Case Studies B: QR decomposition

QR decomposition Can be used as tools for

I: Solving Linear Systems

$$Ax = b, \quad A \in \mathbb{R}^{n\times n}, \quad x, b \in \mathbb{R}^n$$

When the QR of A is computed,

$$QRx = b,$$

where $Q \in \mathbb{R}^{n\times n}$ is orthogonal and $R \in \mathbb{R}^{n\times n}$ is upper triangular

Then

$$Ax = b \iff \underbrace{R}_{\text{upper triangular}}x = Q^T b$$

easy to solve by back substitution.

More cost and more stable than Gaussian Elimination.

II: Least Squares problem

Consider a linear system $Ax = b$, where $A \in \mathbb{R}^{m\times n}$.

If $m > n$, we have more equations than unknowns. So typically, there is no solution of $Ax = b$ if $m > n$. In this case, we expect a solution such that the error of $Ax = b$ is as small as possible.

That is, we solve

$$\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2$$

This problem is called least squares problem.

Theorem: x is a solution of $\min_x \|Ax-b\|_2^2$

$$\iff A^T A x = A^T b \quad \text{(Called normal}$$

$$\iff (b - Ax) \perp \operatorname{Ran}(A)$$

Proof. x is a solution of $\min_{x \in \mathbb{R}^n} \|Ax - b\|_2^2$

$$\Leftrightarrow \quad \forall y \in \mathbb{R}^n, \quad \|A(x-y)-b\|_2^2 \geq \|Ax-b\|_2^2$$

$$\Leftrightarrow \forall y \in \mathbb{R}^n, \quad \|(Ax-b)-Ay\|_2^2 = \|Ax-b\|_2^2 - 2\langle Ax-b, Ay\rangle + \|Ay\|_2^2 \geq \|Ax-b\|_2^2$$

$$\Leftrightarrow \forall y \in \mathbb{R}^n, \quad \langle Ax-b, Ay\rangle \leq \frac{1}{2}\|Ay\|_2^2$$

$$\Leftrightarrow \forall y \in \mathbb{R}^n, \quad \langle Ax-b, Ay\rangle = 0 \quad \Leftrightarrow \quad \langle b-Ax, z\rangle = 0 \quad \forall z \in Ran(A)$$

proof: "$\Leftarrow$" is obvious.

"$\Rightarrow$". replace y by $ty \in \mathbb{R}^n$, we still have

$$t\langle Ax-b, Ay\rangle = \langle Ax-b, A(ty)\rangle \leq \frac{1}{2}\|A(ty)\|_2^2 = \frac{1}{2}t^2\|Ay\|_2^2$$

Choose $t \to 0_+$, $\quad \langle Ax-b, Ay\rangle \leq \frac{t}{2}\|Ay\|_2^2 \to 0$

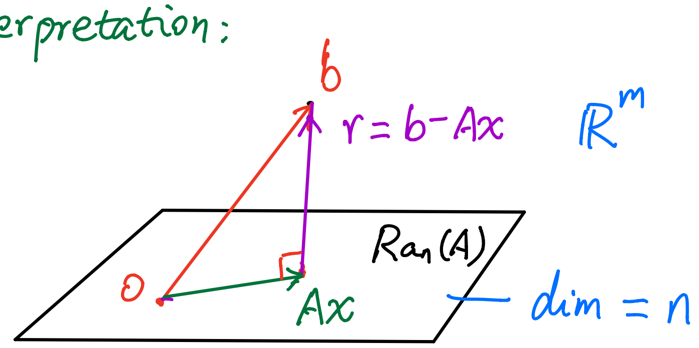
Choose $t \to 0_-$, $\quad \langle Ax-b, Ay\rangle \geq \frac{t}{2}\|Ay\|_2^2 \to 0$

Thus, $\quad 0 \leq \langle Ax-b, Ay\rangle \leq 0$

$$\Leftrightarrow \forall y \in \mathbb{R}^n, \quad \langle A^T(Ax-b), y\rangle = 0 \quad \Leftrightarrow \quad A^T(Ax-b) = 0$$

$$\Leftrightarrow \quad A^TAx = A^Tb \qquad \boxtimes$$

Geometric Interpretation:



Now, to solve the least squares problem, we only need to solve the normal equation:

$$A^TAx = A^Tb$$

We assume $rank(A) = n$ so that A^TA is SPD

Step 1: Compute A^TA (needs $2mn^2$ flops)

Step 2: Compute A^Tb (needs $2mn$ flops)

Step 3: Solve $A^TAx = A^Tb$ by, e.g., Cholesky decomposition)
(needs $\frac{1}{3}n^3 + O(n^2)$ flops)

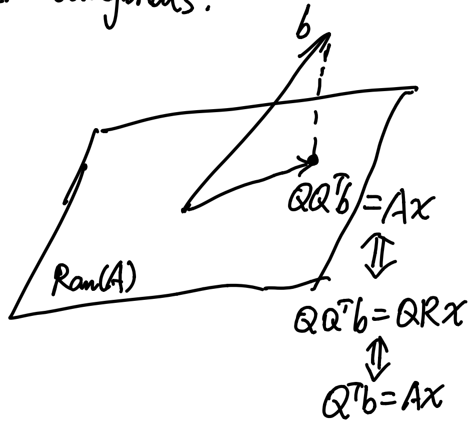
We can use QR decomposition provide another solution of least squares problem.

Assume $m > n$ and $\operatorname{rank}(A) = n$. Then, there is a QR decomposition

$$A = QR,$$

where $Q \in \mathbb{R}^{m \times n}$ is orthogonal (i.e. $Q^TQ = I$) and

$R \in \mathbb{R}^{n \times n}$ is upper triangular with non-zero diagonals.



Then, $$\|Ax - b\|_2^2 = \|QRx - b\|_2^2$$
$$= \|(QRx - QQ^Tb) - (I - QQ^T)b\|_2^2$$

Since $\langle QRx - QQ^Tb, (I - QQ^T)b \rangle$
$$= \langle Rx - Q^Tb, \underbrace{Q^T(I - QQ^T)b}_{=0} \rangle = 0$$

By using Pythagorean Theorem,

$$\|Ax - b\|_2^2 = \|Q(Rx - Q^Tb)\|_2^2 + \|(I - QQ^T)b\|_2^2$$
$$= \|Rx - Q^Tb\|_2^2 + \|(I - QQ^T)b\|_2^2 \quad \text{Constant of } x$$
$$\left(\text{Since } \|Qy\|_2^2 = \langle Qy, Qy \rangle = \langle y, Q^TQy \rangle = \langle y, y \rangle = \|y\|_2^2\right)$$

Therefore, $\min_x \|Ax - b\|_2^2 \iff \min_x \|Rx - Q^Tb\|_2^2$.

Since R has non-zero diagonals, R is invertible. Thus, $\|Rx - Q^Tb\|_2^2$ is minimized when we solve

$$Rx = Q^Tb.$$

The equation can be solved by back-substitution.

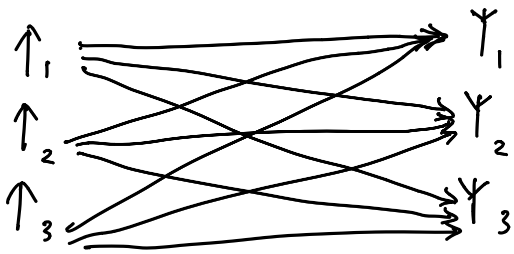
Step 1: Compute $A = QR$ — $2mn^2$ operations if we use modified G-S

Step 2: Compute Q^Tb — $2mn$ operations

Step 3: Solve $Rx = Q^Tb$ by back substitution. — $O(n^2)$ operations

Application Cases I: Multiple input and Multiple Output (MIMO)

In radio, MIMO means using multiple anttenas at both transmitter and receiver to improve transmission performances



If we want to transmit a signal x, then each transmitter anttena will transmit an entry through the communication network to each receiver anttena. What a receiver antenna gets is a weighted sum of entries of x. In a compact form, the process can be modeled as

$$b = Ax + e,$$

where x is the signal to be transmitted. e represent noise

A represents the wireless network b the received measurement

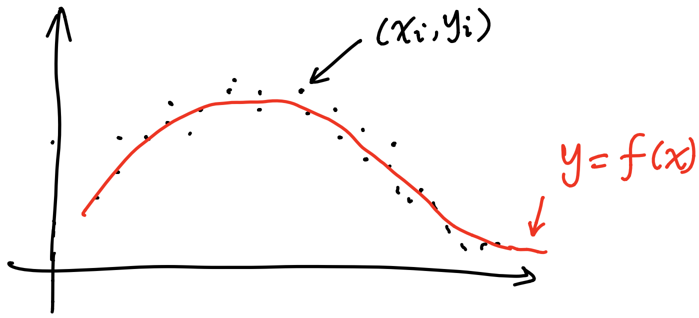
So naturally we solve

$$\min_x \|b - Ax\|_2^2$$

to reconstruct x from b. This is a least squares problem and can be solved by e.g., QR decomposition.

Application Case II: Linear regression

Goal: Given m pairs of points (x_i, y_i), $i = 1, 2, \cdots, m$, fit a curve passing through these points to explore the relationship between x and y. i.e., find a function f s.t. $y \approx f(x)$.



To find f, we solve

$$\min_f \sum_{i=1}^{m} (y_i - f(x_i))^2,$$

i.e., we want each pair (x_i, y_i) is close to the curve $y=f(x)$.

The problem is not tractable, because we search f from the set of ALL functions, which is too large. Then, we consider $f(x)$ that can be expressed by n $(n<m)$ basis functions $\{\psi_i\}_{i=1}^{n}$ as follows:

$$f(x) = \sum_{j=1}^{n} c_j \psi_j(x).$$

We only need to find $c = \begin{pmatrix} c_1 \\ \vdots \\ c_n \end{pmatrix} \in \mathbb{R}^n$. Thus, the minimization becomes

$$\min_{c\in\mathbb{R}^n} \sum_{i=1}^{m} \Big(y_i - \sum_{j=1}^{n} c_j \psi_j(x_i) \Big)^2$$

If we define

$$A = \begin{bmatrix} \psi_1(x_1) & \psi_2(x_1) & \cdots & \psi_n(x_1) \\ \psi_1(x_1) & \psi_2(x_2) & \cdots & \psi_n(x_2) \\ \vdots & \vdots & \vdots & \vdots \\ \psi_1(x_m) & \psi_2(x_m) & \cdots & \psi_n(x_m) \end{bmatrix} \in \mathbb{R}^{m\times n}, \quad c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} \in \mathbb{R}^n, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \in \mathbb{R}^m$$

then the minimization is rewritten as

$$\min_{c\in\mathbb{R}^n} \|Ac - y\|_2^2,$$

which is a least squares problem and can be solved by QR decomposition.

Results: We choose polynomials, i.e.,

$$\psi_1(x)=1 \quad \psi_2(x)=x, \quad \psi_3(x)=x^2, \quad \cdots, \quad \psi_n(x)=x^{n-1}$$

Be careful of over-fitting.

(e.g., for the pic on right: when $n=11$, the error is small but the curve is not what we want)