Least-Squares Systems and The QR factorization

- Orthogonality
- Least-squares systems.
- The Gram-Schmidt and Modified Gram-Schmidt processes.
- The Householder QR and the Givens QR.

Orthogonality

1. Two vectors $u$ and $v$ are orthogonal if $(u, v) = 0$.
2. A system of vectors $\{v_1, \ldots, v_n\}$ is orthogonal if $(v_i, v_j) = 0$ for $i \neq j$; and orthonormal if $(v_i, v_j) = \delta_{ij}$.
3. A matrix is orthogonal if its columns are orthonormal.

Notation: $V = [v_1, \ldots, v_n] = $ matrix with column-vectors $v_1, \ldots, v_n$.

Orthogonality is essential in understanding and solving least-squares problems.

Least-Squares systems

- Given: an $m \times n$ matrix $n < m$. Problem: find $x$ which minimizes:

$$\|b - Ax\|_2$$

- Good illustration: Data fitting.

Typical problem of data fitting: We seek an unknown function as a linear combination $\phi$ of $n$ known functions $\phi_i$ (e.g. polynomials, trig. functions). Experimental data (not accurate) provides measures $\beta_1, \ldots, \beta_m$ of this unknown function at points $t_1, \ldots, t_m$. Problem: find the 'best' possible approximation $\phi$ to this data.

$$\phi(t) = \sum_{i=1}^{n} \xi_i \phi_i(t) \ , \ s.t. \ \phi(t_j) \approx \beta_j, j = 1, \ldots, m$$

Question: Close in what sense?

- Least-squares approximation: Find $\phi$ such that

$$\phi(t) = \sum_{i=1}^{n} \xi_i \phi_i(t), \ & \sum_{j=1}^{m} |\phi(t_j) - \beta_j|^2 = \text{Min}$$

- In linear algebra terms: find 'best' approximation to a vector $b$ from linear combinations of vectors $f_i, i = 1, \ldots, n$, where

$$b = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_m \end{pmatrix}, \ f_i = \begin{pmatrix} \phi_i(t_1) \\ \phi_i(t_2) \\ \vdots \\ \phi_i(t_m) \end{pmatrix}$$
We want to find $x = \{\xi_i\}_{i=1,\ldots,n}$ such that
\[ \left\| \sum_{i=1}^{n} \xi_i f_i - b \right\|_2 \]
Minimum
Define
\[ F = [f_1, f_2, \ldots, f_n], \quad x = \begin{pmatrix} \xi_1 \\ \vdots \\ \xi_n \end{pmatrix} \]
We want to find $x$ to minimize $\|b - Fx\|_2$
This is a Least-squares linear system: $F$ is $m \times n$, with $m \geq n$.

Formulate the least-squares system for the problem of finding the polynomial of degree 2 that approximates a function $f$ which satisfies $f(-1) = -1; f(0) = 1; f(1) = 2; f(2) = 0$

Theorem. The vector $x^*$ minimizes $\psi(x) = \|b - Fx\|_2^2$ if and only if it is the solution of the normal equations:
\[ FTFx = Ft \]

Proof: Expand out the formula for $\psi(x^* + \delta x)$:
\[ \psi(x^* + \delta x) = ((b - Fx^*) - F\delta x)^T((b - Fx^*) - F\delta x) \]
\[ = \psi(x^*) - 2(F\delta x)^T(b - Fx^*) + (F\delta x)^T(F\delta x) \]
\[ = \psi(x^*) - 2(\delta x)^T(F^T(b - Fx^*)) + (F\delta x)^T(F\delta x) \]
\[ \leq \psi(x^*) \quad \text{always positive} \]
Can see that $\psi(x^* + \delta x) \geq \psi(x^*)$ for any $\delta x$, iff the boxed quantity [the gradient vector] is zero. Q.E.D.

Solution: $\phi_1(t) = 1; \phi_2(t) = t; \phi_2(t) = t^2$
- Evaluate the $\phi_i$’s at points $t_1 = -1; t_2 = 0; t_3 = 1; t_4 = 2$:
\[ f_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad f_2 = \begin{pmatrix} -1 \\ 0 \\ 1 \\ 4 \end{pmatrix}, \quad f_3 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} \]
- So the coefficients $\xi_1, \xi_2, \xi_3$ of the polynomial $\xi_1 + \xi_2 t + \xi_3 t^2$
are the solution of the least-squares problem $\min \|b - Fx\|$ where:
\[ F = \begin{pmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} -1 \\ 1 \\ 2 \end{pmatrix} \]

Illustration of theorem: $x^*$ is the best approximation to the vector $b$ from the subspace $\text{span}\{F\}$ if and only if $b - Fx^*$ is $\perp$ to the whole subspace $\text{span}\{F\}$. This in turn is equivalent to $F^T(b - Fx^*) = 0$ ➤ Normal equations.
1) Approximations by polynomials of degree one:

- \( \phi_1(t) = 1, \phi_2(t) = t \).
- \( F^T F = \begin{pmatrix} 5.0 & 0 \\ 0 & 2.5 \end{pmatrix} \)
- \( F^T b = \begin{pmatrix} 0.9 \\ -0.15 \end{pmatrix} \)

- Best approximation is \( \phi(t) = 0.18 - 0.06t \).

2) Approximation by polynomials of degree 2:

- \( \phi_1(t) = 1, \phi_2(t) = t, \phi_3(t) = t^2 \).
- Best polynomial found:

\[
0.3085714285 - 0.06 \times t - 0.2571428571 \times t^2
\]

**Problem with Normal Equations**

- Condition number is high: if \( A \) is square and non-singular, then

\[
\kappa_2(A) = \|A\|_2 \cdot \|A^{-1}\|_2 = \sigma_{\text{max}}/\sigma_{\text{min}}
\]

\[
\kappa_2(A^T A) = \|A^T A\|_2 \cdot \|(A^T A)^{-1}\|_2 = (\sigma_{\text{max}}/\sigma_{\text{min}})^2
\]

- Example: Let \( A = \begin{pmatrix} 1 & 1 & -\epsilon \\ \epsilon & 0 & 1 \\ 0 & \epsilon & 1 \end{pmatrix} \).

- Then \( \kappa(A) \approx \sqrt{2}/\epsilon \), but \( \kappa(A^T A) \approx 2\epsilon^{-2} \).

- \( fl(A^T A) = fl \begin{pmatrix} 2 + \epsilon^2 & 1 & 0 \\ 1 & 1 + \epsilon^2 & 0 \\ 0 & 0 & 1 + \epsilon^2 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \)

is singular to working precision (if \( \epsilon < u \)).
Finding an orthonormal basis of a subspace

Goal: Find vector in \( \text{span}(X) \) closest to \( b \).

Much easier with an orthonormal basis for \( \text{span}(X) \).

Problem: Given \( X = [x_1, \ldots, x_n] \), compute \( Q = [q_1, \ldots, q_n] \) which has orthonormal columns and s.t. \( \text{span}(Q) = \text{span}(X) \).

Note: each column of \( X \) must be a linear combination of certain columns of \( Q \).

We will find \( Q \) so that \( x_j \) (\( j \) column of \( X \)) is a linear combination of the first \( j \) columns of \( Q \).

\[ X = QR \]

\( R \) is upper triangular, \( Q \) is orthogonal. This is called the QR factorization of \( X \).

Another decomposition:
A matrix \( X \), with linearly independent columns, is the product of an orthogonal matrix \( Q \) and a upper triangular matrix \( R \).

ALGORITHM: 1. Classical Gram-Schmidt

1. For \( j = 1, \ldots, n \) Do:
2. Set \( \hat{q} := x_j \)
3. Compute \( r_{ij} := (\hat{q}, q_i) \), for \( i = 1, \ldots, j - 1 \)
4. For \( i = 1, \ldots, j - 1 \) Do:
5. Compute \( \hat{q} := \hat{q} - r_{ij}q_i \)
6. EndDo
7. Compute \( r_{jj} := \|\hat{q}\|_2 \),
8. If \( r_{jj} = 0 \) then Stop, else \( q_j := \hat{q}/r_{jj} \)
9. EndDo

All \( n \) steps can be completed iff \( x_1, x_2, \ldots, x_n \) are linearly independent. \( \square \) Prove this result.
The operations in lines 4 and 5 can be written as
\[ \tilde{q} := \text{ORTH}(\hat{q}, q_i) \]
where \( \text{ORTH}(x, q) \) denotes the operation of orthogonalizing a vector \( x \) against a unit vector \( q \).

Result of \( z = \text{ORTH}(x, q) \)

**Modified Gram-Schmidt algorithm is much more stable than classical Gram-Schmidt in general.** [A few examples easily show this].

Suppose MGS is applied to \( A \) yielding computed matrices \( \hat{Q} \) and \( \hat{R} \). Then there are constants \( c_i \) (depending on \( (m, n) \)) such that

\[
A + E_1 = \hat{Q} \hat{R} \quad \| E_1 \|_2 \leq c_1 \| A \|_2 \\
\| \hat{Q}^T \hat{Q} - I \|_2 \leq c_2 \| A \|_2 \| \kappa_2(A) \| + O((\| A \|_2 \| \kappa_2(A) \|)^2)
\]

for a certain perturbation matrix \( E_1 \), and there exists an orthonormal matrix \( Q \) such that

\[
A + E_2 = Q \hat{R} \quad \| E_2(:,j) \|_2 \leq c_3 \| A(:,j) \|_2
\]

for a certain perturbation matrix \( E_2 \).

**An equivalent version:**

**ALGORITHM : 3, Modified Gram-Schmidt - 2 -

0. Set \( \hat{Q} := X \)
1. For \( i = 1, \ldots, n \) Do:
2. Compute \( r_{ii} := \| \hat{q}_i \|_2 \)
3. If \( r_{ii} = 0 \) then Stop, else \( q_i := \hat{q}_i/r_{ii} \)
4. For \( j = i + 1, \ldots, n \), Do:
5. \( r_{ij} := (\hat{q}_j, q_i) \)
6. \( \hat{q}_j := \hat{q}_j - r_{ij} q_i \)
7. EndDo
8. EndDo

**Does exactly the same computation as previous algorithm, but in a different order.**
Example:
Orthonormalize the system of vectors:

\[ X = [x_1, x_2, x_3] = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \]

Answer:
\[ q_1 = \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix}; \quad q_2 = x_2 - (x_2, q_1)q_1 = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix} - 1 \times \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} = \begin{pmatrix} 1/2 \\ -1/2 \\ -1/2 \end{pmatrix} \]

\[ \hat{q}_3 = x_3 - (x_3, q_1)q_1 = \begin{pmatrix} 0 \\ 1 \\ -1 \\ 4 \end{pmatrix} - 2 \times \begin{pmatrix} \frac{1}{2} \\ 1/2 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -1 \\ -2 \\ 3 \end{pmatrix} \]

\[ \hat{q}_3 = \hat{q}_3 - (\hat{q}_3, q_2)q_2 = \begin{pmatrix} 1/2 \\ -1/2 \\ -1/2 \\ 2.5 \end{pmatrix} - (-1) \times \begin{pmatrix} \frac{1}{2} \\ 1/2 \\ 1/2 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -1/2 \\ -1/2 \\ -2.5 \end{pmatrix} \]

\[ \|\hat{q}_3\|_2 = \sqrt{13} \rightarrow q_3 = \frac{\hat{q}_3}{\|\hat{q}_3\|_2} = \begin{pmatrix} 1/2 \\ -1/2 \\ -1/2 \\ 2.5 \end{pmatrix} \]

Application: another method for solving linear systems.

Ax = b
A is an \( n \times n \) nonsingular matrix. Compute its QR factorization.

Multiply both sides by \( Q^T \) to get \( Q^TQRx = Q^Tb \).

Method:

- Compute the QR factorization of \( A \), \( A = QR \).
- Solve the upper triangular system \( Rx = Q^Tb \).

Cost??
**Use of the QR factorization**

Problem: \( Ax \approx b \) in least-squares sense

\( A \) is an \( m \times n \) (full-rank) matrix. Let \( A = QR \)

the QR factorization of \( A \) and consider the normal equations:

\[
A^T Ax = A^T b \quad \rightarrow \quad R^T Q^T R Rx = R^T Q^T b \\
R^T Rx = R^T Q^T b \quad \rightarrow \quad Rx = Q^T b
\]

\((R^T\) is an \( n \times n \) nonsingular matrix\). Therefore,

\[
x = R^{-1} Q^T b
\]

**Another derivation:**

- Recall: \( \text{span}(Q) = \text{span}(A) \)
- So \( \|b - Ax\|_2^2 \) is minimum when \( b - Ax \perp \text{span}\{Q\} \)
- Therefore solution \( x \) must satisfy \( Q^T (b - Ax) = 0 \)

\[
Q^T (b - QRx) = 0 \quad \rightarrow \quad Rx = Q^T b
\]

\[
x = R^{-1}Q^Tb
\]

**Method:**

- Compute the QR factorization of \( A \), \( A = QR \).
- Compute the right-hand side \( f = QTb \)
- Solve the upper triangular system \( Rx = f \).
- \( x \) is the least-squares solution

As a rule it is not a good idea to form \( A^T A \) and solve the normal equations. Methods using the QR factorization are better.

\[
\text{Total cost??} \quad (\text{depends on the algorithm used to get the QR decomposition}).
\]

Using matlab find the parabola that fits the data in previous data fitting example (p. 8-10) in L.S sense [verify that the result found is correct.]

\[
\|b - Ax\|_2^2 = \|b - QRx\|^2 \\
= \|(I - QQ^T)b + Q(Q^Tb - Rx)\|^2 \\
= \|(I - QQ^T)b\|^2 + \|Q(Q^Tb - Rx)\|^2 \\
= \|(I - QQ^T)b\|^2 + \|Q^Tb - Rx\|^2
\]

Min is reached when 2nd term of r.h.s. is zero.