Chapter 7

Dimension and Structure

7.1 Basis and Dimension

Exercise 7.1. (*Linear Combination and Independence*)
Are any of the vectors in the set

$$S = \{(2,6,3,4,2),\ (3,1,5,8,3),\ (5,1,2,6,7),\ (8,4,3,2,6),\ (5,5,6,3,4)\}$$

linear combinations of predecessors? Justify your answer.

Solution. One strategy is to form a matrix V of the column vectors $\mathbf{v}_k$ mentioned above and decide whether the system $V\mathbf{x} = \mathbf{0}$ has nontrivial solutions. If so, then at least one column is a linear combination of previous ones. Otherwise, the columns are linearly independent.

```
v1 = [2 6 3 4 2]'; v2 = [3 1 5 8 3]'; v3 = [5 1 2 6 7]';
v4 = [8 4 3 2 6]'; v5 = [5 5 6 3 4]';

% Construct V of the column vectors v1,v2,v3,v4 and v5.
V = [v1 v2 v3 v4 v5];

format short;

% Find the reduced row echelon form of V.
rref_V = rref(V);

disp('The reduced row echelon form of A is'); disp(rref_V);
```

MATLAB results.

```
The reduced row echelon form of A is
     1     0     0     0     0
     0     1     0     0     0
```

```
     0     0     1     0     0
     0     0     0     1     0
     0     0     0     0     1
```

Since the reduced row echelon form of V has 5 pivots, the columns of V are linearly independent. Hence, no column of V can be a linear combination of any other columns.

7.2 Properties of Bases

Exercise 7.2. In this problem, we make a function file `CheckBasis.m` to check that the vectors $\mathbf{v}_1$, $\mathbf{v}_2$, $\mathbf{v}_3$ and $\mathbf{v}_4$ form a basis of $\mathbb{R}^4$ using the equivalent statements (a), (g), (h), and (o) of Theorem 7.2.7 in the textbook.

(a) Complete the shadow part (/////) of the m-file given below referring to the comments and the execution results.

```
%--- your function file ---%
function [Result]=CheckBasis(v1, v2, v3, v4, case_num)
  %  if case_num=1, check the statement (a),
  %  if case_num=2, check the statement (g),
  %  if case_num=3, check the statement (h).

  % Construct the matrix V.
  ////////////////////////

  % Use the switch statement to check
  % whether one of the statements (a), (g), and (h) holds.
  switch case_num
    case 1
      fprintf('* You enter %d: statement (a) *\n', case_num);
      ////////////////////////
      if ////////////////////////
        disp('Given vectors form a basis of 4 dimensional space.');
      else
        disp('Given vectors do not form a basis of 4 dimensional space.');
      end
    case 2
      fprintf('* You enter %d: statement (g) *\n', case_num);
      Result=det(V);
      if Result~=0
        disp('Given vectors form a basis of 4 dimensional space.');
      else
        disp('Given vectors do not form a basis of 4 dimensional space.');
```

```
      end
    ///////////////////////////  % check statement (h)
        ////////////////////////
        ////////////////////////
        ////////////////////////
        ////////////////////////
        ////////////////////////
        ////////////////////////
        ////////////////////////
  end
end
```

The execution results will be as follows:

```
>> v1=[1 0 0 0]'; v2=[0 2 0 0]'; v3=[0 0 4 5]'; v4=[0 0 0 -1]'; v5=[0 0 0 1]';
>> C=CheckBasis(v1, v2, v3, v4,3)
* You enter 3: statement (h) *
  Given vectors are basis of 4 dimensional space.

C =

     1
     2
    -1
     4

>> CheckBasis(v1, v2, v4, v5, 1);
* You enter 1: statement (a) *
  Given vectors do not form a basis of 4 dimensional space.

>> determinant=CheckBasis(v1, v2, v3, v5, 2)
* You enter 2: statement (g) *
  Given vectors form a basis of 4 dimensional space.

determinant =
     8
```

(b) Using `CheckBasis.m` from (a), check whether

i. $\mathbf{v}_1 = (-1, 0, 1, 0)^T$, $\mathbf{v}_2 = (2, 3, -2, 6)^T$, $\mathbf{v}_3 = (0, -1, 2, 0)^T$ and $\mathbf{v}_4 = (0, 0, 1, 5)^T$ form a basis of $\mathbb{R}^4$.

ii. $\mathbf{v}_1 = (a, b, c, d)^T$, $\mathbf{v}_2 = (-b, a, d, -c)^T$, $\mathbf{v}_3 = (-c, -d, a, b)^T$ and $\mathbf{v}_4 = (-d, c, -b, a)^T$ form a basis of $\mathbb{R}^4$.

Solution.

(a)
```
% ----- your function file ----- %
function [Result]=CheckBasis(v1, v2, v3, v4, case_num)
  %  if case_num=1, check the statement (a),
  %  if case_num=2, check the statement (g),
  %  if case_num=3, check the statement (h).

 % Construct the matrix V.
 V=[v1 v2 v3 v4];


  % Use the switch statement to check
  % whether one of the statements (a), (g), and (h) holds.
  switch case_num
    case 1
      fprintf('* You enter %d: statement (a) *', case_num);
      Result=rref(V)
      if det(Result)~=0
        disp('Given vectors form a basis of 4 dimensional space.');
      else
        disp('Given vectors do not form a basis of 4 dimensional space.');
      end
    case 2
      fprintf('* You enter %d: statement (g) *', case_num);
      Result=det(V);
      if Result~=0
        disp('Given vectors form a basis of 4 dimensional space.');
      else
        disp('Given vectors do not form a basis of 4 dimensional space.');
      end
    case 3  % check statement (h)
      [Q D]=eig(V);
      Result=diag(R);
      if det(R)==0
        disp('Given vectors form a basis of 4 dimensional space.');
      else
        disp('Given vectors do not form a basis of 4 dimensional
space.');
      end
    end
```

(b)-i.
```
>> v1=[-1 0 1 0]'; v2=[2 3 -2 6]';  v3=[0 -1 2 0]'; v4 = [0 0 1 5]';
>> CheckBasis(v1, v2, v3, v4, 1);
>> CheckBasis(v1, v2, v3, v4, 2);
>> CheckBasis(v1, v2, v3, v4, 3);
```

MATLAB results.

```
* You enter 1: statement (a) *
  Given vectors form a basis of 4 dimensional space.
* You enter 2: statement (g) *
  Given vectors form a basis of 4 dimensional space.
* You enter 3: statement (h) *
  Given vectors form a basis of 4 dimensional space.
```

(b)-ii.

```
>> syms a b c d;
>> v1=[a;b;c;d]; v2=[-b;a;d;-c]; v3=[-c;-d;a;b]; v4 = [-d;c;-b;a];
>> CheckBasis(v1, v2, v3, v4, 1);
>> CheckBasis(v1, v2, v3, v4, 2);
```

MATLAB results.

```
* You enter 1: statement (a) *
  Given vectors form a basis of 4 dimensional space.
* You enter 2: statement (g) *
  Given vectors form a basis of 4 dimensional space.
```

7.3 The Fundamental Spaces of a Matrix

Exercise 7.3. In this problem, we make a function file `getFSinfo.m` to get the dimension and basis of the fundamental spaces of a given matrix. For example, we execute the followings:

```
>> A=[1 0 0 0 2; -2 1 -3 -2 -4; 0 5 -14 -9 0; 2 10 -28 -18 4];
>> getFSinfo(A);
```

Then, the Command Window displays the results as follows:

```
Given matrix is:
     1     0     0     0     2
    -2     1    -3    -2    -4
     0     5   -14    -9     0
     2    10   -28   -18     4

== Dimension of the fundamental spaces of a given matrix ==
dim(row(A))=dim(col(A)): 3,   dim(null(A)): 2,   dim(null(A_trans)): 1

== Basis of the fundamental spaces of a given matrix (in row vectors) ==
 row(A)
     1     0     0     0     2
     0     1     0     1     0
     0     0     1     1     0

 col(A)
     1     0     0     2
```

```
    0     1     0     0
    0     0     1     2

null(A)
    0    -1    -1     1     0
   -2     0     0     0     1

null(A_trans)
   -2     0    -2     1

*****************************************************
```

(a) Complete the missing parts of the m-file `getFSinfo` given as follows:

```
%--- function file 'getFSinfo.m' ---%
function [info]=getFSinfo(A)
  % row(A): basis and dimension
  /////// missing part ///////

  % col(A): basis and dimension
  /////// missing part ///////

  % null(A): basis and dimension
  /////// missing part ///////

  % null(A'): basis and dimension
  /////// missing part ///////

  disp('Given matrix is:'); disp(A);
  fprintf('== Dimension of the fundamental spaces of given matrix == \n');
  fprintf('dim(row(A))=dim(col(A)): %d,', rank_A);
  fprintf('\t dim(null(A)): %d,\t dim(null(A_trans)): %d \n\n', nullity, nullity_T
  fprintf('== Basis of the fundamental spaces of given matrix (in row vectors) ==
  disp(' row(A)'); disp(double(rowA_basis));
  disp(' col(A)'); disp(double(colA_basis));
  disp(' null(A)'); disp(nullA_basis);
  disp(' null(A_trans)'); disp(nullAtrans_basis);
  fprintf('\n*****************************************************\n');
end
```

You may use the MATLAB commands *rank, colspace, rref, null* and so on.

(b) Using your function file `getFSinfo.m`, find the dimension and basis of the fundamental spaces of

$$A = \begin{bmatrix} 3 & 2 & 1 & 3 & 5 \\ 6 & 4 & 3 & 5 & 7 \\ 9 & 6 & 5 & 7 & 9 \\ 3 & 2 & 0 & 4 & 8 \end{bmatrix}, B = \begin{bmatrix} 3 & -1 & 3 & 2 & 5 \\ 5 & -3 & 2 & 3 & 4 \\ 1 & -3 & -5 & 0 & -7 \\ 7 & -5 & 1 & 4 & 1 \end{bmatrix}, C = \begin{bmatrix} 1 & 3 & 2 & 1 \\ -2 & -6 & 0 & -6 \\ 3 & 9 & 1 & 8 \\ -1 & -3 & -3 & -6 \\ 1 & 3 & 2 & 1 \\ 4 & 12 & 1 & 11 \end{bmatrix}.$$

Solution.

(a)
```
% ----- function file 'getFSinfo.m' ----- %
function [info]=getFSinfo(A)
  [m,n]=size(A)
  % row(A): basis and dimension
    rank_A=rank(A);    % rank of A;
    rowA=colspace(sym(A')); % find the row basis
    rowA_basis=rowA(:, 1:rank_A)';  % basis of row(A)
  % col(A): basis and dimension
    colA=colspace(sym(A)); % find the column basis
    colA_basis=colA(:, 1:rank_A)';  % basis of col(A)
  % null(A): basis and dimension
    nullA=null(A, 'r');
    nullity=n-rank_A;   % using Dimension theorem
    nullA_basis=nullA(:, 1:nullity)';
  % null(A'): basis and dimension
    nullAtrans=null(A', 'r');
    nullity_T=m-rank_A;
    nullAtrans_basis=nullAtrans(:,1:nullity_T)';

  disp('Given matrix is:'); disp(A);
  fprintf('== Dimension of the fundamental spaces of given matrix == \n');
  fprintf('dim(row(A))=dim(col(A)): %d,', rank_A);
  fprintf('\t dim(null(A)): %d,', nullity);
  fprintf('\t dim(null(A_trans)): %d \n\n', nullity_T);
  fprintf('== Basis of the fundamental spaces ');
  fprintf('of given matrix (in row vectors) == \n');
  disp(' row(A)'); disp(double(rowA_basis));
  disp(' col(A)'); disp(double(colA_basis));
  disp(' null(A)'); disp(nullA_basis);
  disp(' null(A_trans)'); disp(nullAtrans_basis);
  fprintf('\n*****************************************************\n');
end
```

(b)
```
A=[3 2 1 3 5; 6 4 3 5 7; 9 6 5 7 9; 3 2 0 4 8];
B=[3 -1 3 2 5; 5 -3 2 3 4; 1 -3 -5 0 -7; 7 -5 1 4 1];
C=[1 3 2 1; -2 -6 0 -6 ;3 9 1 8; -1 -3 -3 -6; 1 3 2 1; 4 12 1 11];
getFSinfo(A);
getFSinfo(B);
getFSinfo(C);
```

MATLAB results.

```
Given matrix is:
     3     2     1     3     5
     6     4     3     5     7
     9     6     5     7     9
     3     2     0     4     8
== Dimension of the fundamental spaces of given matrix ==
dim(row(A))=dim(col(A)): 2, dim(null(A)): 3, dim(null(A_trans)): 2
== Basis of the fundamental spaces of given matrix (in row vectors) ==
 row(A)
    1.0000    0.6667         0    1.3333    2.6667
         0         0    1.0000   -1.0000   -3.0000
 col(A)
     1     0    -1     3
     0     1     2    -1
 null(A)
   -0.6667    1.0000         0         0         0
   -1.3333         0    1.0000    1.0000         0
   -2.6667         0    3.0000         0    1.0000
 null(A_trans)
     1    -2     1     0
    -3     1     0     1
*****************************************************

Given matrix is:
     3    -1     3     2     5
     5    -3     2     3     4
     1    -3    -5     0    -7
     7    -5     1     4     1
== Dimension of the fundamental spaces of given matrix ==
dim(row(A))=dim(col(A)): 3, dim(null(A)): 2, dim(null(A_trans)): 1
== Basis of the fundamental spaces of given matrix (in row vectors) ==
 row(A)
    1.0000         0    1.7500    0.7500         0
         0    1.0000    2.2500    0.2500         0
         0         0         0         0    1.0000
 col(A)
     1     0    -3     0
     0     1     2     0
```

```
     0     0     0     1
 null(A)
   -1.7500   -2.2500    1.0000         0         0
   -0.7500   -0.2500         0    1.0000         0
 null(A_trans)
     3    -2     1     0
*****************************************************

Given matrix is:
     1     3     2     1
    -2    -6     0    -6
     3     9     1     8
    -1    -3    -3    -6
     1     3     2     1
     4    12     1    11
== Dimension of the fundamental spaces of given matrix ==
dim(row(A))=dim(col(A)): 3, dim(null(A)): 1, dim(null(A_trans)): 3
== Basis of the fundamental spaces of given matrix (in row vectors) ==
 row(A)
     1     3     0     0
     0     0     1     0
     0     0     0     1
 col(A)
    1.0000         0    0.5000         0    1.0000    0.5000
         0    1.0000   -1.2500         0         0   -1.7500
         0         0         0    1.0000         0         0
 null(A)
    -3     1     0     0
 null(A_trans)
   -0.5000    1.2500    1.0000         0         0         0
   -1.0000         0         0         0    1.0000         0
   -0.5000    1.7500         0         0         0    1.0000
*****************************************************
```

Exercise 7.4. (*Bases for the Fundamental Spaces*)

(a) Use the MATLAB commands *sym* and *colspace* to find a basis for the column space of the matrix

$$A = \begin{bmatrix} 2 & -1 & 3 & 5 \\ 4 & -3 & 1 & 3 \\ 3 & -2 & 3 & 4 \\ 4 & -1 & 15 & 17 \\ 7 & -6 & -7 & 0 \end{bmatrix}.$$

(b) Use the same MATLAB commands in (a) to find a basis for the row space of A.

(c) Confirm that the basis obtained in (b) is consistent with the basis obtained from the reduced row echelon form of A.

(d) Tell what happens if you use the MATLAB command *orth*?

Solution.

(a)
```
% Set a matrix A whose entries are symbolic objects.
A = sym([2 -1 3 5; 4 -3 1 3; 3 -2 3 4; 4 -1 15 17; 7 -6 -7 0]);

% Find a basis for the column space of A.
col_basis = colspace(A);

disp('A basis for the column space of A is');
disp(col_basis(:,1)'); disp(col_basis(:,2)'); disp(col_basis(:,3)');
```

MATLAB results.

```
A basis for the column space of A is
[ 1, 0, 0, 2, 1]

[ 0, 1, 0, -3, 5]

[ 0, 0, 1, 4, -5]
```

(b)
```
% Set a matrix A_transpose whose entries are symbolic objects.
A_transpose = sym([2 -1 3 5; 4 -3 1 3; 3 -2 3 4; 4 -1 15 17; 7 -6 -7 0]');

% Finding a basis for the row space of A is equivalent to
% finding a basis for the column space of A_transpose.
rowbasis = colspace(A_transpose);

disp('A basis for the row space of A is');
disp(rowbasis(:,1)'); disp(rowbasis(:,2)'); disp(rowbasis(:,3)');
```

MATLAB results.

```
A basis for the row space of A is
[ 1, 0, 0, 6]

[ 0, 1, 0, 7]

[ 0, 0, 1, 0]
```

(c)
```
% Set a matrix A.
A = [2 -1 3 5; 4 -3 1 3; 3 -2 3 4; 4 -1 15 17; 7 -6 -7 0];

% Find the reduced row echelon form of A.
```

```
rref_A = rref(A);

% The nonzero rows of the reduced row echelon form of A
% form a basis for the row space of A.

disp('A basis for the row space of A is');
disp(rref_A(1,:)); disp(rref_A(2,:)); disp(rref_A(3,:));
```

MATLAB results.

```
A basis for the row space of A is
     1     0     0     6

     0     1     0     7

     0     0     1     0
```

(d)
```
% Set A.
A = [2 -1 3 5; 4 -3 1 3; 3 -2 3 4; 4 -1 15 17; 7 -6 -7 0];

% The command orth gives an orthonormal basis for the column space of A.
B = orth(A);

disp('An orthonormal basis for the column space of A is');
disp('q1='); disp(B(:,1)');
disp('q2='); disp(B(:,2)');
disp('q3='); disp(B(:,3)');
```

MATLAB results.

```
An orthonormal basis for the column space of A is
q1=
   -0.2427   -0.1508   -0.2229   -0.9246    0.1177

q2=
   -0.1189   -0.3624   -0.2060    0.0253   -0.9008

q3=
    0.3760   -0.6016   -0.5930    0.1848    0.3331
```