EECS 16B

Spring 2022

Lecture 16

3/10/2022

LECTURE 16

- wrap up controllability
- orthonormal bases and Gram-Schmidt procedure

Past two lectures: $\vec{x}[i+1] = A\vec{x}[i] + Bu[i]$ (single input for simplicity)

Same condition appeared for two tasks:

1) being able to assign evalues of $A+BF$ arbitrarily by design of F (feedback design);
2) moving state from any $\vec{x}[0]$ to any $\vec{x}_{target}$ ("controllability")

Condition was: $A^{n-1}B, A^{n-2}B, \ldots AB, B$ linearly independent where n: state dimension ($A: n\times n$, $B: n\times 1$), i.e.,

$$C_n = [A^{n-1}B \; \cdots \; AB \;\; B] \text{ invertible.}$$

Example 1:

$A = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$ $B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$

Example 2:

$A = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$ $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$n=2$: AB, B linearly independent?

Example 1 — No:

$AB = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = B$

Uncontrollable

$x_1[i+1] = x_1[i] + x_2[i] + u[i]$

$x_2[i+1] = 2x_2[i]$

$\rightarrow x_2[\ell] = 2^\ell x_2[0]$

1. evalue of 2 remains regardless of feedback
2. we can't take x_2 component anywhere we want

Example 2 — Yes:

$AB = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$ indep. of $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

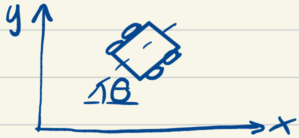
Controllable

$x_1[i+1] = x_1[i] + x_2[i]$

$x_2[i+1] = 2x_2[i] + u[i]$

u doesn't appear in x_1 eq'n but can influence x_1 indirectly through x_2

<u>Note</u> from the above that we can control n variables with fewer than n inputs – you actually do this when you drive a car: using two inputs (steering and longitudinal forces generated by gas/brake pedals) you come not only to the desired x, y coordinates but also to the desired orientation e.g. parallel parking.



$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$: state vector can be brought to target value.

Orthonormal bases and Gram-Schmidt Procedure:

Column vectors $\vec{q}_1, \ldots \vec{q}_k$ are called orthonormal if

$$\vec{q}_i^T \vec{q}_j = \begin{cases} 0 & \text{if } i \neq j \quad \text{(ortho)} \\ 1 & \text{if } i = j \quad \text{(normal)} \end{cases} \qquad --(1)$$

A matrix $Q = [\vec{q}_1 \cdots \vec{q}_k]$ with orthonormal columns satisfies:

$$Q^T Q = \begin{bmatrix} \vec{q}_1^T \\ \vdots \\ \vec{q}_k^T \end{bmatrix} [\vec{q}_1 \, \cdots \, \vec{q}_k] = \begin{bmatrix} \vec{q}_1^T \vec{q}_1 & \cdots & \vec{q}_1^T \vec{q}_k \\ \vdots & & \\ \vec{q}_k^T \vec{q}_1 & \cdots & \vec{q}_k^T \vec{q}_k \end{bmatrix}$$

$$= I_{k \times k} \text{ by def'n (1).}$$

$$\boxed{Q^T Q = I}$$

If Q is square $Q^T Q = I$ means:

$$Q^T = Q^{-1}.$$

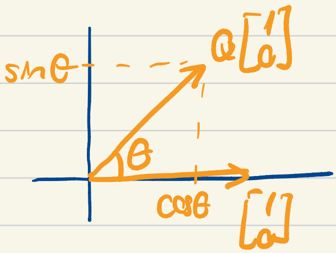
(Q is called orthogonal.)

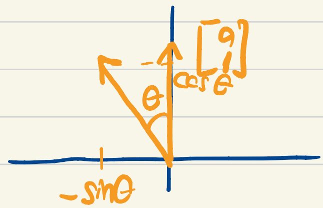
Example: $Q = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ (rotation matrix)

$$\underbrace{\begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}}_{\vec{q}_1} \quad \underbrace{\begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix}}_{\vec{q}_2}$$

$$\vec{q}_1^{\,T}\vec{q}_2 = -\cos\theta\sin\theta + \sin\theta\cos\theta = 0$$

$$\vec{q}_i^{\,T}\vec{q}_i = \cos^2\theta + \sin^2\theta = 1$$





so called because $Q\vec{x}$ rotates $\vec{x}$ by angle θ in the plane without changing its length

Useful features of matrices with orthonormal columns:

1) $\underbrace{\|Q\vec{x}\|}_{\sqrt{(Q\vec{x})^T(Q\vec{x})}} = \underbrace{\|\vec{x}\|}_{\sqrt{\vec{x}^T\vec{x}}}$ (preserves length: what we observed in example above is true for other Q with orthonormal columns)

$$= \sqrt{\vec{x}^T \underbrace{Q^TQ}_{=I} \vec{x}} = \sqrt{\vec{x}^T\vec{x}}$$

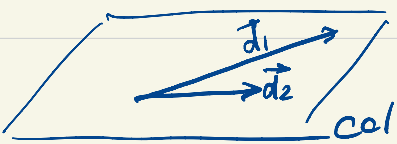
2) Q also preserves dot product: $(Q\vec{x})^T(Q\vec{y})$

$$= \vec{x}^T \underbrace{Q^TQ}_{=I} \vec{y}$$

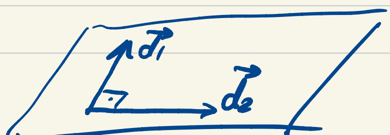
$$= \vec{x}^T\vec{y}$$

3) Easy visualization of column space:

column space of $D = [\vec{d}_1\ \vec{d}_2]$ — if $\vec{d}_1, \vec{d}_2$ were orthonormal

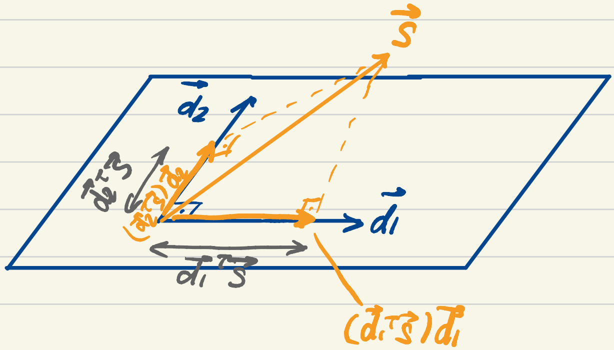


col. space of D



prettier!

and projection onto column space is trivial:



Projection of $\vec{s}$ onto column space of D:

$$= (\vec{d_1}^T\vec{s})\vec{d_1} + (\vec{d_2}^T\vec{s})\vec{d_2}$$

Recall Least-Squares:

$$\vec{s} = D\vec{p} + \vec{e}$$

$$\hat{p} = (D^TD)^{-1}D^T\vec{s}$$

What if (by some miracle) D had orthonormal columns?

$D^TD = I \Rightarrow \quad \hat{p} = D^T\vec{s} \quad$ (no matrix inversion!)

compare to picture above...

Gram-Schmidt:

Even if columns of D are not orthonormal, we can construct an orthonormal basis for the column space that is close to the original columns in the sense that...

ith column $\vec{d_i}$ is a combination of $\vec{q_1}, \dots \vec{q_i}$; that is, $\vec{d_1}$ can be reconstructed from $\vec{q_1}$; $\vec{d_2}$ from $\vec{q_1}, \vec{q_2}$; $\vec{d_3}$ from $\vec{q_1}, \vec{q_2}, \vec{q_3}$ and so on.

Therefore,

$$\underbrace{[\vec{d_1} \cdots \vec{d_k}]}_{l \times k} = \underbrace{\underbrace{[\vec{q_1} \cdots \vec{q_k}]}_{l \times k}}_{Q:\ \text{orthonormal columns}} \underbrace{\overbrace{\begin{bmatrix} * & * & \cdots & * \\ 0 & * & & * \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & \cdots 0 & * \end{bmatrix}}^{k \times k}}_{R:\ \text{upper-triangular}}$$

"QR factorization:" next best thing after having D already with orthonormal columns

Back to Least squares (LS): $\vec{s} = D\vec{p} + \vec{e}$

LS picks $\vec{p}$ such that $\vec{e} = \vec{s} - D\vec{p} \perp$ column space of D

i.e.,

$$D^T\vec{e} = D^T(\vec{s} - D\vec{p}) = 0$$

$$D^T\vec{s} = D^TD\vec{p} \cdots (LS)$$

Instead of inverting D^TD, do Gram-Schmidt (QR factorization) on D:

$$D = QR$$

Rewrite (LS):

$$(QR)^T\vec{s} = (QR)^T(QR)\vec{p}$$

$$R^TQ^T\vec{s} = R^T\underbrace{Q^TQ}_{=I}R\vec{p}$$

$$R^T(Q^T\vec{s}) = R^T(R\vec{p})$$

Solve $R\vec{p} = Q^T\vec{s}$

$$\begin{bmatrix} * & * & \cdots & * \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & \\ 0 & \cdots & 0 & * \end{bmatrix} \begin{bmatrix} p_1 \\ \vdots \\ p_k \end{bmatrix} = Q^T\vec{s}$$

last row gives: $* p_k = (Q^T\vec{s})_k \Rightarrow p_k = \frac{1}{*}(Q^T\vec{s})_k$

2nd to last row = $* p_{k-1} + * p_k = (Q^T\vec{s})_{k-1}$

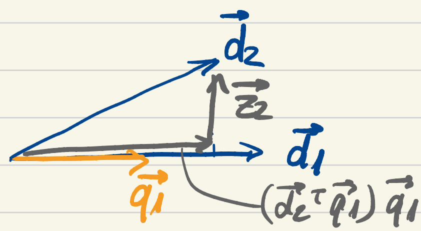
$\hookrightarrow$ p_{k-1} solved from here

Can solve by back substitution (fast!) without matrix inversion.

<u>Gram-Schmidt Procedure</u>: Given linearly independent columns $\vec{d_1}, \ldots \vec{d_k}$:

<u>Step 1</u>: $\vec{q_1} = \frac{1}{\|\vec{d_1}\|} \vec{d_1}$

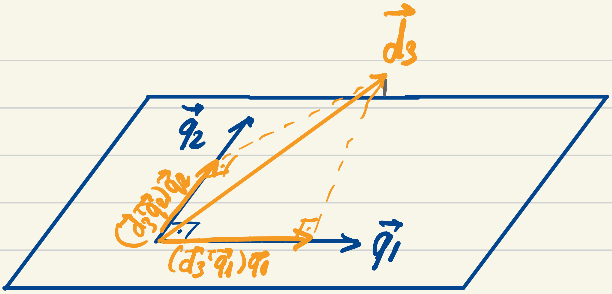
<u>Step 2</u>: $\vec{z_2} = \vec{d_2} - (\vec{d_2}^T\vec{q_1})\vec{q_1}$



$$\vec{q_2} = \frac{1}{\|\vec{z_2}\|}\vec{z_2}$$

<u>Step 3</u>: $\vec{z_3} = \vec{d_3} - (\vec{d_3}^T\vec{q_1})\vec{q_1} - (\vec{d_3}^T\vec{q_2})q_2$

$$\vec{q_3} = \frac{1}{\|\vec{z_3}\|}\vec{z_3}$$



⋮

Step k: $\vec{z}_k = \vec{d}_k - \sum_{j=1}^{k-1} (\vec{d}_k^T \vec{q}_j)\vec{q}_j \quad -- (2)$

$$\vec{q}_k = \frac{1}{\|\vec{z}_k\|}\vec{z}_k$$

Show $\vec{z}_k^T \vec{q}_i = 0 \quad i<k$ (therefore $\vec{q}_k^T \vec{q}_i = 0 \quad i<k$)

substitute (2):

$$\vec{d}_k^T \vec{q}_i - \sum_{j=1}^{k-1} (\vec{d}_k^T \vec{q}_j)\underbrace{\vec{q}_j^T \vec{q}_i}_{=\begin{cases}0 & j\neq i\\ 1 & j=i\end{cases}}$$

this means all terms in summation drop out other than $j=i$

$$\left((\vec{d}_k^T\vec{q}_j)\vec{q}_j^T\vec{q}_i\right)\Big|_{j=i} = (\vec{d}_k^T\vec{q}_i)\underbrace{\vec{q}_i^T\vec{q}_i}_{=1}$$

$$= \vec{d}_k^T\vec{q}_i - \vec{d}_k^T\vec{q}_i = 0.$$

So we have shown $\vec{q}_k^T \vec{q}_i = 0$ when $k < i$. -- (3)

Same is true when $k > i$: $\vec{q}_k^T \vec{q}_i = \vec{q}_i^T \vec{q}_k$ where now i is smaller than k and (3) applies with k, i swapped.

Combining: $\vec{q}_k^T \vec{q}_i = 0$ when $k \neq i$. By the normalization $\vec{q}_k = \frac{1}{\|\vec{z}_k\|} \vec{z}_k$ in each step, we also have $\|\vec{q}_k\| = 1$ for each k, so Gram-Schmidt generates an orthonormal basis. In addition, $\vec{a}_1$ depends on $\vec{q}_1$ only; $\vec{a}_2$ on $\vec{q}_1, \vec{q}_2$; $\vec{a}_k$ on $\vec{q}_1, \dots \vec{q}_k$ as desired.