting מלא

הרעיון הוא שבשלב ה- $k$ , מסתכלים לא רק על (תת) העמודה  $k$ , אלא על כל תת המטריצה שמתחילה מ- $a_{kk}$ , ומחפשים את  $\max_{\substack{i \geq k \\ j \geq k}} |a_{ij}|$ . מחליפים גם שורה וגם עמודה - וצריך להביא את זה בחשבון כשמציבים בסוף.

## Scaling

הרעיון כאן הוא למצוא את קנה המידה הגדול ביותר של כל השורה. מחלקים האיבר בעמודה ה- $k$  בכל שורה שעוד לא דורגה באיבר הגדול ביותר בערכו המוחלט (בשורה), ורק אז בוחרים את האיבר הכי גדול.

## הערה

יש איזשהו קשר בין Pivoting מלא לבין Scaling, שכן בשניהם מסתכלים גם על האיברים שלא נמצאים בעמודה  $k$ .