Numerik 1 2018-11-18 Okhtay Wahid Far

1.Projekt: LR-Zerlegung und Gauß-Elimination

Thema:
Es soll eine reguläre Matrix A in eine linke untere und rechte obere Dreiecksmatrix zerlegt werden. Hierbei wird eine Pivotisierung vorgenommen wobei zusätzlich ein Permutationsvektor gebildet wird, der die Zeilenvertauschungen nachvollziehbar macht.

Die LR Zerlegung mit Pivotisierung

Nach Definition kann man eine reguläre Matrix in eine linke untere dreiecksmatrix (Lower) und eine rechte obere (Upper) zerlegen.

Linke untere Dreiecksmatrix rechte obere Dreiecksmatrix

Es wird dazu eine Permutationsmatrix benötigt, die indes eine Einheitsmatrix darstellt, die die Zeilenvertauschungen registriert.
So gilt also für die Permutationsmatrix und es gilt zudem
.

Das heißt mithilfe der Permutationsmatrix (hier Permutationsvektor) wird gemäß der Pivotisierung die Zeile mit dem betragsmäßig höchsten Eintrag in einer Spalte mit der obrigen vertauscht. Hiebei ist zu beachten,dass man sich unterhalb der Hauptdiagonalen orientiert und demnach die Möglichkeiten der Zeilenvertauschungen kleiner werden , je näher man sich der Stufenform anpasst.

Mithilfe der Frobeniusmatrix $L_{(n)}$ wird dann die tatsächliche Gausselimination durchgeführt, die i) auf ihrer Hauptdiagonalen nur Einsen enthält und ii) höchstens nur in einer Spalte unter der Hauptdiagonalen beliebige Einträge zu stehen hat.
Für jeden Eliminationsschritt wird dann eine spezielle Frobeniusmatrix gebildet, worin die Faktoren für die Elimination gemäß der Stufenform enthalten sind.
So gilt also: $L_{(n)}*P_{(n)}*...*L_{(1)}*P_{(1)}*A = R$

Durch bilden der Inverse der Frobeniusmatrizen kann man die L-Matrix bestimmen:
$P*A = L_{(n)}^{-1} * L_{(n-1)}^{-1} *...*R$

Implementierung:

Um eine LR-Zerlegung mit Pivotisierung zu realisieren, haben wir vier Funktionen verfasst.
Wir nutzen hierzu in Matlab das Schlüsselwort function und die folgende syntax hierzu:

[<AusgabeParameter1>,<AusgParameter2>]=function(<Parameter1>,<Parameter2>)
.
.
.
endfunction

Im Folgenden werde ich die vier geschriebenen funktionen näher erläutern.
Die Funktion der LR-Zerlegung erstellt eine rechtsobere und linksuntere Dreiecksmatrix
und einen Permutationsvektor aus A.

Der Permutationsvektor ist ein Spaltenvektor der nach jeder neuen Pivotisierung und Gaußelimination neu bestimmt wird. Er muss daher wieder korrekt ausgerichtet sein, was wir mit der Zeile *l = l+ (j-1)* in unserem Code erreichen.

Durch den Befehl *triu(A)* wird eine rechte obere Dreiecksmatrix aus der neu erzeugten Matrix A erstellt. Der Befehl *tril(A)* bestimmt hingegen eine linke untere Dreiecksmatrix.
Die Funktion der Vorwärtssubstitution benötigt als Eingabeparameter die linke untere Dreiecksmatrix L und den Vektor b. Vor allem sind jene Codezeilen ausschlaggebend:

x(i)=b(i);
*x(i) = x(i)-R(i,k)*x(k);*

Das heißt, der i-Wert von b wird i-ten Wert von x zugewiesen, womit immer der nächste i-Wert von x aufgrund der Zählvariable der for-Schleife neu bestimmt wird.

Die Rückwärtssubstitution wird ähnlich berechnet, wobei der Wert des letzten berechneten Schrittes durch den Vorfaktor geteilt wird:
x(l) = x(l) / R(l, l);

Die Funktion lgs hat als Eingabeparameter die Matrix und den Vektor b und gibt den gesuchten Lösungsvektor x aus. Diese Funktion hat Einfluss auf alle anderen Funktionen, da sie überprüft ob die Matrix A quadratisch ist und ob die Dimension des Vektors b mit der von A übereinstimmt. Andernfalls wäre ein LR-Zerlegung garnicht möglich.