National University of Singapore
MA2001 Linear Algebra

MATLAB Worksheet 5
Dot Product, Orthogonal Sets and Least Squares Solutions

A. Dot Product and Norm

Let $\boldsymbol{u} = (u_1, \ldots, u_n)$ and $\boldsymbol{v} = (v_1, \ldots, v_n)$ be vectors in $\mathbb{R}^n$. Their **dot product** is defined by

$$\boldsymbol{u} \cdot \boldsymbol{v} = u_1 v_1 + \cdots + u_n v_n = \begin{pmatrix} u_1 & \cdots & u_n \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix}.$$

If both $\boldsymbol{u}$ and $\boldsymbol{v}$ are defined as row vectors, then $\boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}\boldsymbol{v}^{\mathrm{T}}$.
If both $\boldsymbol{u}$ and $\boldsymbol{v}$ are defined as column vectors, then $\boldsymbol{u} \cdot \boldsymbol{v} = \boldsymbol{u}^{\mathrm{T}}\boldsymbol{v}$.
For example, let $\boldsymbol{u} = (1, 2, 3, 4, 5)$ and $\boldsymbol{v} = (1, 0, 1, -1, 2)$.

```
>>  u = [1 2 3 4 5];  v = [1 0 1 -1 2];
>>  u * v'
ans =  10
```

Alternatively, MATLAB provides a command `dot` for dot product, regardless whether the vectors are defined as row or column vectors.

```
>>  dot(u, v)
ans =  10
>>  dot(u', v)
ans =  10
>>  dot(u, v')
ans =  10
>>  dot(u', v')
ans =  10
```

The **norm** of a vector $\boldsymbol{v} = (v_1, \ldots, v_n)$ in $\mathbb{R}^n$ is defined by

$$\|\boldsymbol{v}\| = \sqrt{v_1^2 + \cdots + v_n^2} = \sqrt{\boldsymbol{v} \cdot \boldsymbol{v}}.$$

Using the vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ above, their norms can be evaluated using the `sqrt` and `dot` commands. We shall use the default format in MALTAB.

```
>>  format short
>>  sqrt(dot(u, u))
ans =  7.4162
>>  sqrt(dot(v, v))
ans =  2.6458
```

Alternatively, in MATLAB `norm` can be used to find the norm of a vector.

```
>>  norm(u)
ans =  7.4162
```

Note that the norm of a vector is usually irrational (because of the square root), and the output is in floating-point. We can use `sym` to define a vector as **symbolic** object, and use `norm` to get the exact value of the norm. For example,

```
>> u = sym([1 2 3 4 5])
u =  [1, 2, 3, 4, 5]
>> norm(u)
ans =  55^(1/2)
```

B. Orthogonal Sets

A set of vectors $S = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ in $\mathbb{R}^n$ is said to be an **orthogonal** set if

$$\boldsymbol{v}_i \cdot \boldsymbol{v}_j = 0 \quad \text{for all } i \neq j.$$

View each vector as a row vector and consider $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_k \end{pmatrix}$. Then $\boldsymbol{A}^{\mathrm{T}} = \begin{pmatrix} \boldsymbol{v}_1^{\mathrm{T}} & \cdots & \boldsymbol{v}_k^{\mathrm{T}} \end{pmatrix}$ and

$$\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}} = \begin{pmatrix} \boldsymbol{v}_1\boldsymbol{v}_1^{\mathrm{T}} & \cdots & \boldsymbol{v}_1\boldsymbol{v}_k^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ \boldsymbol{v}_k\boldsymbol{v}_1^{\mathrm{T}} & \cdots & \boldsymbol{v}_k\boldsymbol{v}_k^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{v}_1 \cdot \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_1 \cdot \boldsymbol{v}_k \\ \vdots & \ddots & \vdots \\ \boldsymbol{v}_k \cdot \boldsymbol{v}_1 & \cdots & \boldsymbol{v}_k \cdot \boldsymbol{v}_k \end{pmatrix}.$$

Hence, to determine whether the set $S = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ is orthogonal, we can use MATLAB to check whether $\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}$ is a diagonal matrix.

For example, consider a set of vectors

$$\{(1, 1, 1, 1), (1, 0, -1, 0), (1, -1, 1, -1)\}$$

in $\mathbb{R}^4$.

(i) Define the matrix $\boldsymbol{C}$ whose rows are the given vectors.

```
>> C = [1 1 1 1; 1 0 -1 0; 1 -1 1 -1];
```

(ii) Evaluate $\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}$.

```
>> C * C'
ans =  4    0    0
       0    2    0
       0    0    4
```

Since $\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}$ is a diagonal matrix, the set of vectors $\{(1, 1, 1, 1), (1, 0, -1, 0), (1, -1, 1, -1)\}$ is an orthogonal set.

A set of vectors $S = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ in $\mathbb{R}^n$ is said to be an **orthonormal** set if

$$\boldsymbol{v}_i \cdot \boldsymbol{v}_j = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j. \end{cases}$$

View each vector as a row vector and consider $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{v}_1 \\ \vdots \\ \boldsymbol{v}_k \end{pmatrix}$.

Hence, to determine whether the set $S = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k\}$ is orthonormal, we can use MATLAB to check whether $\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}} = \boldsymbol{I}_k$, the identity matrix of order k.

For example, consider a set of vectors

$$\left\{(\cos\frac{\pi}{3}, 0, \sin\frac{\pi}{3}, 0), (0,1,0,0), (\sin\frac{\pi}{3}, 0, -\cos\frac{\pi}{3}, 0)\right\}$$

in $\mathbb{R}^4$.

(i) Define the matrix $\boldsymbol{D}$ whose rows are the given vectors.

```
>>  D = [cos(pi/3) 0 sin(pi/3) 0; 0 1 0 0; sin(pi/3) 0 -cos(pi/3) 0];
```

(ii) Evaluate $\boldsymbol{D}\boldsymbol{D}^{\mathrm{T}}$.

```
>>  D * D'

ans =   1.0000         0    0.0000
             0    1.0000         0
        0.0000         0    1.0000
```

Since $\boldsymbol{D}\boldsymbol{D}^{\mathrm{T}} = \boldsymbol{I}_3$, the given set of vectors $\left\{(\cos\frac{\pi}{3}, 0, \sin\frac{\pi}{3}, 0), (0,1,0,0), (\sin\frac{\pi}{3}, 0, -\cos\frac{\pi}{3}, 0)\right\}$ is an orthonormal set.

C. Orthonormal Bases

If $V = \mathrm{span}(S)$, and S is an orthonormal set, then S is linearly independent, and S is called an **orthonormal basis** for V.
In MATLAB, `orth` can be used to get an **orthonormal basis** for the column space of a matrix.

Suppose $V = \mathrm{span}(S)$, where $S = \{(1,1,1,1), (1,1,0,0), (0,1,1,0)\}$.

(i) Define matrix $\boldsymbol{E} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}$ whose columns are the vectors in the spanning set S. By definition, $V = \mathrm{span}(S)$ is the column space of the matrix $\boldsymbol{E}$.

```
>>  E = [1 1 0; 1 1 1; 1 0 1; 1 0 0];
```

It might be easier to input in MATLAB the vectors of S as rows of a matrix and then take the transpose, as shown below.

```
>>  E = [1 1 1 1; 1 1 0 0; 0 1 1 0]';
```

(ii) Use the command `orth` to get an orthonormal basis for the column space of $\boldsymbol{E}$, i.e., for V.

```
>>  orth(E)
ans =  -0.4835     0.7071      -0.1273
       -0.6635     0.0000       0.5565
       -0.4835    -0.7071     -0.1273
       -0.3035    -0.0000     -0.8111
```

The columns of the resulting matrix give an orthonormal basis for the vector space V:

$$\{(-0.4835, -0.6635, -0.4835, -0.3035), (0.7071, 0, -0.7071, 0), (-0.1273, 0.5565, -0.1273, -0.8111)\}$$

Important note:
The (classical) Gram-Schmidt process in the textbook (Theorem 5.2.19) is numerically unstable. The MATLAB command `orth` above uses a **modified Gram-Schmidt process**[1] to generate an orthonormal basis which is generally different from the classical process.

In order to get the exact form of the orthonormal basis, we can use `sym` to define the matrix as **symbolic** object. By dealing with symbolic objects, there is no longer numerical error. In this case, `orth` uses the classical Gram-Schmidt process to generate the orthonormal basis.

Again, suppose $V = \text{span}(S)$, where $S = \{(1, 1, 1, 1), (1, 1, 0, 0), (0, 1, 1, 0)\}$.

(i) Define matrix $\boldsymbol{E}$ whose columns are the vectors in S as an symbolic object.

```
>>  E = sym([1 1 0; 1 1 1; 1 0 1; 1 0 0]);
E =  [1, 1, 0]
     [1, 1, 1]
     [1, 0, 1]
     [1, 0, 0]
```

(ii) Use `orth` to get an orthonormal basis for the column space of $\boldsymbol{E}$, i.e., for V.

```
>>  orth(E)
ans =  [1/2, 1/2, -1/2]
       [1/2, 1/2, 1/2]
       [1/2, -1/2, 1/2]
       [1/2, -1/2, -1/2]
```

[1]Your may refer to the session **Numerical Stability** on Gram-Schmidt process in WIKIPEDIA: `https://en.wikipedia.org/wiki/Gram-Schmidt_process`.

The columns of the resulting matrix give an orthonormal basis for the vector space V.

$$\{(\tfrac{1}{2},\tfrac{1}{2},\tfrac{1}{2},\tfrac{1}{2}),(\tfrac{1}{2},\tfrac{1}{2},-\tfrac{1}{2},-\tfrac{1}{2}),(-\tfrac{1}{2},\tfrac{1}{2},\tfrac{1}{2},-\tfrac{1}{2})\}$$

The command `orth` also allows us to generate an orthogonal basis (i.e. the vectors need not be of norm 1) in symbolic form by "skipping the normalization" in Gram-Schmidt process:

```
>> orth(E, 'skipnormalization')
ans =  [1, 1/2, -1/2]
       [1, 1/2, 1/2]
       [1, -1/2, 1/2]
       [1, -1/2, -1/2]
```

D. Least Squares Solutions

When a linear system $\boldsymbol{Ax} = \boldsymbol{b}$ is inconsistent, we can find its **least squares solutions**, which are the "best approximation" in place of an exact solution for the system.
Recall that the least squares solution can be found by solving the following linear system

$$\boldsymbol{A}^T\boldsymbol{Ax} = \boldsymbol{A}^T\boldsymbol{b}.$$

For example, consider the linear system

$$\begin{cases} x + 2y + z = 1 \\ x + 2y + 2z = 1 \\ 2x + 4y + z = 3 \end{cases}$$

We input the system to MATLAB as an augmented matrix `[A b]`:

```
>> A = [1 2 1; 1 2 2; 2 4 1];
>> b = [1; 1; 3];
>> rref[A b];
ans =  1    2    0    0
       0    0    1    0
       0    0    0    1
```

The RREF shows the system is inconsistent.
We proceed to solve the system $\boldsymbol{A}^T\boldsymbol{Ax} = \boldsymbol{A}^T\boldsymbol{b}$:

```
>> format rat
>> rref[A'*A A'*b];
ans =  1    2    0    18/11
       0    0    1    -4/11
       0    0    0    0
```

This system is consistent and has infinitely many solutions.
Observe that the 2nd column of the reduced row-echelon form is a pivot column. Set $x_2 = s$ to be arbitrary parameter, and solve the other variables:

$$x_1 = \frac{18}{11} - 2s, \quad x_3 = \frac{-4}{11}.$$

This gives us the least squares solutions for the original system $\boldsymbol{Ax} = \boldsymbol{b}$:

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{18}{11} - 2s \\ s \\ \frac{-4}{11} \end{pmatrix}.$$

E. Projections

In this section, we show how to use least squares solutions to find the **projection** $\boldsymbol{p}$ of a given vector $\boldsymbol{v}$ onto a vector space V. Recall that $\boldsymbol{p}$ is defined to be the unique vector such that $\boldsymbol{v} - \boldsymbol{p}$ is orthogonal to every vector in V.

Let's take $V = \text{span}\{(1,1,2),(2,2,4),(1,2,1)\}$ and $\boldsymbol{v} = (1,1,3)$. To find the projection $\boldsymbol{p}$ of $\boldsymbol{v}$ onto V, we use the spanning vectors (written in column form) of V to form the matrix $\boldsymbol{A} = \begin{pmatrix} 1 & 2 & 1 \\ 1 & 2 & 2 \\ 2 & 4 & 1 \end{pmatrix}$, and let $\boldsymbol{b} = \begin{pmatrix} 1 \\ 1 \\ 3 \end{pmatrix}$ which is the vector $\boldsymbol{v}$ in column form. Note that the column space of $\boldsymbol{A}$ is precisely V.

Recall that the projection of a vector $\boldsymbol{b}$ onto the column space of a matrix $\boldsymbol{A}$ is given by $\boldsymbol{Ax}_0$ where $\boldsymbol{x}_0$ is a least squares solution of the system $\boldsymbol{Ax} = \boldsymbol{b}$.

Note that our $\boldsymbol{A}$ and $\boldsymbol{b}$ are the examples from the previous section, and we have found the least squares solutions of the system $\boldsymbol{Ax} = \boldsymbol{b}$, given by

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} \frac{18}{11} - 2s \\ s \\ \frac{-4}{11} \end{pmatrix}.$$

Take $\boldsymbol{x}_0$ to be any particular least squares solution (say we let $s = 0$):

$$\boldsymbol{x}_0 = \begin{pmatrix} \frac{18}{11} \\ 0 \\ \frac{-4}{11} \end{pmatrix}.$$

Input this solution in MATLAB:

```
>>  x0=[18/11 0 -4/11];
```

Then the required projection $\boldsymbol{p}$ can be computed as $\boldsymbol{Ax}_0$.

```
>>  p=A*x0
p =  14/11
     10/11
     32/11
```

F. Practices

Use MATLAB to solve Questions 5.1, 5.2, 5.6, 5.10, 5.11(a), 5.12, 5.13, 5.14, 5.15, 5.17, 5.22, 5.24, 5.25, 5.26, 5.27(a) in the textbook Exercise 5.