Εισαγωγή στη Matlab 3

Εισαγωγή στην Αριθμητική Ανάλυση

Διδάσκων: Γεώργιος Ακρίβης

Βοηθός: Δημήτριος Ζαβαντής
email: dzavanti@cs.uoi.gr

Περιεχόμενα

- Εξίσωση $2ου$ Βαθμού
- Επίλυση Συστημάτων Γραμμικών Εξισώσεων
  - Μέθοδος Gauss
  - Μέθοδος Gauss – Jordan
  - Υπολογισμός Ορίζουσας
  - Υπολογισμός Αντίστροφου Πίνακα
- Υπολογισμός ριζών μη γραμμικών αλγεβρικών εξισώσεων
  - Μέθοδος διχοτόμησης
  - Μέθοδος Newton-Raphson
  - Μέθοδος τέμνουσας

Εξίσωση 2$ου$ βαθμού

$for\ i = 1:1:3$

$\quad fprint f('a\%d\ \ = ', i);$

$\quad a(i) = input('');$

$\quad fprint f('\backslash\mathrm{n}');$

end

$res\ \ = \ roots([a(1)\ a(2)\ a(3)])$

Μέθοδος Gauss

$\% \, a11x + a12y + a13z = a14$

$\% \, a21x + a22y + a23z = a24$

$\% \, a31x + a32y + a33z = a34$

$li = 3;$

$co = li + 1;$

$[a] = fun1(li, co);$

$[a] = fun3(li, co, a);$

$[r] = fun4(li, co, a);$

$for \; j = 1{:}1{:}li$

$\quad fprintf('x\%d = \%.4f\backslash\mathrm{n}', j, r(j));$

end

Μέθοδος Gauss

$function\ [b] = fun1(lin, col)$

$for\ i = 1{:}1{:}lin$

$\quad for\ j = 1{:}1{:}col$

$\qquad fprintf('a\%d\%d = ', i, j);$

$\qquad b(i, j) = input('');$

$\qquad fprintf('\backslash n');$

$\quad end$

end

Είσοδος τιμών

Μέθοδος Gauss

$function\ [a] = fun3(lin, col, a)$

$for\ diag = 1{:}1{:}lin - 1$

$\quad for\ i = diag + 1{:}1{:}lin$

$\quad\quad a\_temp = -a(i, diag)/a(diag, diag);$

$\quad\quad for\ k = 1{:}1{:}col$

$\quad\quad\quad a(i, k) = a(i, k) + a(diag, k) * a\_temp;$

$\quad\quad end$

$\quad end$

end

Τριγωνοποίηση

Μέθοδος Gauss

$function\ [re] = fun4(lin, col, a)$

$for\ i1 = lin{:}-1{:}\,1$

$\quad sum\ =\ 0;$

$\quad for\ c1 = i1 + 1{:}\,1{:}\,col - 1$

$\qquad sum\ =\ sum\ +\ a(i1, c1) * re(c1);$

$\quad end$

$\quad re(i1) = (a(i1, col) - sum)/a(i1, i1);$

end

Πίσω Αντικατάσταση

Μέθοδος Gauss-Jordan

```
function [a] = fun3_b(lin, col, a)

for diag = 1:1:lin
   for i = 1:1:lin
      if(i~ = diag)
         a_temp = -a(i, diag)/a(diag, diag);
         for k = 1:1:col
            a(i, k) = a(i, k) + a(diag, k) * a_temp;
         end
      end
   end
end
end
```

```
function [re] = fun4_b(lin, col, a)

for i1 = 1:1:lin
   re(i1)  =  a(i1, col)/a(i1, i1);
end
end
```

Υπολογισμός Ορίζουσας

- Οι μέθοδοι Gauss & Gauss – Jordan δημιουργούν άνω τριγωνικούς και διαγώνιους πίνακες αντίστοιχα
- Ο υπολογισμός της ορίζουσας προκύπτει από τον πολλαπλασιασμό των στοιχείων της διαγωνίου

Υπολογισμός Αντίστροφου Πίνακα

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Πρέπει να ισχύει:

$$\mathrm{AB} = \mathrm{BA} = I$$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}\begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Υπολογισμός Αντίστροφου Πίνακα

Επιλέγοντας το πρώτο γινόμενο μπορούμε να το σπάσουμε στα ακόλουθα 3 γραμμικά συστήματα με 3 αγνώστους το καθένα.

Προκύπτουν 3 πίνακες που επιλύονται με τις μεθόδους Gauss ή Gauss – Jordan

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} b_{11} \\ b_{21} \\ b_{31} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} b_{12} \\ b_{22} \\ b_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \qquad \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} b_{13} \\ b_{23} \\ b_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Μέθοδος Διχοτόμησης

$epsilon \ = \ 1e-6;$

$a \ = \ -1;$

$b \ = \ 1;$

$while(abs(b-a) > epsilon)$

$\quad fa \ = \ fx(a);$

$\quad fb \ = \ fx(b);$

$\quad c = (a+b)/2;$

$\quad fc \ = \ fx(c);$

$\quad if(fa * fc < 0)$

$\quad\quad b = c;$

$\quad else$

$\quad\quad a = c;$

$\quad end$

end

$fprintf('a = \%.7f\backslash\mathrm{nb} = \%.7f\backslash\mathrm{nf}(a) = \%.7f\backslash\mathrm{nf}(b) = \%.7f\backslash\mathrm{n}', a, b, fa, fb);$

$function\ [f] = fx(x)$

$f \ = \ x\text{^}5 + x + 1;$

end

$a = -0.7548780$
$b = -0.7548771$
$f(a) = -0.0000035$
$f(b) = 0.0000015$

Μέθοδος Newton-Raphson

- Η αναδρομική σχέση της μεθόδου έχει προκύψει από τη σχέση: $\rho = x - \frac{f(x)}{f\prime(x)}$ όπου στην τυχαία τιμή του x αντικαθιστούμε την n-οστή προσέγγιση και στη θέση της ρίζας ρ την επόμενη προσέγγιση
- Η σχέση που προκύπτει είναι: $x_{n+1} = x_n - \frac{f(x_n)}{f\prime(x_n)}$
- Η χρήση της παραπάνω σχέσης πρέπει να γίνεται με ένα μέγιστο αριθμό επαναλήψεων (αποφυγή ατέρμων βρόχου)

Μέθοδος Τέμνουσας

- Η μόνη διαφορά από τη μέθοδο Newton – Raphson είναι ότι στη συγκεκριμένη περίπτωση η $f'(x_n)$ αντικαθίσταται κατά προσέγγιση από την διηρημένη διαφορά: $f'(x_n) \approx \frac{f(x_n)-f(x_{n-1})}{x_n-x_{n-1}}$
- Η σχέση που προκύπτει είναι: $x_{n+1} = x_n - \frac{f(x_n)(x_n-x_{n-1})}{f(x_n)-f(x_{n-1})}$
- Πολύ θετικό το γεγονός ότι δεν χρειάζεται να γνωρίζουμε την παράγωγο της συνάρτησης

Μέθοδος Τέμνουσας

$function\ [f] = fx(x)$

$f\ =\ 5 * (x.\,^\wedge 3) - 2 * (x.\,^\wedge 2) + \cos(x);$

end

$\gg temnousa('fx', -3, 1, 5 * 10^{-15}, 10000)$

$H\ riza\ vre8hke\ me\ thn\ akriveia\ pou\ zhth8hke$

$step$	x	y
1.0000	− 3.0000	− 153.9900
2.0000	1.0000	3.5403
3.0000	0.9101	2.7262
4.0000	0.6091	1.2079
5.0000	0.3695	0.9117
6.0000	− 0.3676	0.4146
7.0000	− 0.9825	− 6.1168
8.0000	− 0.4066	0.2516
9.0000	− 0.4294	0.1447
10.0000	− 0.4602	− 0.0147
11.0000	− 0.4573	0.0007
12.0000	− 0.4575	0.0000
13.0000	− 0.4575	− 0.0000

Ευχαριστώ!!!