(6) assignment

Assignment on topics in linear Algebra & systems of Equations.

$\boxed{\text{Bahare Zare}}$

1. if $A \in \mathbb{R}^{m\times n}$, with $\operatorname{rank}(A)=m$, explain why the system $Ax=b$ has always a solution. Consider Matrix A as belows:

$$A = \begin{pmatrix} a_{11} & a_{12} \cdots & a_{1n} \\ a_{21} & a_{22} \cdots & a_{2n} \\ \\ a_{m1} & \cdots\cdots & a_{mn} \end{pmatrix}.$$

$\operatorname{rank}(A)=m$, it means that A is spanned by m independent vectors. lets define them $V_1, \dots, V_n \Rightarrow A = \begin{pmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{pmatrix}$

and we recall that $Ax = x_1 v_1 + x_2 v_2 + \dots + x_n v_n$'s! is the linear combination of the Columns of A whose Coefficients are the components of x. Thus the system $Ax=b$ has a solution if and only if b can written as a linear combination of the columns of A. we define the Column space of A to be $R(A) = \operatorname{Span}\langle v_1, \dots, v_n\rangle$.

Since every such linear combination takes the form Ax for some x in $\mathbb{R}^n$ and since conversely every vector of the form Ax is such a linear combination we can express the Column space as $R(A) = \{Ax \mid x \in \mathbb{R}^n\}$.

Since the Columns of A are vectors in $\mathbb{R}^m$, or since equivalently Ax is in $\mathbb{R}^m$ for every x in $\mathbb{R}^n$, the Column space of A is subset of $\mathbb{R}^m$. So we have this proposition that the System $Ax=b$ has a solution if and only if b is in $R(A)$.

1) Countinue of answer 1/

by definition, $\dim(R(A)) = m$. since $R(A)$ is a subset of $\mathbb{R}^m$, it follows that $R(A) = \mathbb{R}^m$. therefore any vector $b \in \mathbb{R}^m$ satisfies $b \in R(A)$. By proposition we recalled for any vector $b \in \mathbb{R}^m$, there exists at least one solution x to $Ax = b$.

2) Consider a vector $x \in \mathbb{R}^n$ and the matrix $A = xx^T$.

Prove that Ay is parallel to x for every $y \in \mathbb{R}^n$.
verify this numerically using MATLAB and random vectors x, y.

for proving that Ay is parallel to x for every $y \in \mathbb{R}^n$, we equivalently will show $\forall y \in \mathbb{R}^n \; Ay = \lambda x$ that λ is a constant value. we will prove it by induction on n.

$* \; n=2 \quad x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \; x^T = [x_1 \; x_2], \; y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$

$$A = xx^T \Rightarrow A = x_1 \begin{bmatrix} x_1 & 0 \\ x_2 & 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 & x_1 \\ 0 & x_2 \end{bmatrix}$$

$$\forall y \quad Ay = \Big(x_1 \underbrace{\begin{bmatrix} x_1 & 0 \\ x_2 & 0 \end{bmatrix}}_{2\times 2} + x_2 \begin{bmatrix} 0 & x_1 \\ 0 & x_2 \end{bmatrix} \Big) \underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{2\times 1}$$

$$= \begin{bmatrix} x_1^2 y_1 \\ x_1 x_2 y_1 \end{bmatrix} + \begin{bmatrix} x_2 x_1 y_2 \\ x_2^2 y_2 \end{bmatrix} = \begin{bmatrix} x_1^2 y_1 + x_2 x_1 y_2 \\ x_1 x_2 y_1 + x_2^2 y_2 \end{bmatrix}$$

$$= \begin{bmatrix} x_1 (x_1 y_1 + x_2 y_2) \\ x_2 (x_1 y_1 + x_2 y_2) \end{bmatrix} = \underbrace{(x_1 y_1 + x_2 y_2)}_{\text{Constant Coefficient}} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda x \quad \checkmark$$

2/

Continue of answer 2)

hypothesis: $\forall y$ $Ay = \lambda x$ is for $n=K$ satisfied. ✓

Now we want to show for $n=K+1$ it is True.

$$n=K+1. \quad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{K+1} \end{bmatrix}_{K+1\times 1}, \quad x^T = \begin{bmatrix} x_1 & x_2 & \cdots & x_K & x_{K+1} \end{bmatrix}_{1\times K+1}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_{K+1} \end{bmatrix}$$

$$A = xx^T \Rightarrow A = x_1 \begin{bmatrix} x_1 & 0 & \cdots & 0 \\ x_2 & \ddots & & \\ \vdots & & \ddots & \\ x_K & & & \\ x_{K+1} & & & 0 \end{bmatrix} + x_2 \begin{bmatrix} 0 & x_1 & \cdots & 0 \\ \vdots & x_2 & & \vdots \\ & \vdots & \ddots & \\ 0 & x_{K+1} & \cdots & 0 \end{bmatrix} + \cdots + x_{K+1} \begin{bmatrix} 0 & \cdots & x_1 \\ & & x_2 \\ 0 & \ddots & \vdots \\ 0 & \cdots & x_{K+1} \end{bmatrix}$$

$$\forall y = \begin{bmatrix} y_1 \\ \vdots \\ y_{K+1} \end{bmatrix}, \quad Ay = \begin{bmatrix} x_1^2 y_1 \\ x_1 x_2 y_1 \\ \vdots \\ x_1 x_K y_1 \\ x_1 x_{K+1} y_1 \end{bmatrix} + \begin{bmatrix} x_2 x_1 y_2 \\ x_2^2 y_2 \\ \vdots \\ x_2 x_{K+1} y_2 \end{bmatrix} + \cdots + \begin{bmatrix} x_K x_1 y_K \\ x_K x_2 y_K \\ \vdots \\ x_K x_{K+1} y_K \end{bmatrix} + \begin{bmatrix} x_{K+1} x_1 y_{K+1} \\ x_{K+1} x_2 y_{K+1} \\ \vdots \\ x_{K+1}^2 y_{K+1} \end{bmatrix}$$

$$= \begin{bmatrix} x_1 (\overbrace{x_1 y_1 + x_2 y_2 + \cdots + x_K y_K}^{\lambda} + x_{K+1} y_{K+1}) \\ x_2 (x_1 y_1 + x_2 y_2 + \cdots + x_K y_K + x_{K+1} y_{K+1} \\ \vdots \\ x_K (x_1 y_1 + x_2 y_2 + \cdots + x_K y_K + x_{K+1} y_{K+1} \\ x_{K+1} (x_1 y_1 + x_2 y_2 + \cdots + x_K y_K + x_{K+1} y_{K+1} \end{bmatrix}$$

$$= \begin{bmatrix} x_1 (\lambda + x_{K+1} y_{K+1}) \\ x_2 (\lambda + x_{K+1} y_{K+1}) \\ \vdots \\ x_{K+1} (\lambda + x_{K+1} y_{K+1}) \end{bmatrix} = \underbrace{\begin{bmatrix} x_1 \lambda \\ x_2 \lambda \\ \vdots \\ x_{K+1} \lambda \end{bmatrix}}_{*} + \begin{bmatrix} x_1 x_{K+1} y_{K+1} \\ x_2 x_{K+1} y_{K+1} \\ \vdots \\ x_{K+1} x_{K+1} y_{K+1} \end{bmatrix}$$

$$* = \underbrace{\begin{bmatrix} x_1 \lambda \\ x_2 \lambda \\ \vdots \\ x_K \lambda \\ 0 \end{bmatrix}}_{(**)}{}_{K+1\times 1} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ x_{K+1} \lambda \end{bmatrix}$$ and we know that $(**)$ Can change to

$$(**) = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & \ddots & \\ 0 & 0 & \cdots & \cdots & 1 \\ 0 & 0 & \cdots & \cdots & 0 \end{bmatrix}_{K+1\times K} \times \begin{bmatrix} x_1 \\ \vdots \\ x_K \end{bmatrix}_{K\times 1}$$

3/

Continue of answer 2)

As (**) is a Coeffient of $\begin{bmatrix} x_1 \\ \vdots \\ x_k \end{bmatrix}$, based on hypothesis

$$\begin{bmatrix} x_1\lambda \\ \vdots \\ x_k\lambda \\ 0 \end{bmatrix} \text{ is a Coeffient of } x = \begin{bmatrix} x_1 \\ \vdots \\ x_k \\ x_{k+1} \end{bmatrix}.$$

Now we just need to show, $\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ x_{k+1}\lambda \end{bmatrix} + \begin{bmatrix} x_1 x_{k+1} y_{k+1} \\ x_r x_{k+1} y_{k+1} \\ \vdots \\ x_{k+1}^2 y_{k+1} \end{bmatrix}$

is a Coefficient of x.

$$\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ x_{k+1}\lambda \end{bmatrix} + \begin{bmatrix} x_1 x_{k+1} y_{k+1} \\ x_r x_{k+1} y_{k+1} \\ \vdots \\ x_{k+1}^2 y_{k+1} \end{bmatrix} = \begin{bmatrix} x_1 x_{k+1} y_{k+1} \\ x_r x_{k+1} y_{k+1} \\ \vdots \\ x_{k+1}(\lambda + x_{k+1} y_{k+1}) \end{bmatrix} = \lambda' \begin{bmatrix} x_1 \\ x_r \\ \vdots \\ x_{k+1} \end{bmatrix}$$

and we are done, because $Ay = \lambda' x$. ✓

3. suppose that $A \in \mathbb{R}^{3\times 3}$ such that

$R(A) = \text{Span}\left\{\begin{pmatrix}1\\2\\3\end{pmatrix}, \begin{pmatrix}1\\-1\\2\end{pmatrix}\right\}$ and $\text{null}(A) = 1$.

Consider the system $Ax = b$, where $b = \begin{pmatrix}1\\-7\\0\end{pmatrix}$

a) explain why $Ax = b$ has a solution.
$Ax = b$ So we will have:

$$2\times\begin{cases} x_1 + x_2 = 1 \\ 2x_1 - x_2 = -7 \\ 3x_1 + 2x_2 = 0 \end{cases} \Rightarrow \begin{array}{l} 3x_1 = -6 \Rightarrow x_1 = -2 \\ x_2 = 3 \end{array}$$

Because we have 3 equations and 2 Variables, so by solving the above equation we will a find a solution.

5/

3-b) Explain why $Ax=b$ Can not have a uniqe solution.

nulity (A) is 1. By definition, dim $(N(A))=1>0$. Hence $N(A)$ Consists of many vectors. Hence a translation of $N(A)$ Consists of many vectors. Here we will Prove one Preposition.

preposition: Suppose x_p is any Particular solution of the inhemogeneous system $Ax=b$. Then the set of solutions of the system $Ax=b$ Consists of all vectors of the form x_p+x_h, where x_h is a solution of $Ax=0$.

Proof: First we show that every vector of the form x_p+x_h is a Solution of $Ax=b$.

$$A(x_p+x_h)=Ax_p+Ax_h=b+0=b. \checkmark$$

So x_p+x_h is a solution of $Ax=b$. Next we show that every Solution of $Ax=b$ equals x_p+x_h for Some Solutions x_h for $Ax=0$. Suppose x is any solution of $Ax=b$. Then

$$A(x-x_p)=Ax-Ax_p=b-b=0$$

So $x-x_p$ is a Solution of $Ax=0$. Call this solution x_h.

Then $x-x_p=x_h$, So $x=x_p+x_h$.

Now by above preposition, if there exists a solution x to $Ax=b$, then the set of all solutions Consists of many solutions, So the Solution x is not unique.

3-b) Continue of the answer.
As we obtained in part (a), $x_p = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} -2 \\ 3 \\ 0 \end{pmatrix}$

Now we want to obtain x_h, that is the solution of $Ax=0$.

Because $null(A)=1$, there exists a vector $\alpha = \langle \alpha_1, \alpha_2, \alpha_3 \rangle$
suppose x is any solution of A.

$$\underbrace{A(x-x_p)}_{x_h} = Ax - A\begin{pmatrix} -2 \\ 3 \\ 0 \end{pmatrix} = b - b = 0$$

So $x-x_p$ is a solution. let $x_h = x - x_p$. So $x = x_p + x_h$
and for different coefficient of α, we will get the different x_h and so different x. As a result the solution is not unique.

3-C) For a general $A=(a_1, a_2, a_3) \in \mathbb{R}^{3\times 3}$ give necessary and sufficient conditions for the columns of A so that $N(A) = \mathrm{span}\left\{\begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}\right\}$

$$N(A) = \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix} \Rightarrow \begin{matrix} x_1 = -2x_2 \\ x_2 = x_2 \\ x_3 = 0 \end{matrix} \Rightarrow \begin{cases} x_1 + 2x_2 = 0 \\ x_2 = x_2 \\ x_3 = 0 \end{cases}$$

$\lambda \in \mathbb{R}$

$$A = \begin{bmatrix} | & | & | \\ a_1 & a_2 & a_3 \\ | & | & | \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} -2x_2 \\ x_2 \\ 0 \end{pmatrix} \Rightarrow A_* = \begin{pmatrix} 0 & -2\lambda & 0 \\ 0 & \lambda & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

as we have $\mathrm{rank}(A) + \mathrm{null}(A) = \dim(A)$

we get that $\mathrm{rank}(A) = 2$. So there exists α and β that $\mathrm{rank}(A) = \mathrm{span}\langle \alpha, \beta \rangle \Rightarrow x_1\alpha + x_2\beta = b$. now by considering A_*

we see that $\mathrm{rank}(A_*) = 1$, so we need to add another condition to satisfy $\mathrm{rank}(A) = 2$.

3-C Continue of answer:

So we have: $\begin{bmatrix} c & -2\lambda & 0 \\ -c & \lambda & 0 \\ 0 & 0 & 0 \end{bmatrix}$ or $\begin{bmatrix} 0 & -2\lambda & c \\ 0 & \lambda & -c \\ 0 & 0 & 0 \end{bmatrix}$

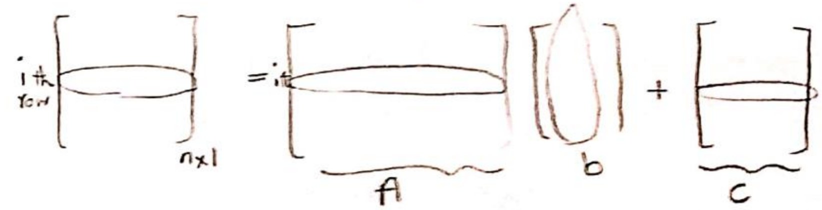
or $\begin{bmatrix} -c & -2\lambda & 0 \\ c & \lambda & 0 \\ 0 & 0 & 0 \end{bmatrix}$ or $\begin{bmatrix} 0 & -2\lambda & -c \\ 0 & \lambda & c \\ 0 & 0 & 0 \end{bmatrix}$

that λ and $c \in \mathbb{R}$.

4) Run the given Matlab script function "Computation_of_d" which computes d in two different ways, and times them using matlab's tic, toc build-in functions. Explain mathematically what d is and why the two methods of computing d gives different times.

in the first matlab code d is calculated as follows. at first d is $n \times 1$ matrix of zeroes. for i in n iterations, in each step the ith row of d is updated. $d = [\vdots]$ $d(i) = A(i,:) * b + \underbrace{c(i)}_{\text{number}}$

the ith row of d is the multiplication of ith row of A in whole b and adding the ith row of C.



=> d(i) is one number

For example let n = 3 . lets assume $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$, $b = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$

$c = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$, $d = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

step 1: of iteration $d(1) = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + 0 = 1+4+9+0 = 14$

d is updated to $d = \begin{bmatrix} 14 \\ 0 \\ 0 \end{bmatrix}$

step 2 of iteration: $d(2) = \begin{bmatrix} 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + 1 = 4+10+18+1$
$= 33$

d is updated to $\begin{bmatrix} 14 \\ 33 \\ 0 \end{bmatrix} = d$

step 3: $d(3) = \begin{bmatrix} 7 & 8 & 9 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} + 2 = 7+16+27+2 = 52$

d is updated to $\begin{bmatrix} 14 \\ 33 \\ 52 \end{bmatrix}$ and done.

For the first algorithm, the running of the algorithm is as follows:

2 $\boxed{\text{For loops}}$ we have, one for m iterations and for n iterations

So running time is $O(nm)$. in each step of multiplication of $\begin{bmatrix} 1 & \cdots & n \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ n \end{bmatrix}$, the running time is $O(n^2)$

in adding of each step running time is $O(1)$

So total running time is $O(n^2 + mn + 1) \xrightarrow{n > m} O(n^2)$

in the second matlab code d is calculated as follows,

at first d is $n\times1$ Matrix which is equals to C.

for j in n iterations, in each step the whole matrix of d is constructed by multiplication of jth column and jth row of b and adding the previous d.

$$d = C$$

$$d = A(:,j) * b(j) + d$$

the answer is one Matrix

For example let n=3. lets assume A, b, C as previous example

Step j=1 $C = d = \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix}$ is updated to

$$d = \begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix} \times 1 + \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 9 \end{bmatrix}$$

Step j=2

$$d = \begin{bmatrix} 2 \\ 5 \\ 8 \end{bmatrix} \times 2 + \begin{bmatrix} 1 \\ 5 \\ 9 \end{bmatrix} = \begin{bmatrix} 5 \\ 15 \\ 25 \end{bmatrix}$$

step j=3

$$d = \begin{bmatrix} 3 \\ 6 \\ 9 \end{bmatrix} \times 3 + \begin{bmatrix} 5 \\ 15 \\ 25 \end{bmatrix} = \begin{bmatrix} 14 \\ 33 \\ 52 \end{bmatrix}$$ and done.

d is the same as previous algorithm. now we are calculating the running time of this algorithm.

2 $\boxed{\text{for loops}}$ we have, one for m iterations and for n

10/ iterations, so running time is $O(mn)$. in each step of multiplication $\begin{bmatrix} 1 \\ \vdots \\ n \end{bmatrix} \times \underbrace{\square}_{\text{Constant number}}$ $\Rightarrow$ the running time of

multiplication is $O(n)$

in adding of each step running time $O(1)$

So total running time is $O(nm + n + 1) \rightsquigarrow O(mn)$

because $n > m \Rightarrow O(n^2) > O(mn)$

So the first algorithm running time is biger than the second one.

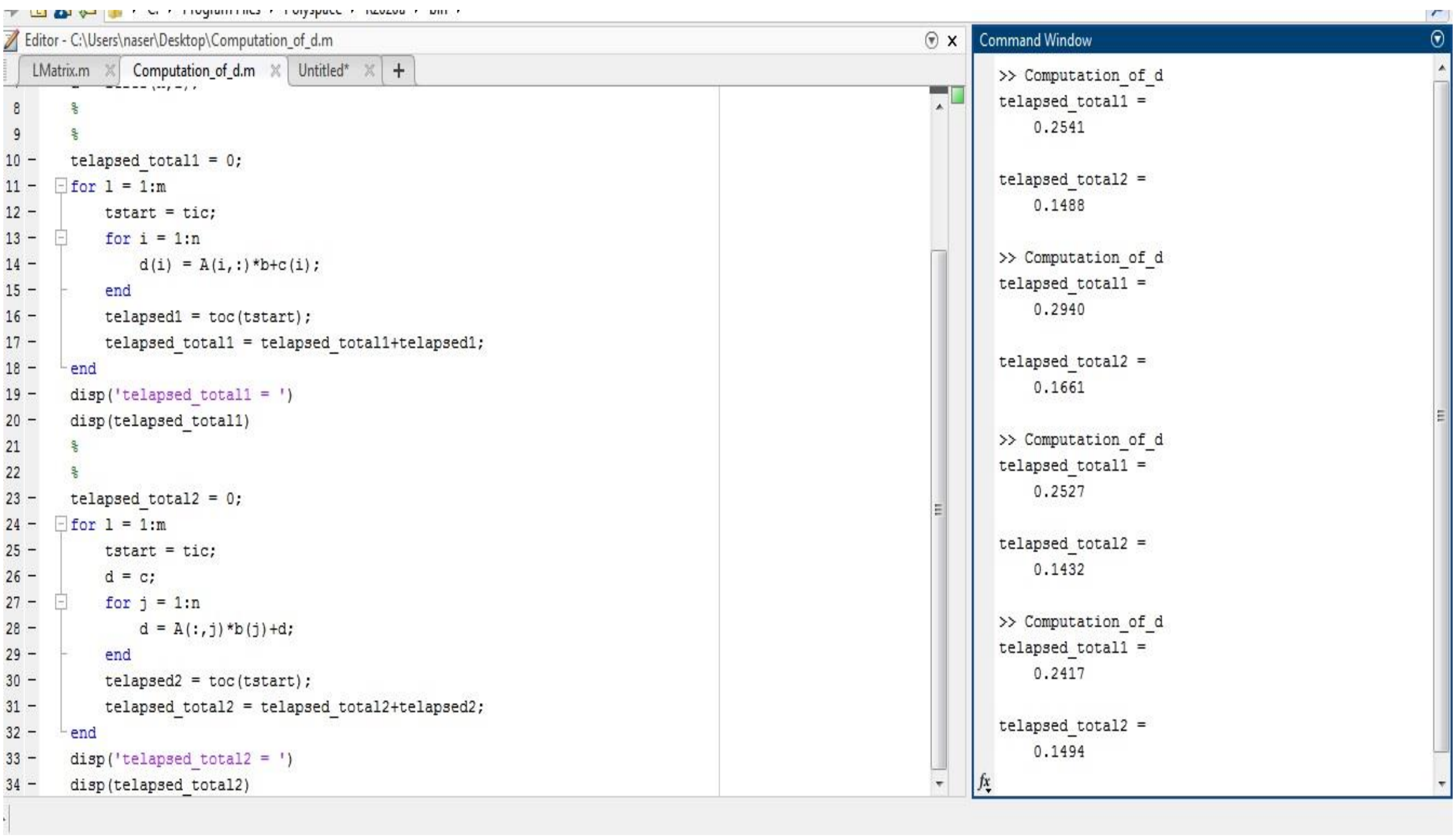
Question 2 the MATLAB code for random x and y is as follows: Ay is parallel to x for every y. The explanation is in the handwriting part.

LMatrix.m | Computation_of_d.m | Ay_x.m

```
%
n = 3;
x = randn(n,1);
y = randn(n,1);
z = x.' ;
disp("vector of x : ");
disp(x);
disp("vector of x transposed : ");
disp(z);
disp("vector of y : ");
disp(y);
A = x*z ;
disp("matrix of A : ");
disp(A);
w = A*y;
disp("the A multiply by y : ");
disp(w);
s = zeros(n,1);
for i= 1:n
    s(i) = w(i)/x(i);
end
lambda = s(i);
disp("The constant that shows A*y is parallel to x");
disp(lambda);
%
```

```
>> Ay_x
vector of x :
    0.0259
    1.2979
    1.7506

vector of x transposed :
    0.0259    1.2979    1.7506

vector of y :
    1.6461
    0.7991
   -1.7722

matrix of A :
    0.0007    0.0336    0.0453
    0.0336    1.6845    2.2721
    0.0453    2.2721    3.0646

the A multiply by y :
   -0.0523
   -2.6252
   -3.5410

The constant that shows A*y is parallel to x
   -2.0227
```

Question 4 MATLAB code was given. Just we can observe the different running time of calculating of d as follows:



Question 5 MATLAB code is as follows: A is random matrix that by using “tril(A)” we make it as a lower triangular matrix called “L”. b is the answer of equation of Lx=b. Then we use the recursive function rec(A,b) for calculating the unknown x. The procedure of function: in the first row of matrix A, all the elements are zero except the first one, so we calculate first element of x called “x_1”, by dividing first row and column element of A by first element of b, in the second row of matrix A, we will have two non-zero elements, by using previous step we use x_1 and find the second row element of x. $x_2 = \frac{b_2 - a_{21}x_1}{a_{22}}$ and recursively finding the other unknown elements of x.

Editor - C:\Users\naser\Desktop\LMatrix.m

LMatrix.m | Computation_of_d.m | Untitled*

```matlab
%
n = 7;
A = randn(n);
b = randn(n,1);
c = tril(A);
disp("lower triangular matrix L:");
disp(c);
disp("The vector of b: ")
disp(b)
x = rec(c,b);
disp("answer of Lx=b: ")
disp(x);
%
function x = rec(A,b)
n = length(b);
x = zeros(n, 1);
if n == 1
    x = b/A;
else
    x(1) = b(1)/A(1,1);
    b(2:n) = b(2:n) - A(2:n,1)*x(1);
    x(2:n) = rec(A(2:n, 2:n),b(2:n));
end
end
```

Command Window

```
>> LMatrix
lower triangular matrix L:
   -0.6122         0         0         0         0         0         0
   -0.2915    0.0794         0         0         0         0         0
    0.0504   -0.1168   -0.4196         0         0         0         0
   -0.8699    0.8953    0.4657    0.3341         0         0         0
   -0.0950    3.3636   -0.4216    0.3537    0.9236         0         0
    1.1313   -1.0014   -1.5528   -1.2367   -0.6006   -0.6410         0
   -1.0097   -0.4585   -0.7089   -0.3652   -1.3188    0.0155    1.5328

The vector of b:
   -0.8930
   -0.3301
    0.3325
   -0.0587
   -0.5682
    1.8852
    0.1209

answer of Lx=b:
    1.4586
    1.1969
   -0.9505
    1.7396
   -5.9241
    2.2600
   -3.7472
```

UTF-8 | script | Ln 25 Col 1