Recap of Linear Algebra Basics

1) Linearly independent vectors :

A set of $n \times 1$ vectors $\underline{v}_i \in \mathbb{R}^n$, $i = 1, \dots, k$, is linearly dependent if there exist k scalars : $\beta_1, \beta_2, \dots, \beta_k$, not all equal to zero, such that

$$\beta_1 \underline{v}_1 + \beta_2 \underline{v}_2 + \dots + \beta_k \underline{v}_k = 0 .$$

A set of vectors $\underline{v}_i \in \mathbb{R}^n$, $i = 1, \dots, k$, is linearly independent if they are NOT linearly dependent, that is, $\beta_1 \underline{v}_1 + \beta_2 \underline{v}_2 + \dots + \beta_k \underline{v}_k = 0$ holds only for $\beta_1 = \beta_2 = \dots = \beta_k = 0$

Examples :

- Any list of vectors containing the zero vector is linearly dependent .

- $\underline{v}_1 = \begin{pmatrix} 0.2 \\ -7.0 \\ 8.6 \end{pmatrix}$, $\underline{v}_2 = \begin{pmatrix} -0.1 \\ 2.0 \\ -1.0 \end{pmatrix}$, $\underline{v}_3 = \begin{pmatrix} 0.0 \\ -1.0 \\ 2.2 \end{pmatrix}$ are linearly <u>dependent</u>, since $\underline{v}_1 + 2\underline{v}_2 - 3\underline{v}_3 = 0$.

- $\underline{v}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $\underline{v}_2 = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}$, $\underline{v}_3 = \begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix}$ are linearly <u>independent</u> since

$$\beta_1 \underline{v}_1 + \beta_2 \underline{v}_2 + \beta_3 \underline{v}_3 = 0$$

$$\Rightarrow \beta_1 = \beta_2 = \beta_3 = 0.$$

- Standard basis vectors $\underline{e}_1, \underline{e}_2, \dots, \underline{e}_n$ in $\mathbb{R}^n$ are linearly <u>independent</u> since

$$\sum_{i=1}^{n} \beta_i \underline{e}_i = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_n \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$\Rightarrow \beta_1 = \beta_2 = \dots = \beta_n = 0.$$

Recall:

$$\underline{e}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad \underline{e}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{ etc.}$$

Independence Dimension Inequality:

If the $n \times 1$ vectors $\underline{v}_1, \underline{v}_2, \ldots, \underline{v}_k$ are linearly independent, then $k \leq n$.

Another way to say this:

A linearly independent collection of $n \times 1$ vectors can have at most n elements.

Yet another way to say this:

Any collection of $n+1$ or more vectors of size $n \times 1$, must be linearly dependent.

Example:

Any 3 vectors $\underline{v}_1, \underline{v}_2, \underline{v}_3$, each of size 2×1, must be linearly dependent.

② Orthogonal and Orthonormal vectors :

A collection of k vectors in $\mathbb{R}^n$ is (mutually) orthogonal if $\underline{v}_i \perp \underline{v}_j$ for all $(i, j) \in \{1, \dots, k\}$ such that $i \neq j$.

that is, $\underline{v}_i^T \underline{v}_j = 0$ for all $i \neq j$.

They are called orthonormal if $\underline{v}_i^T \underline{v}_j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$

unit 2-norm (magnitude)

③ Gram-Schmidt Algorithm :

(Determines if $\underline{v}_1, \dots, \underline{v}_k \in \mathbb{R}^n$ are linearly independent or not)

Idea: • If $\underline{v}_1, \dots, \underline{v}_k \in \mathbb{R}^n$ are indeed linearly independent, then the Gram-Schmidt algorithm produces an orthonormal collection of vectors : $\underline{q}_1, \dots, \underline{q}_k$ with the property that :

for each $i \in \{1, \ldots, k\}$, the vector $\underline{v}_i$ is a linear combination of $\underline{q}_1, \underline{q}_2, \ldots, \underline{q}_i$

AND $\underline{q}_i$ is a linear combination of $\underline{v}_1, \ldots, \underline{v}_i$.

- If the vectors $\underline{v}_1, \ldots, \underline{v}_{j-1}$ are linearly independent, but the vectors $\underline{v}_1, \ldots, \underline{v}_j$ are linearly dependent, the Gram-Schmidt algorithm detects this, and terminates. In other words, the algorithm finds the first vector $\underline{v}_j$ that is a linear combination of the previous vectors: $\underline{v}_1, \ldots, \underline{v}_{j-1}$.

Algorithm (Gram-Schmidt):

Given vectors $\underline{v}_1, \ldots, \underline{v}_k \in \mathbb{R}^n$

For $i=1$, $\underline{\tilde{q}}_1 = \underline{v}_1$

for $i = 1:k$

- (orthogonalization) $\tilde{q}_i = \underline{v}_i - (\underline{q}_1^T \underline{v}_i)\underline{q}_1 - \ldots - (q_{i-1}^T \underline{v}_i)\underline{q}_{i-1}$

- (check linear dependence) if $\tilde{\underline{a}}_i = \underline{0}$, quit.
- (normalization) $\underline{a}_i = \tilde{\underline{a}}_i / \| \tilde{\underline{a}}_i \|_2$.

end.

Complexity: $O(nk^2)$

If the algorithm does not quit (in "check linear dependence" step), that is, $\tilde{\underline{a}}_1, \tilde{\underline{a}}_2, \ldots, \tilde{\underline{a}}_k$ are all non-zero, then we conclude that the original collection of vectors $\underline{v}_1, \ldots, \underline{v}_k$ are linearly independent.

If the algorithm does quit early, with $\tilde{\underline{a}}_j = \underline{0}$, then we conclude that the original collection of vectors is linearly dependent, and indeed, $\underline{v}_j$ is a linear combination of $\underline{v}_1, \ldots, \underline{v}_{j-1}$

Example: $\underline{v}_1 = \begin{pmatrix} -1 \\ 1 \\ -1 \\ 1 \end{pmatrix}$, $\underline{v}_2 = \begin{pmatrix} -1 \\ 3 \\ -1 \\ 3 \end{pmatrix}$, $\underline{v}_3 = \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \end{pmatrix}$

Applying Gram-Schmidt algorithm:

$\boxed{i=1}$ we get $\|\underline{\tilde{q}}_1\|_2 = 2$, so $\underline{q}_1 = \dfrac{\underline{\tilde{q}}_1}{\|\underline{\tilde{q}}_1\|_2} = \begin{pmatrix} -1/2 \\ +1/2 \\ -1/2 \\ +1/2 \end{pmatrix}$

which is simply $\underline{v}_1$ normalized.

$\boxed{i=2}$ we get $\underline{q}_1^T \underline{v}_2 = 4$, so

$$\underline{\tilde{q}}_2 = \underline{v}_2 - (\underline{q}_1^T \underline{v}_2)\,\underline{q}_1 = \begin{pmatrix} -1 \\ 3 \\ -1 \\ 3 \end{pmatrix} - 4\begin{pmatrix} -1/2 \\ +1/2 \\ -1/2 \\ +1/2 \end{pmatrix}$$

$$= \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix},$$

which is indeed orthogonal to $\underline{q}_1$ (and $\underline{v}_1$). Further, $\|\underline{\tilde{q}}_2\|_2 = 2$,

$$\therefore \quad \underline{q}_2 = \frac{\underline{\tilde{q}}_2}{\|\underline{\tilde{q}}_2\|_2} = \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}.$$

$\boxed{i = 3}$ we get $\underline{q}_1^T \underline{v}_3 = 2$ and $\underline{q}_2^T \underline{v}_3 = 8$.

$$\therefore \quad \underline{\tilde{q}}_3 = \underline{v}_3 - (\underline{q}_1^T \underline{v}_3)\underline{q}_1 - (\underline{q}_2^T \underline{v}_3)\underline{q}_2$$

$$= \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \end{pmatrix} - 2\begin{pmatrix} -1/2 \\ +1/2 \\ -1/2 \\ +1/2 \end{pmatrix} - 8\begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}$$

$$= \begin{pmatrix} -2 \\ -2 \\ +2 \\ +2 \end{pmatrix},$$ which is orthogonal to $\underline{q}_1$ and $\underline{q}_2$ (and $\underline{v}_1$ and $\underline{v}_2$).

We have: $\|\underline{\tilde{q}}_3\|_2 = 4$, so $\underline{q}_3 = \dfrac{\underline{\tilde{q}}_3}{\|\underline{\tilde{q}}_3\|_2} = \begin{pmatrix} -1/2 \\ -1/2 \\ +1/2 \\ +1/2 \end{pmatrix}.$

Since the algorithm did NOT terminate early, $\therefore$ $\underline{v}_1, \underline{v}_2, \underline{v}_3$ are linearly independent.

4) QR factorization:

Suppose $\underbrace{A}_{n\times k} = \left[\underline{a}_1 \quad \underline{a}_2 \quad \ldots \quad \underline{a}_k\right]$

each of these are column vectors of size $n\times 1$

To say that the column vectors $\underline{a}_1, \underline{a}_2, \ldots, \underline{a}_k$ are orthonormal, is same as saying: $\boxed{A^TA = I}$

I: $k\times k$ identity matrix

Any rectangular matrix A that satisfies $A^TA = I$ is called **Orthonormal**.

If A is square and it satisfies $A^TA = I$, then it is called Orthogonal. (we already know examples of orthogonal: identity matrix, permutation matrix etc.)

Any orthonormal matrix A satisfies;

$$\| A\underline{x} \|_2 = \| \underline{x} \|_2$$

$\langle A\underline{x}, A\underline{y} \rangle = \langle \underline{x}, \underline{y} \rangle$, where $\langle \underline{a}, \underline{b} \rangle$ denotes inner (dot) product

i.e., $\langle \underline{a}, \underline{b} \rangle \equiv \underline{a}^T \underline{b}$.

$\angle(A\underline{x}, A\underline{y}) = \angle(\underline{x}, \underline{y})$

angle between the vectors } Why?

$$\angle(Ax, Ay) = \arccos\left(\frac{(A\underline{x})^T(A\underline{y})}{\|A\underline{x}\|_2 \|A\underline{y}\|_2}\right)$$

$$= \arccos\left(\frac{\underline{x}^T\underline{y}}{\|\underline{x}\|_2 \|\underline{y}\|_2}\right)$$

$$= \angle(\underline{x}, \underline{y}).$$

We can express the Gram-Schmidt algorithm in a compact form using matrices.

Suppose $\underbrace{A}_{n \times k}$ has columns $\underline{v}_1, \dots, \underline{v}_k$, which are linearly indepen-dent

By independence-dimension inequality, $n \geq k$
i.e., A is tall or square.

Let Q be the $n \times k$ matrix with columns $\underline{q}_1, \ldots, \underline{q}_k$, the orthonormal vectors produced by the Gram-Schmidt algorithm applied to the collection of $n \times 1$ vectors: $\underline{v}_1, \ldots, \underline{v}_k$.

Orthonormality of $\underline{q}_1, \underline{q}_2, \ldots, \underline{q}_k$ is same as saying

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

Recall that the equation that relates $\underline{v}_i$ with $\underline{q}_i$ is:

$$\underline{v}_i = (\underline{q}_1^T \underline{v}_i)\underline{q}_1 + \ldots + (\underline{q}_{i-1}^T \underline{v}_i)\underline{q}_{i-1} + \|\tilde{\underline{q}}_i\|_2 \underline{q}_i$$

where $\tilde{q}_i$ is the vector obtained in the first step of the Gram-Schmidt algorithm, as

$$\underline{v}_i = R_{1i}\underline{q}_1 + \ldots + R_{ii}\underline{q}_i,$$

where $R_{ij} = \begin{cases} \underline{q}_i^T \underline{v}_j & \text{for } i < j \\ \|\tilde{\underline{q}}_i\| & \text{for } i = j \\ 0 & \text{for } i > j \end{cases}$

zero $\rightarrow 0$

So, we can write the above equations in compact form:

$$\underbrace{A}_{n\times k} = \underbrace{Q}_{n\times k}\ \underbrace{R}_{k\times k}$$

$\boxed{n \geq k}$ ($\because$ A has linearly independent columns)

Q is orthonormal matrix (i.e., $Q^TQ = I$)

R is square, upper triangular, with positive diagonal

This is called QR factorization of matrix A.

Special case: If A is square (i.e., if $n=k$) then Q will be orthogonal.

Gram-Schmidt is NOT the only algorithm for QR factorization

There are other algorithms (e.g., Householder algorithm) for this

MATLAB command "qr"

5) Left Inverse, Right Inverse, Inverse, Pseudo-Inverse

<u>Left inverse:</u> X is the left inverse of A if $XA = I$

If A has size $m \times n$, then X is of size $n \times m$.

<u>Example:</u> $A = \begin{bmatrix} -3 & -4 \\ 4 & 6 \\ 1 & 1 \end{bmatrix}$ has two different left inverses:

$$X_1 = \frac{1}{9}\begin{bmatrix} -11 & -10 & 16 \\ 7 & 8 & -11 \end{bmatrix}, \quad X_2 = \frac{1}{2}\begin{bmatrix} 0 & -1 & 6 \\ 0 & 1 & -4 \end{bmatrix},$$

Since $X_1 A = X_2 A = I$. This shows that a left-invertible matrix A may have more than one left inverse.

<u>Example:</u> A has orthonormal columns $\Leftrightarrow A^T A = I$.

Then A is left invertible. Its left-inverse is $X = A^T$.

<u>Fact:</u> Matrix A has left inverse if and only if its columns are linearly independent.

<u>Example:</u> Suppose $\underset{m \times n}{\underline{A}}$ with $m < n$, i.e., A is wide.

Then, by independence-dimension inequality the columns are linearly dependent $\Rightarrow$ $\therefore$ A is NOT left invertible. In other words, only square or tall matrices are left-invertible.

Right inverse: X is the right inverse of A if $AX = I$.

Examples: • A matrix is right invertible if and only if its rows are linearly independent.

• A tall matrix cannot have right inverse. Only square or wide matrices are right invertible.

Inverse: If a matrix is left AND right invertible, then the left and right inverses are unique and equal.

Why: $AX = I$, $YA = I$, $\Rightarrow X = (YA)X = Y(AX) = Y$.

We say $X = Y$ is simply the "inverse" of A, and denote it by A^{-1}.

Pseudo-inverse:

Fact: $\underset{m\times n}{A}$ has linearly $\overbrace{\text{independent columns}}^{\text{(meaning } A \text{ is square or tall)}}$ $\Leftrightarrow$ $\underset{m\times n}{A}$ has left inverse

$\Leftrightarrow$ A^TA is invertible

Fact: The left inverse of A is equal to $(A^TA)^{-1}A^T$

why! $\left((A^TA)^{-1}A^T\right)A = (A^TA)^{-1}(A^TA) = I$.

This particular left inverse has a name: it is called the $\boxed{\text{pseudo-inverse}}$ of A, denoted as A^+

$$\therefore \quad A^+ = (A^TA)^{-1}A^T.$$

If A is square, then $A^+ = A^{-1}$ (that is, pseudo-inverse equals the inverse)

why: $A^+ = (A^TA)^{-1}A^T = A^{-1}A^{-T}A^T = A^{-1}I = A^{-1}$!

Obviously, the above eqn. does not make sense if A is tall.

We can also define pseudo-inverse of a wide matrix as:

$$A^T(AA^T)^{-1} \quad \text{since} \quad A\,A^T(AA^T)^{-1} = I.$$

Pseudo-inverse via QR factorization:

If A is left-invertible, its columns are linearly independent and the QR factorization $A = QR$ exists.

We have:

$$A^TA = (QR)^T QR = R^TQ^TQR = R^TR.$$

$$\Rightarrow A^+ = (A^TA)^{-1}A^T = (R^TR)^{-1}(QR)^T$$

$$= R^{-1}R^{-T}R^TQ^T$$

$$= \underbrace{R^{-1}Q^T}_{\text{simple formula to compute pseudo-inverse}}$$

Complexity: $O(mn^2)$.