Lecture 7. Matrix formulation of orthogonal projections and Gram-Schmidt orthogonalization, QR decomposition, and its application to the least-squares fit problem.

1. Matrix formulation of orthogonal projections (see p. 235 of the book for an alternative approach)

Let \( S = \{ v_1, \ldots, v_q \} \) be an orthonormal set in a \( p \)-dimensional vector space \( W \) (we assume \( q < p \)). In Lecture 6 we showed that the projection of any \( v \in W \) on \( \operatorname{Sp}(S) \) is:

\[
P_S v = \alpha_1 v_1 + \ldots + \alpha_q v_q,
\]

\[\alpha_j = \frac{\langle v, v_j \rangle}{\langle v_j, v_j \rangle} \rightarrow 1.\]

Then:

\[
P_S v = v_1 \frac{\langle v, v_1 \rangle}{P_1 v} + \ldots + v_q \frac{\langle v, v_q \rangle}{P_q v}
\]

\[
= \frac{v_1 (v_1^T v)}{P_1 v} + \ldots + \frac{v_q (v_q^T v)}{P_q v}
\]

\[
= \frac{v_1 (v_1^T v)}{P_1 v} + \ldots + \frac{v_q (v_q^T v)}{P_q v}
\]

\[
= \left( \frac{v_1 (v_1^T v)}{P_1 v} + \ldots + \frac{v_q (v_q^T v)}{P_q v} \right)
\]

\[
= \left( \frac{v_1 v_1^T}{P_1 v} + \ldots + \frac{v_q v_q^T}{P_q v} \right)
\]

\[
= \frac{P_S v}{[v_1, \ldots, v_q][v_1^T, \ldots, v_q^T] v} = QQ^T v
\]
Thus, \( P_\Sigma \mathbf{v} = \frac{\mathbf{QQ}^T}{\mathbf{P}} \mathbf{v} \) \hspace{1cm} (3) \\

\( P = \text{projection matrix} \).

This is the matrix form of (1), (2).

Properties of \( \mathbf{Q} \) and \( \mathbf{P} \) (see p. 236):

\( \mathbf{Q}^T \mathbf{Q} = \mathbf{I}_q \) \hspace{1cm} (4) \\
\( q \times p \rightarrow p \times q \)

\[
\begin{bmatrix}
\mathbf{v}_1^T \\
\mathbf{v}_2^T \\
\vdots \\
\mathbf{v}_q^T
\end{bmatrix}
\begin{bmatrix}
\mathbf{v}_1 \\
\mathbf{v}_2 \\
\vdots \\
\mathbf{v}_q
\end{bmatrix} = \mathbf{I}_q \quad \text{since} \quad \mathbf{v}_j \perp \mathbf{v}_k, \ k \neq j, \quad \| \mathbf{v}_j \|^2 = 1.
\]

Note that \( \mathbf{QQ}^T \neq \mathbf{I}_p \) unless \( p=q \).

\( \mathbf{P}^2 = \mathbf{P} \quad \mathbf{P} \rightarrow \mathbf{I}_p \quad \mathbf{P} \) \hspace{1cm} (5)

\( \mathbf{P}^2 = \mathbf{Q} \left( \mathbf{Q}^T \mathbf{Q} \right) \mathbf{Q}^T = \mathbf{Q} \mathbf{Q}^T = \mathbf{P} \).

Meaning: projecting a projection doesn't change that projection:

\( \text{projections} \left( \text{projections} \mathbf{v} \right) = \text{projections} \mathbf{v} \).

\(10/6/08\)

2. **QR decomposition**.

We have found the matrix form of the projection operation. Let's now find the matrix form of the Gram-Schmidt orthogonalization.

Let's do it for a 3x3 matrix, it's similar for \( p \times q \).

So, let's orthogonalize the set \( \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \} \) via G-S:
1) \[ u_1 = v_1, \text{ or } v_1 = u_1, \text{ or } \]
\[ v_1 = \begin{bmatrix} u_1, u_2, u_3 \end{bmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \quad (\star) \]

2) \[ u_2 = v_2 - P \mathbf{v}_2 = v_2 - \alpha_{1,2} u_1 \]
\[ \Rightarrow v_2 = \alpha_{1,2} u_1 + u_2. \]
\[ \Rightarrow v_2 = \begin{bmatrix} u_1, u_2, u_3 \end{bmatrix} \begin{pmatrix} \alpha_{1,2} \\ 1 \\ 0 \end{pmatrix}. \quad (\star \star) \]

3) \[ u_3 = v_3 - P \mathbf{v}_3 - P \mathbf{v}_3 = v_3 - \alpha_{1,3} u_1 - \alpha_{2,3} u_2 \]
\[ \Rightarrow v_3 = \alpha_{1,3} u_1 + \alpha_{2,3} u_2 + u_3. \]
\[ \Rightarrow v_3 = \begin{bmatrix} u_1, u_2, u_3 \end{bmatrix} \begin{pmatrix} \alpha_{1,3} \\ \alpha_{2,3} \\ 1 \end{pmatrix}. \quad (\star \star \star) \]

Combining (\star), (\star \star), (\star \star \star), we get:

\[ \begin{bmatrix} v_1, v_2, v_3 \end{bmatrix} = \begin{bmatrix} u_1, u_2, u_3 \end{bmatrix} \begin{pmatrix} 1 & \alpha_{1,2} & \alpha_{1,3} \\ 0 & 1 & \alpha_{2,3} \\ 0 & 0 & 1 \end{pmatrix} \]

where the matrix with orthogonal columns and upper-D matrix with 1 on the diagonal.

Note: What if the set \( \{ v_1, v_2, v_3 \} \) is l.d.? Suppose \( v_3 \) is l.d. on \( v_1, v_2 \). Then
\[ v_3 \in \text{Sp} \{ v_1, v_2 \} = \text{plane} \ (v_1, v_2). \]
But \( \text{plane} \ (v_1, v_2) = \text{plane} \ (u_1, u_2) \)
\[ v_1 \in \text{lin.comb. of } u_1, u_2. \]
Then \( v_3 \in \text{plane} \ (u_1, u_2) \).
But then \( v_3 = P_{u_1} (v_3) + P_{u_2} (v_3) \).
Then
\[ y_3 = v_3 - (P_{v_1} v_3 + P_{v_2} v_3) = 0 \]

In general:

\[ Q_0 = [u_1, \ldots, u_9] \] has zero vectors iff the set \( \{ v_3, \ldots, v_9 \} \) is l.d.

**Thm 1 (Key Thm. 5.82(a))**

Any \( A = [A_1, \ldots, A_9] = p \times q \) for any \( p, q \) can be represented as

\[ A = Q_0 R_0 \quad (6) \]

\[ p \times q \quad p \times q \quad q \times q \]

where:

(i) columns of \( Q_0 \) are either mutually

- or zero, and

(ii) \( R_0 \) is upper-D, has 1s on the

main diagonal, and hence is nonsingular

(HEW3, #1(a)).

(iii) \( \| u_j \|_2 = \) distance from \( A_j \) to

\( j \)th column \( \uparrow \)

\( \text{Sp} \{ A_1, \ldots, A_9 \} = \text{Sp} \{ Q_1, \ldots, Q_9 \} \)

(see (17) of lec. 6) \( \uparrow \)

Proof:

(i), (ii) - already done.

(iii) \( u_j = A_j - (P_{A_1} A_j + \ldots + P_{A_{j-1}} A_j) = A_j - P_{\text{sp}(A_1, \ldots, A_{j-1})} A_j \)

So the claim follows from this picture. q.e.d.
Now, as we said, some columns of $Q_0$ can be zero. They can be thrown out. Indeed:

$$Q_0 = \begin{bmatrix} u_1 & \ldots & u_q \end{bmatrix}, \quad R_0 = \begin{bmatrix} \tilde{r}_1 \\ \vdots \\ \tilde{r}_q \end{bmatrix}.$$ 

Some of these $= 0$

$$Q_0 \cdot R_0 = u_1 \tilde{r}_1 + \ldots + u_q \tilde{r}_q \quad \text{(by Eq. (8) of Sec. 3) (8)}$$

So we can throw out zero columns of $Q_0$ and their corresponding rows $R_0$. If $\text{rank}(A) = k$, we will have $k$ non-zero vectors $u_1, \ldots, u_k$ that span $\text{span } \mathcal{R}(A) = \mathcal{R}^1 \mathcal{R}^2 \ldots \mathcal{R}^k$. Then $Q_0|_{\text{reduced}} = p \times k$, $R_0|_{\text{reduced}} = k \times q$.

It is also convenient (will see later why) to normalize the columns $[u_1, \ldots, u_k]$ of $Q_0$ to have length 1:

$$\tilde{u}_j = \frac{u_j}{\|u_j\|} \quad \text{or} \quad u_j = \tilde{u}_j \cdot \|u_j\|.$$ 

Then:

$$A = Q_0 \cdot R_0 = \begin{bmatrix} u_1 & \ldots & u_k \end{bmatrix} \begin{bmatrix} \tilde{r}_1 \\ \vdots \\ \tilde{r}_k \end{bmatrix}$$

$$= \tilde{u}_1 \left( \|u_1\| \tilde{r}_1 \right) + \ldots + \tilde{u}_k \left( \|u_k\| \tilde{r}_k \right) \quad (9)$$

$$= \begin{bmatrix} \tilde{u}_1 & \ldots & \tilde{u}_k \end{bmatrix} \begin{bmatrix} \|u_1\| \tilde{r}_1 \\ \vdots \\ \|u_k\| \tilde{r}_k \end{bmatrix}$$

$$= QR.$$
Thus we have proved

**Thm. 2** (*key Thm. 5.82(b)*)

Any \( A = p \times q \) s.t. \( \text{rank}(A) = k \leq q \) can be written as

\[
A = QR
\]

\[
\begin{array}{c}
p \times q \\
p \times k \\
k \times q
\end{array}
\]

where:

(i) all columns of \( Q \) are orthonormal;

(ii) \( R \) is upper-\( \Delta \) and \( \text{rank}(R) = k \);

(iii) \( |R_{ij}| = \|u_j \| = \text{distance from } A_j \text{ to } \text{Sp}\{B_1, \ldots, B_{j-1}\} \).

**Important Note:** Eq. (9),

\[
[A_1, \ldots, A_q] = \tilde{u}_1 \tilde{p}_1 + \cdots + \tilde{u}_k \tilde{p}_k
\]

says that any column of \( A \) is a lin. combination of columns \( \tilde{u}_1, \ldots, \tilde{u}_k \) of \( Q \),

\[
\Rightarrow \text{all } \text{Sp}\{A_1, \ldots, A_q\} = \text{Sp}\{\tilde{u}_1, \ldots, \tilde{u}_k\}, \text{ or}
\]

\[
\text{R}(A) = \text{R}(Q)
\]

**Ex. 5.83** & **5.84** — on your own.

3. **Application of the QR decomposition to the least-squares fit problem.**

In lecture 6 we showed that if

\[
A \times x \approx b
\]

is an inconsistent system of equations, then the vector \( a \) that best approximates
\( x \) in the LS sense satisfies
\[ A \hat{a} = P_A b \quad (13) \]
where \( P_A b \) is the projection of \( b \) on the
span \( \text{span} \{ A_1, \ldots, A_q \} \). (See Eq. (10) in Lecture 6.)

Computationally, this QR can be found by
solving the normal eqn:
\[ A^T A \hat{a} = A^T b \quad (14) \]
(we explained that we couldn't cancel by \( A^T \)).
As you will see in the new problem,
solving (14) can be sensitive to small changes
in \( b \) when columns of \( A \) are nearly l.d.
However, if one knows the QR decomposition
of \( A \), then (13) can be solved in an
efficient and robust way. (Recall that
the traditional Gram-Schmidt is not an
efficient method to find the QR decomposition;
e.g., Matlab uses other methods.)

We now show how (13) can be
solved using \( A = QR \).

\textbf{Step 1:} Recall (see Eq. (11)) that \( \mathcal{R}(A) = \mathcal{R}(Q) \).
Then \( P_A b = P_Q b \).

\textbf{Step 2:}
\[ P_Q b = \underbrace{Q Q^T b}_{\text{projection matrix on the orthogonal set of columns of } Q} \]
Step 3  \[ Aq = P_A \mathbf{b} \implies QRa = QQ^T \mathbf{b} \implies (\text{Eq. (4)}) \]
\[ Q^TQ Ra = Q^TQ Q^T \mathbf{b} \]
\[ \mathbf{I}_q \quad \mathbf{I}_q \]
\[ Ra = Q^T \mathbf{b} \quad \text{(15)} \]

Since \( R \) is upper-\( \Delta \), (15) can be easily solved by back-substitution.

See Ex. (5.87) on your own.

You may also see an alternative derivation of (15) on p. 240.