$$\underbrace{\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}_{m\times n}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}_{n\times 1}}_{x} = \underbrace{\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}_{m\times 1}}_{b}$$

CX1104: **Linear Algebra for Computing**

Chap. No : **6.3.1**

Lecture : **Orthogonality**

Topic : **Gram–Schmidt for QR**

Concept : **Motivation and Review of Concepts**

Instructor: **A/P Chng Eng Siong**

TAs: **Zhang Su, Vishal Choudhari**

Rev: ces-22July2020

Rev: 29th June 2020

Introducing A = QR

QR decomposition, given A (mxn) matrix, we can decompose it into the product of 2 matrixes,

$$A = QR$$

- Properties of Q:
  - C(Q) == C(A)
  - $Q^T Q = I$, i.e Q has orthonormal column (BUT not necessary square)
  - QQ^T = projection matrix into col (A)

- R is a square upper triangle matrix and Depending if
  - A has independent col, then R is invertible,
  - A has dependent col, then R is NOT-invertible.

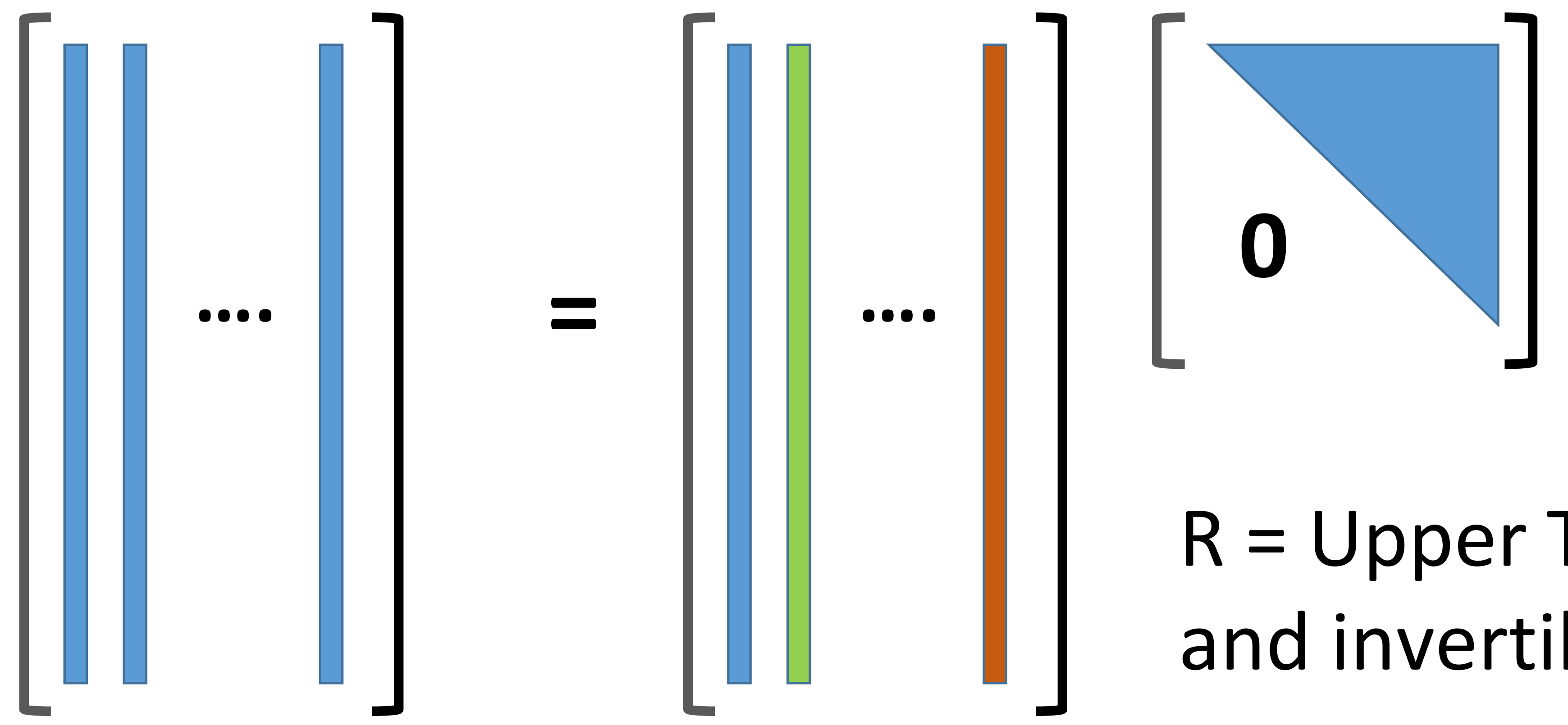
Warning: Theorem 12 is for A Having independent column.

THEOREM 12

The QR Factorization

If A is an $m \times n$ matrix with linearly independent columns, then A can be factored as $A = QR$, where Q is an $m \times n$ matrix whose columns form an orthonormal basis for Col A and R is an $n \times n$ upper triangular invertible matrix with positive entries on its diagonal.

$$A = [\,\mathbf{x}_1 \quad \cdots \quad \mathbf{x}_n\,] = [\,Q\mathbf{r}_1 \quad \cdots \quad Q\mathbf{r}_n\,] = QR$$



R = Upper Triangle and invertible

A $\in R^{m*n}$ Q = columns are orthonormal

Ref: 1) https://en.wikipedia.org/wiki/QR_decomposition
2) http://ee263.stanford.edu/lectures/qr.pdf

Motivation for QR

It has many applications, e.g,
solving least squares:

$$
\begin{aligned}
Ax &= y \\
QRx &= y \\
Q^T QRx &= Q^T y \\
\\
Rx &= Q^T y
\end{aligned}
$$

Is can be easily solved because R is upper triangle
(If R is not invertible, it will be more involved, see least squares chapter)

There are at least 3 approaches to realise QR decomposition,
we will only introduce Gram-Schmidt orthogonalization to get Q first
https://www.math.ucla.edu/~yanovsky/Teaching/Math151B/handouts/GramSchmidt.pdf
https://towardsdatascience.com/can-qr-decomposition-be-actually-faster-schwarz-rutishauser-algorithm-a32c0cde8b9b

What does having same column space mean? C(Q) == C(A)

When we can convert

$$Ax = b$$

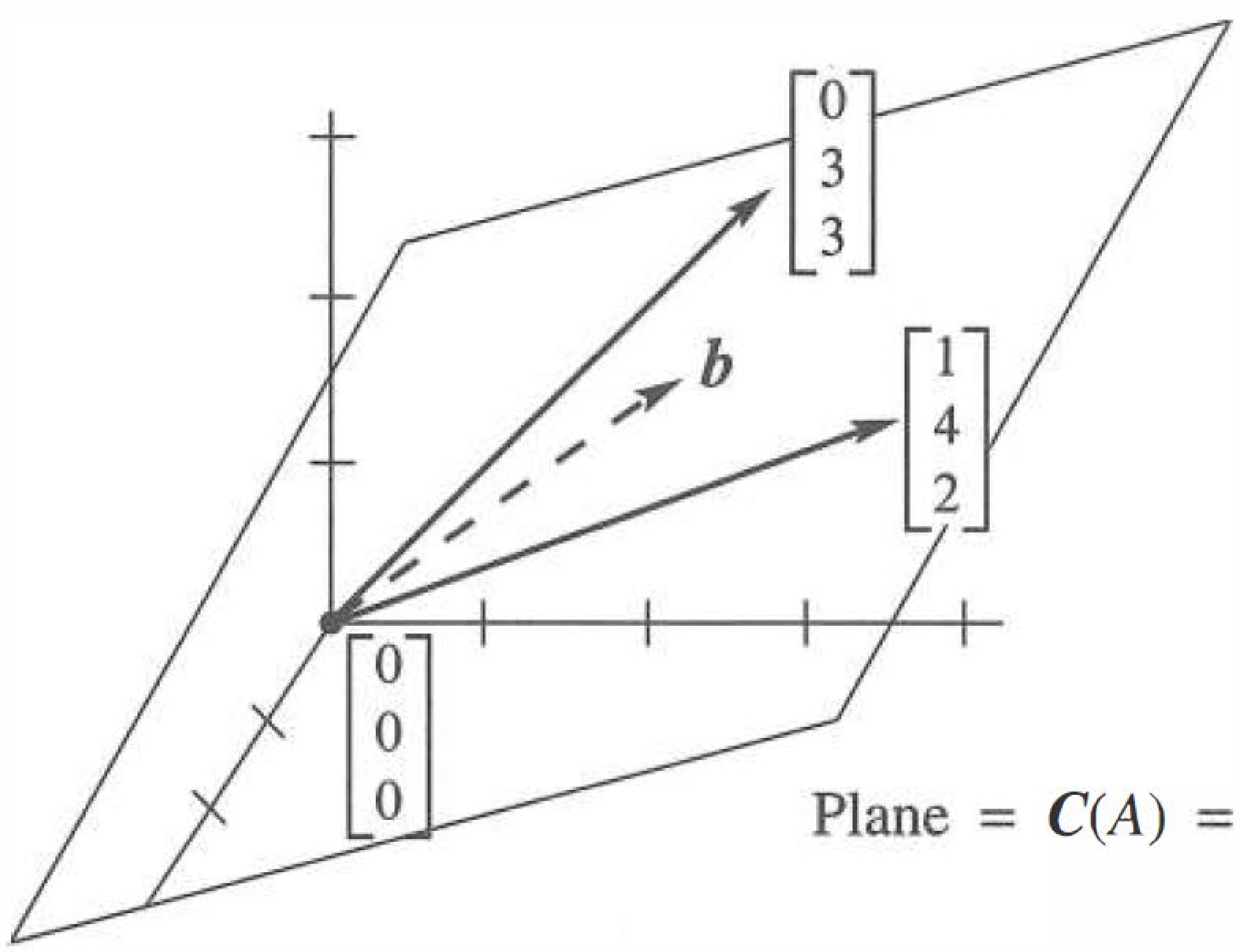
To

$$QRx = b$$

Then finding the solution
x is easier and more efficient especially when A is large sized
(e.g thousands of rows and columns)

The found solution x will be the same.

Chapter 3. Vector Spaces and Subspaces



$$A = \begin{bmatrix}1 & 0\\4 & 3\\2 & 3\end{bmatrix}$$

$$b = .4\begin{bmatrix}1\\4\\2\end{bmatrix} + .3\begin{bmatrix}0\\3\\3\end{bmatrix}$$

$Ax = b$ has $x = \begin{bmatrix}.4\\.3\end{bmatrix}$

Plane $= C(A) =$ all vectors Ax

Figure 3.2: The column space $C(A)$ is a plane containing the two columns. $Ax = b$ is solvable when b is on that plane. Then b is a combination of the columns.

Interpretation: since C(Q) == C(A),
then Columns of Q are orthonormal basis for range(A), since $Q^T Q = I$

Revision: Projecting y onto an orthogonal vs orthonormal basis

Lay5e pg 350

THEOREM 8 **The Orthogonal Decomposition Theorem**

Let W be a subspace of $\mathbb{R}^n$. Then each $\mathbf{y}$ in $\mathbb{R}^n$ can be written uniquely in the form

$$\mathbf{y} = \hat{\mathbf{y}} + \mathbf{z} \tag{1}$$

where $\hat{\mathbf{y}}$ is in W and $\mathbf{z}$ is in $W^{\perp}$. In fact, if $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ is any orthogonal basis of W, then

$$\hat{\mathbf{y}} = \frac{\mathbf{y}\cdot\mathbf{u}_1}{\mathbf{u}_1\cdot\mathbf{u}_1}\mathbf{u}_1 + \cdots + \frac{\mathbf{y}\cdot\mathbf{u}_p}{\mathbf{u}_p\cdot\mathbf{u}_p}\mathbf{u}_p \tag{2}$$

and $\mathbf{z} = \mathbf{y} - \hat{\mathbf{y}}$.

The vector $\hat{\mathbf{y}}$ in (1) is called the **orthogonal projection of y onto** W and often is written as $\text{proj}_W\, \mathbf{y}$. See Figure 2. When W is a one-dimensional subspace, the formula for $\hat{\mathbf{y}}$ matches the formula given in Section 6.2.

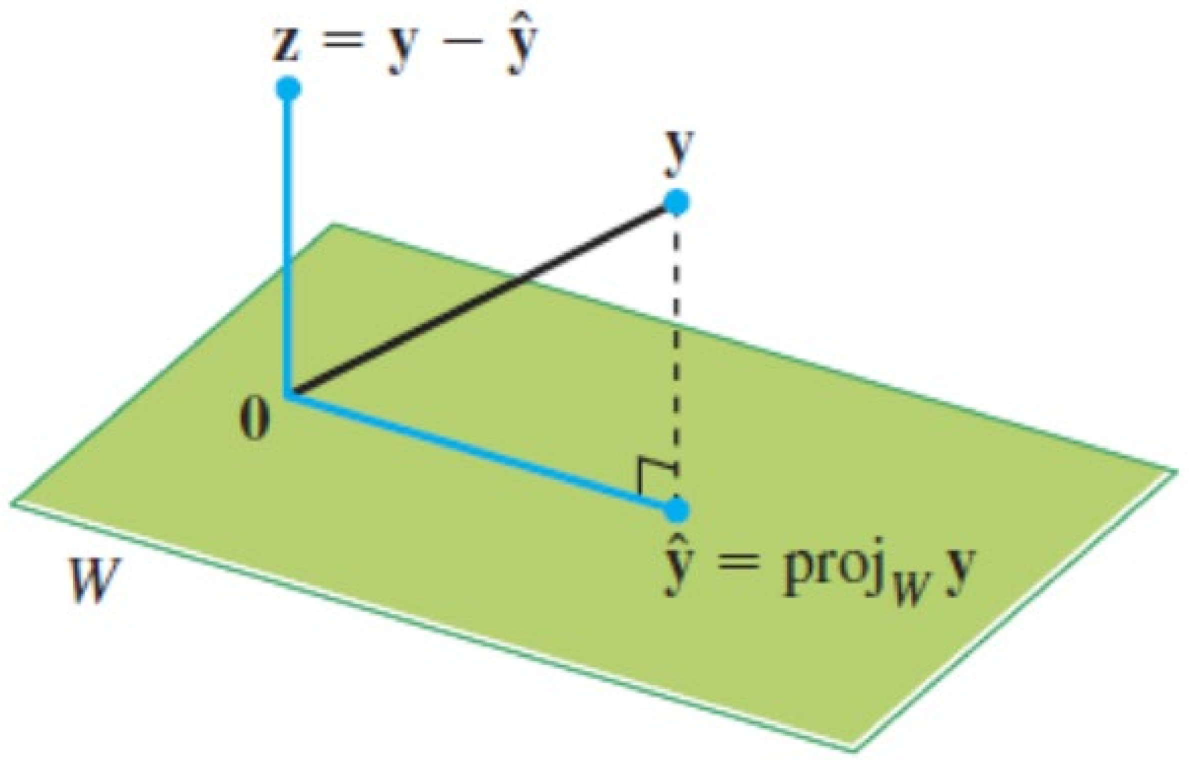


FIGURE 2 The orthogonal projection of $\mathbf{y}$ onto W.

Lay5e pg 353

The final theorem in this section shows how formula (2) for $\text{proj}_W\, \mathbf{y}$ is simplified when the basis for W is an orthonormal set.

THEOREM 10

If $\{\mathbf{u}_1, \dots, \mathbf{u}_p\}$ is an orthonormal basis for a subspace W of $\mathbb{R}^n$, then

$$\text{proj}_W\, \mathbf{y} = (\mathbf{y}\cdot\mathbf{u}_1)\mathbf{u}_1 + (\mathbf{y}\cdot\mathbf{u}_2)\mathbf{u}_2 + \cdots + (\mathbf{y}\cdot\mathbf{u}_p)\mathbf{u}_p \tag{4}$$

If $U = [\,\mathbf{u}_1 \quad \mathbf{u}_2 \quad \cdots \quad \mathbf{u}_p\,]$, then

$$\text{proj}_W\, \mathbf{y} = UU^T\mathbf{y} \quad \text{for all } \mathbf{y} \text{ in } \mathbb{R}^n \tag{5}$$

PROOF Formula (4) follows immediately from (2) in Theorem 8. Also, (4) shows that $\text{proj}_W\, \mathbf{y}$ is a linear combination of the columns of U using the weights $\mathbf{y}\cdot\mathbf{u}_1, \mathbf{y}\cdot\mathbf{u}_2, \dots, \mathbf{y}\cdot\mathbf{u}_p$. The weights can be written as $\mathbf{u}_1^T\mathbf{y}, \mathbf{u}_2^T\mathbf{y}, \dots, \mathbf{u}_p^T\mathbf{y}$, showing that they are the entries in $U^T\mathbf{y}$ and justifying (5). ∎

How to find Q from A?

What does Gram-Schmidt Process Do?

It orthogonalises a set of vectors!

- A is a mxn matrix,
  assume columns are Not orthogonal
- $\boldsymbol{v_i} \in \boldsymbol{R^m}, i = 1..\boldsymbol{n}$

- The columns of Q are
  Orthonormal u_i
  $i = 1..n$
- **And spans the same space as A**

$A = \{v_1, v_2, v_3, \cdots, v_n\} \rightarrow$ Gram-Schmidt Process $\rightarrow Q = \{u_1, u_2, u_3, \cdots, u_n\}$

$$Span(A) = Span(Q)$$

Note:Q spans the same m-dimensional subspace of R^m as that of A

Ref: [http://www.seas.ucla.edu/~vandenbe/133A/lectures/qr.pdf   Slide 6.7](http://www.seas.ucla.edu/~vandenbe/133A/lectures/qr.pdf)
See Matlab: https://www.mathworks.com/help/matlab/ref/qr.html
(economy QR factorization vs full QR decomposition)
More explanations of full vs economy qr : http://www.ece.northwestern.edu/local-apps/matlabhelp/techdoc/math_anal/mat_li23.html