

Αριθμητική Γραμμική Άλγεβρα Project 2

Μιχαήλ Ανάργυρος Ζαμάγιας
TP5000

26 Μαΐου 2020

Περιεχόμενα

1	Άσκηση 1	2
1.1	Εκφώνηση	2
1.2	Απάντηση	3
1.2.1	Πρόγραμμα	3
2	Άσκηση 2	6
2.1	Εκφώνηση	6
2.2	Απάντηση	7
2.2.1	Πρόγραμμα	7
3	Άσκηση 3	8
3.1	Εκφώνηση	8
3.2	Απάντηση	9
3.2.1	Έλεγχος με το χέρι	9
3.2.2	Πρόγραμμα	11
4	Άσκηση 4	12
4.1	Εκφώνηση	12
4.2	Απάντηση	13
5	Πηγές	14
5.1	Βιβλία	15
5.2	Video	15
5.3	Σύνδεσμοι	15
5.4	Χρήσιμα αρχεία	15

Άσκηση 1

1.1 Εκφώνηση

Δίνεται το σύστημα (1.1). Γράψτε αυτό το σύστημα στην μορφή $A\bar{x} = \bar{b}$ για κατάλληλα διανύσματα \bar{x} και \bar{b} . Εδώ μ είναι το τελευταίο ψηφίο του αριθμού μητρώου σας. Φτιάξτε ένα πρόγραμμα που να δέχεται ως δεδομένα έναν 3×3 πίνακα A και ένα 3×1 διάνυσμα \bar{b} και να είναι σε θέση να μας πει αν το σύστημα είναι αόριστο ή αδύνατο ή έχει μοναδική λύση. Αν το σύστημα έχει μοναδική λύση πρέπει να δίνει αυτή την λύση υπολογισμένη με την μέθοδο του επαυξημένου πίνακα (απαλοιφή Gauss). Δώστε τα αποτελέσματα του προγράμματός σας στο σύστημα εξισώσεων (1.1) καθώς και στο σύστημα (1.2).

$$\begin{aligned} 3x + 4y + 2z &= 3 \\ x - 2y &= 3 \\ \mu x + 3y + z &= 2 \end{aligned} \tag{1.1}$$

$$\begin{aligned} \mu x + y + z &= 3 \\ x + 2y &= \mu \\ (\mu - 1) + 3y + z &= 2 \end{aligned} \tag{1.2}$$

1.2 Απάντηση

1.2.1 Πρόγραμμα

Το (1.1) γράφεται ως (1.3) στην μορφή $A\bar{x} = \bar{b}$. Ομοίως, το (1.2) γράφεται ως (1.4), με $\mu = 0$ για αριθμό μητρώου 5000.

$$\underbrace{\begin{pmatrix} 3 & 4 & 2 \\ 1 & -2 & 0 \\ 0 & 3 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} x \\ y \\ z \end{pmatrix}}_{\bar{x}} = \underbrace{\begin{pmatrix} 3 \\ 3 \\ 2 \end{pmatrix}}_{\bar{b}} \quad (1.3)$$

$$\underbrace{\begin{pmatrix} 0 & 1 & 1 \\ 1 & 2 & 0 \\ -1 & 3 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} x \\ y \\ z \end{pmatrix}}_{\bar{x}} = \underbrace{\begin{pmatrix} 3 \\ 0 \\ 2 \end{pmatrix}}_{\bar{b}} \quad (1.4)$$

Η έξοδος του προγράμματός μου για τα συστήματα (1.3) και (1.4):

```
A =
  3   4   2
  1  -2   0
  0   3   1
```

```
b =
  3
  3
  2
```

The system has a unique non-trivial solution:

```
x = -2.0000
y = -2.5000
z =  9.5000
```

```
A =
  0   1   1
  1   2   0
 -1   3   1
```

```
b =
  3
  0
  2
```

The system has a unique non-trivial solution:

```
x =  0.50000
y = -0.25000
z =  3.2500
```

Το κύριο πρόγραμμα:

```
1 clc;
2 clear all;
3 close all;
4
5 # matrix A with system coefficients
6 A = [3, 4, 2; 1, -2, 0; 0, 3, 1];
7 # matrix b with system constants
8 b = [3; 3; 2];
9 # print A and b
10 disp("A =");
11 disp(A);
12 disp("b =");
13 disp(b);
14 # run first task's logic
15 [x, y, z] = function_task1(A, b)
16
17 # add new line to seperate outputs
18 disp("\n")
19
20 # matrix A with system coefficients
21 A = [0, 1, 1; 1, 2, 0; -1, 3, 1];
22 # matrix b with system constants
23 b = [3; 0; 2];
24 # print A and b
25 disp("A =");
26 disp(A);
27 disp("b =");
28 disp(b);
29 # run first task's logic
30 [x, y, z] = function_task1(A, b)
```

Η συνάρτηση function_task1:

```
1 function [x, y, z] = function_task1(A, b)
2     # calculate A's determinant
3     det_A = det(A);
4     # check if the system has a unique non-trivial solution
5     # check if A's determinant is non-zero
6     if (det_A != 0)
7         disp("The system has a unique non-trivial solution:")
8         # calculate system's unique non-trivial solution
9         # using Gaussian elimination
10        Axb_roots = A\b;
11        # convert negative zeros to zeros (-0 to 0)
12        Axb_roots(Axb_roots == 0) = 0;
13        # display solution
14        x = Axb_roots(1);
15        y = Axb_roots(2);
16        z = Axb_roots(3);
17    else
18        # get the size of b matrix
19        n = size(b)(1);
20        # calculate Cramer's determinants for the system
21        x_up = A;
22        y_up = A;
23        z_up = A;
24        for i = 1 : n
25            x_up(i, 1) = b(i);
26        end
27        det_x = det(x_up)/det_A;
28        for i = 1 : n
29            y_up(i, 2) = b(i);
30        end
31        det_y = det(y_up)/det_A;
32        for i = 1 : n
33            z_up(i, 3) = b(i);
34        end
35        det_z = det(z_up)/det_A;
36        # check if all of Cramer's determinants are equal to zero,
37        # then the system has infinite solutions
38        if (det_x == 0 && det_y == 0 && det_z == 0)
39            disp("The system has infinite solutions.")
40        # if not all of Cramer's determinants are equal to zero,
41        # then the system has no non-trivial solutions
42        else
43            disp("The system has no non-trivial solutions.")
44        endif
45    endif
46 end
```

Άσκηση 2

2.1 Εκφώνηση

Φτιάξτε ένα πρόγραμμα για την αντιστροφή ενός πίνακα A με την μέθοδο Gauss, προσαρτώντας δηλαδή τον ταυτοτικό πίνακα A και ακολουθώντας τον αλγόριθμο της απαλοιφής Gauss. Εξηγήστε γιατί με αυτόν τον τρόπο παίρνετε τον αντίστροφο πίνακα. Το πρόγραμμά σας πρέπει να είναι σε θέση να βρίσκει αν ο πίνακας A είναι αντιστρέψιμος. Εφαρμόστε το πρόγραμμά σας στον πίνακα (2.1).

$$A = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 1 & 0 \\ -1 & 1 & \mu \end{pmatrix} \quad (2.1)$$

2.2 Απάντηση

2.2.1 Πρόγραμμα

Ο πίνακας (2.1) γράφεται ως (2.2) για $\mu = 0$.

$$A = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 1 & 0 \\ -1 & 1 & 0 \end{pmatrix} \quad (2.2)$$

Η έξοδος του προγράμματός μου για τον πίνακα (2.2):

Matrix A is non-invertible.

Το πρόγραμμα:

```
1 clc;
2 clear all;
3 close all;
4
5 # square matrix A
6 A = [1, 2, 1; 2, 1, 0; -1, 1, 0];
7 # make an identity matrix the size of A
8 I_M = eye(size(A)(1));
9 # apply gaussian elimination to A and I
10 B = round(A\I_M);
11 # convert negative zeros to zeros (-0 to 0)
12 B(B == 0) = 0;
13
14 # check if A is invertible based on the relation A*A^-1=I,
15 # check if A*B is equal to I, where B=A^-1
16 if (isequal(A*B, I_M) == 1)
17     disp("Matrix A is invertible.")
18     disp("A =");
19     disp(A);
20     disp("B = A\I");
21     disp(B);
22 else
23     disp("Matrix A is non-invertible.")
24 endif
```

Άσκηση 3

3.1 Εκφώνηση

Θεωρήστε τα διανύσματα του (3.1), όπου μ είναι το τελευταίο ψηφίο του αριθμού μητρώου σας. Ελέγξτε με το χέρι αν τα παραπάνω διανύσματα είναι γραμμικά ανεξάρτητα ή όχι χωρίς την χρήση του κριτηρίου της ορίζουσας. Αποτελούν αυτά τα διανύσματα βάση του R^3 ; Κατόπιν φτιάξτε ένα πρόγραμμα που να αποφασίζει αν μία οποιαδήποτε τριάδα διανυσμάτων στο R^3 αποτελεί βάση του R^3 κάνοντας χρήση του κριτηρίου της ορίζουσας.

$$\begin{aligned}u &= 3i - 4j + 5k \\v &= 2i - 3j + k \\w &= i - j + \mu k\end{aligned}\tag{3.1}$$

3.2 Απάντηση

3.2.1 Έλεγχος με το χέρι

Στο χέρι

Το σύστημα διανυσμάτων (3.1) γράφεται ως το (3.2), για $\mu = 0$.

$$\begin{aligned} u &= 3i - 4j + 5k \\ v &= 2i - 3j + k \\ w &= i - j + 0k \end{aligned} \quad (3.2)$$

Από το (3.2) μπορώ να πάρω την σχέση:

$$\lambda_1 \bar{u} + \lambda_2 \bar{v} + \lambda_3 \bar{w} = \bar{0} \quad (3.3)$$

Και για να είναι γραμμικώς ανεξάρτητα τα \bar{u} , \bar{v} , \bar{w} πρέπει να ισχύει $\lambda_1 = 0$, $\lambda_2 = 0$, $\lambda_3 = 0$.

$$\begin{aligned} \lambda_1 \begin{pmatrix} 3 \\ -4 \\ 5 \end{pmatrix} + \lambda_2 \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix} + \lambda_3 \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \implies \\ \underbrace{\begin{pmatrix} 3 & -4 & 5 \\ 2 & -3 & 1 \\ 1 & -1 & 0 \end{pmatrix}}_A \underbrace{\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix}}_{\bar{x}} &= \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}}_{\bar{b}} \end{aligned} \quad (3.4)$$

Με γραμμοπράξεις μετατρέπω τον πίνακα A του (3.4) σε ανηγμένο κλιμακωτό:

$$\begin{aligned} \begin{pmatrix} 3 & -4 & 5 \\ 2 & -3 & 1 \\ 1 & -1 & 0 \end{pmatrix} &\xrightarrow[\substack{R_3 - (1/3)R_1}]{R_2 - (2/3)R_1} \begin{pmatrix} 3 & -4 & 5 \\ 0 & -\frac{1}{3} & -\frac{7}{3} \\ 0 & \frac{1}{3} & -\frac{5}{3} \end{pmatrix} \xrightarrow{R_3 + R_2} \begin{pmatrix} 3 & -4 & 5 \\ 0 & -\frac{1}{3} & -\frac{7}{3} \\ 0 & 0 & -4 \end{pmatrix} \xrightarrow[\substack{R_3 - (1/4)R_3}]{R_2 + (7/3)R_2} \\ \begin{pmatrix} 3 & -4 & 5 \\ 0 & -\frac{1}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix} &\xrightarrow[\substack{-3R_2}]{R_1 - 5R_3} \begin{pmatrix} 3 & -4 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow{R_1 + 4R_2} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow{(1/3)R_1} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned}$$

Είναι:

$$\underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_A \underbrace{\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix}}_{\bar{x}} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}}_{\bar{b}} \implies \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Άρα όντως είναι $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$, οπότε και τα $\bar{u}, \bar{v}, \bar{w}$ είναι γραμμικώς ανεξάρτητα στο R^3 .

Στην Octave

Για να είναι γραμμικώς ανεξάρτητα τα $\bar{u}, \bar{v}, \bar{w}$ πρέπει να ισχύει η σχέση από το (3.3):

$$\lambda_1 = \lambda_2 = \lambda_3 = 0$$

Από το γραμμικό σύστημα εξισώσεων (3.4):

$$\underbrace{\begin{pmatrix} 3 & -4 & 5 \\ 2 & -3 & 1 \\ 1 & -1 & 0 \end{pmatrix}}_A \underbrace{\begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix}}_{\bar{x}} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}}_{\bar{b}} \quad (3.5)$$

Εφαρμόζω στο (3.5) την συνάρτηση `function_task1`¹ από την Άσκηση 1 κι έχω έξοδο:

```
The system has a unique non-trivial solution:
l1 = 0
l2 = 0
l3 = 0
```

Έτσι, με $\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$ τα $\bar{u}, \bar{v}, \bar{w}$ είναι γραμμικώς ανεξάρτητα στο R^3 .

¹(βλ. σελίδα 5)

Το πρόγραμμα:

```
1 clc;
2 clear all;
3 close all;
4
5 # vectors u, v, w
6 u = [3, -4, 5];
7 v = [2, -3, 1];
8 w = [1, -1, 0];
9 # matrix A from vectors
10 A = [u; v; w];
11 # make zero column matrix b the size of A
12 b = zeros(size(A)(1), 1);
13 # run first task's logic
14 [l1, l2, l3] = function_task1(A, b)
```

3.2.2 Πρόγραμμα

Άσκηση 4

4.1 Εκφώνηση

Θεωρήστε τα διανύσματα του (4.1). Δείξτε ότι αυτά τα διανύσματα αποτελούν βάση του R^3 . Εφαρμόστε με το χέρι τον αλγόριθμο Gram-Smith για την ορθοκανονικοποίηση της βάσης. Φτιάξτε ένα πρόγραμμα που θα δέχεται σαν είσοδο τρία διανύσματα σαν στήλες ενός 3×3 πίνακα, θα ελέγχει αν αυτά τα διανύσματα αποτελούν βάση του R^3 , και κατόπιν θα εφαρμόζει τον αλγόριθμο Gram-Smith και θα δίνει σαν έξοδο μία ορθοκανονική βάση.

$$\begin{aligned}u &= i - 2j + 3k \\v &= 2i - j + 2k \\w &= i - 2j + k\end{aligned}\tag{4.1}$$

4.2 Απάντηση

Πηγές

5.1 Βιβλία

5.2 Video

- [Inverse Matrix Using Gauss-Jordan](#)
- [Solving a 3x3 System Using Cramer's Rule](#)
- [Εύρεση αντίστροφου πίνακα με μέθοδο Gauss](#)
- [Γραμμική εξάρτηση διανυσμάτων](#)
- [Linear Independence and Linear Dependence, Ex 1](#)

5.3 Σύνδεσμοι

- [Systems of Linear Equations](#)
- [Γραμμικά συστήματα Εξισώσεων](#)
- [11. Solving \$Ax = b\$, Matrix division and the slash operator](#)
- [Gaussian elimination calculator](#)
- [Cramer's Rule](#)
- [Inserting code in a LaTeX document](#)

5.4 Χρήσιμα αρχεία

- [Octave, Vectors and Matrices](#)
- [Hardcore LaTeX Math](#)