1(a) Implement gaussian elimination

一張含有 文字, 筆跡, 字型, 書法 的圖片

AI 產生的內容可能不正確。

(b) Do partial pivoting

一張含有 文字, 筆跡, 字型, 書法 的圖片

AI 產生的內容可能不正確。

(c)Chop the number rather than rounding

一張含有 文字, 筆跡, 字型, 書法 的圖片

AI 產生的內容可能不正確。

(d) The result show that the Gaussian elimination best matches the original right-hand sides. Surprisingly, using partial pivoting does not do better. I think it may be because we only keep 3 significant numbers, which lead to the result. Also, part(c) match the right-hand side worse, since we always discard all the number after three significant numbers, which may lead to bigger error.

一張含有 文字, 筆跡, 字型 的圖片

AI 產生的內容可能不正確。

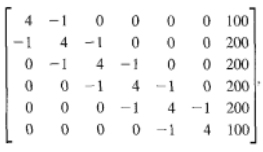
2(a)

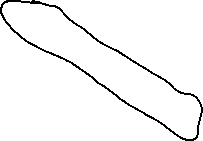
Since the matrix is symmetric, we can only use three columns for each equation, one for b and the other two for A (4, -1). To solve this equation, we just need to use simple Gaussian elimination. For each row, we must use A[i+1][i] / A[i][i] for gaussian elimination to eliminate the (-1) and update b[i+1], A[i+1][i+1]. The information we need is enough to store in a n\*3 matrix.

After finishing gaussian elimination, we use back substitution to get all the answers.

(b) Implementation

一張含有 文字, 螢幕擷取畫面, 軟體, 字型 的圖片

AI 產生的內容可能不正確。



Matrix A stores all the entries I circle in the above picture. Since the matric is symmetric, the value in constructed matrix A[0][1] store both the value in red circle and A[1~n-1][1] store the value in the same way. A[0~n-1][0] store all the diagonal value. In this way, we can store all the matrix in this form into n\*3 matrix without losing the information.

We can use the compact form matric to carefully processing the gaussian elimination and use back substitution for all the answer.

(c)

For each row (0 ~ n-2), when implementing Gaussian elimination, we need to compute the ratio for next row, then one value in A and one value in B need to multiply that ratio and minus to the next row. Therefore, we need 5(n-1) for this step. For each row(0~n-1), when implementing back substitution, we need to multiply one entry in A and minus it to B and finally do a division. Therefore, we need another 3\*n for this step, but the bottom row doesn’t need multiplication and subtraction.

Therefore, total arithmetic operations = 5n-5+3\*n-2 = 8n-7

3(a)

一張含有 文字, 螢幕擷取畫面, 軟體, 多媒體軟體 的圖片

AI 產生的內容可能不正確。 一張含有 文字, 時鐘, 字型 的圖片

AI 產生的內容可能不正確。

Using the equation from lecture slide to obtain new value for x at each iteration. When the max value from x\_new – x\_old < tol, I view it as converge and get the result x.

Result: 

(b)

一張含有 文字, 螢幕擷取畫面, 軟體, 字型 的圖片

AI 產生的內容可能不正確。一張含有 字型, 螢幕擷取畫面, 時鐘 的圖片

AI 產生的內容可能不正確。

Gauss-Seidel is similar to Jacobi method, but we use the new x value as soon as possible. I use the above equation from lecture slide to compute new x value. This equation is obtained from the below equation. The first summation is b minus the result from Ux and the second summation is solving the forward substitution of (L+D). We can get the above equation. The converge condition is same as Jacobi method. The converge iteration times is less than Jacobi method.

Result: 



4

一張含有 螢幕擷取畫面, 文字, 字型 的圖片

AI 產生的內容可能不正確。

一張含有 文字, 字型, 螢幕擷取畫面, 時鐘 的圖片

AI 產生的內容可能不正確。

Every time we update a value; we multiply a factor in order to get a faster convergence. I iterate all the possible factor from 1~2(gap : 0.01), the best result is

5.

I use infinity norm condition number to computer the 4 matrices’ condition number, here’s the result:

(I use function np.linalg.norm, np.linalg.inv since matlab have similar function)

一張含有 文字, 字型, 螢幕擷取畫面, 數字 的圖片

AI 產生的內容可能不正確。

Therefore, A, D is ill-conditioned and B, C is well-conditioned

Since D is singular matrix, the condition number of D is infinity.