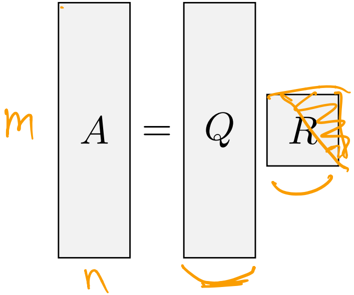
QR factorisation

For any $A \in \mathbb{C}^{m\times n}$, $\exists$ factorisation

$m \geq n$

$Q^*Q = I_n$

$Q^TQ = I_n$ if A real.



$Q \in \mathbb{R}^{m\times n}$: orthonormal, $R \in \mathbb{R}^{n\times n}$: upper triangular

- Many algorithms available: Gram-Schmidt, Householder, CholeskyQR, ...
- various applications: least-squares, orthogonalisation, computing SVD, manifold retraction...

  $\min \|Ax - b\|_2$ (QR) vs $Ax = b$ (via PLU)
- With Householder, pivoting $A = QRP$ not needed for numerical stability
  - but pivoting gives rank-revealing QR (nonexaminable)

QR via Gram-Schmidt

Gram-Schmidt: Given $A = [a_1, a_2, \dots, a_n] \in \mathbb{R}^{m\times n}$ (assume full rank $\text{rank}(A) = n$), find orthonormal $[q_1, \dots, q_n]$ s.t. $\text{span}(q_1, \dots, q_n) = \text{span}(a_1, \dots, a_n)$

Q m n $Q^TQ = I_n$

recall C-F

G-S process: $q_1 = \frac{a_1}{\|a_1\|}$, then $\tilde{q}_2 = a_2 - q_1 \underbrace{q_1^T a_2}_{\text{scalar}}$, $q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_2\|}$,

repeat for $j = 3, \dots, n$: $\tilde{q}_j = a_j - \sum_{j=1}^{j-1} q_j q_j^T a_j$, $q_j = \frac{\tilde{q}_j}{\|\tilde{q}_j\|}$.

$\sigma_i(A) = \min_{\dim(S)=i} \max_{x \in S, \|x\|=1} \|Ax\|_2$

$\Leftrightarrow \min_{Q^TQ = I_i} \max_{x = Qy, \|y\|=1} \|Ax\|_2$.

$q_1^T q_2 = 0$. m $[q_1\ q_2]$ $[q_1\ \tilde{q}_2]$

$\tilde{q}_j^T q_i = 0$

for $i = 1, \dots, j-1$

2 $= \begin{bmatrix} 1 & 0 \\ 0 & \|\tilde{q}_2\|^2 \end{bmatrix}$

$[q_1\ q_2]^T [q_1\ q_2] = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

$[q_1\ q_2 \dots q_{j-1}\ \tilde{q}_j]^T [\ \ ''\ \ ] = \begin{bmatrix} 1 & & 0 \\ & \ddots & \\ 0 & & \|\tilde{q}_j\|^2 \end{bmatrix}$

QR via Gram-Schmidt

Gram-Schmidt: Given $A = [a_1, a_2, \dots, a_n] \in \mathbb{R}^{m\times n}$ (assume full rank $\mathrm{rank}(A) = n$), find orthonormal $[q_1, \dots, q_n]$ s.t. $\mathrm{span}(q_1, \dots, q_n) = \mathrm{span}(a_1, \dots, a_n)$

G-S process: $q_1 = \frac{a_1}{\|a_1\|}$, then $\tilde{q}_2 = a_2 - q_1 q_1^T a_2$, $q_2 = \frac{\tilde{q}_2}{\|\tilde{q}_2\|}$,

repeat for $j = 3, \dots, n$: $\tilde{q}_j = a_j - \sum_{j=1}^{j-1} q_j q_j^T a_j$, $q_j = \frac{\tilde{q}_j}{\|\tilde{q}_j\|}$.

what if 0?

This gives QR! Let $r_{ij} = q_i^T a_j$ $(i \neq j)$ and $r_{jj} = \|a_j - \sum_{i=1}^{j-1} r_{ij} q_i\|$,

$$\begin{bmatrix} q_1 = \frac{a_1}{r_{11}} \\ q_2 = \frac{a_2 - r_{12}q_1}{r_{22}} \\ q_j = \frac{a_j - \sum_{i=1}^{j-1} r_{ij}q_i}{r_{jj}} \end{bmatrix} \Leftrightarrow \begin{bmatrix} a_1 = r_{11}q_1 \\ a_2 = r_{12}q_1 + r_{22}q_2 \\ a_j = r_{1j}q_1 + r_{2j}q_2 + \dots + r_{jj}q_j \end{bmatrix} \Leftrightarrow A = QR$$

$a_1 \cdots a_n$ a_j

q_1 q_j

$q_1 \cdots q_n$

Wait 10^{-15} ish

$\|Q^TQ - I_n\| = O(\epsilon)$

but GS $\approx \kappa_2(A)^2 \epsilon \geq 1$

- But this isn't the recommended way to do QR; numerically unstable

Householder reflectors

$$\boxed{H = I - 2vv^T,} \quad \|v\| = 1$$

$\in \mathbb{R}^{n\times n}$ $\forall$ $v \in \mathbb{R}^n$ $\mathbb{C}^n$

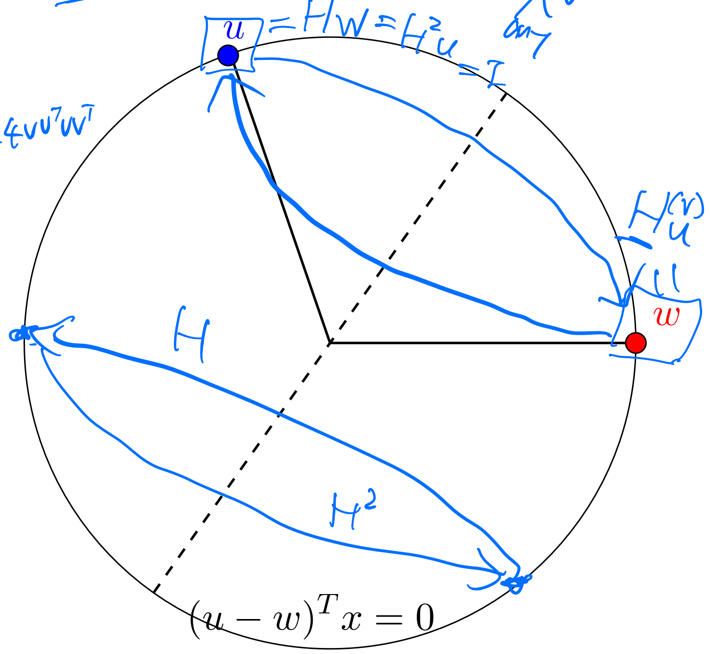
$\Leftrightarrow H = I - 2\frac{\tilde{v}\tilde{v}^T}{\|\tilde{v}\|^2}$ for any $\tilde{v}$

- H orthogonal and symmetric: $H^T H = H^2 = I$, eigvals 1 ($n-1$ copies) and -1 (1 copy)

  $H^TH = (I - 2vv^T)^2 = I - 4vv^T + 4vv^Tvv^T = I$

  s.v. $\sigma_i(H) = 1$

- For any given $u, w \in \mathbb{R}^n$ s.t. $\|u\| = \|w\|$ and $u \neq w$ $H = I - 2vv^T$ with $v = \frac{w-u}{\|w-u\|}$ gives $\boxed{H}u = w$ ($\Leftrightarrow u = Hw$, thus 'reflector')
- We'll use this mostly for $w = [*, 0, 0, \ldots, 0]^T$

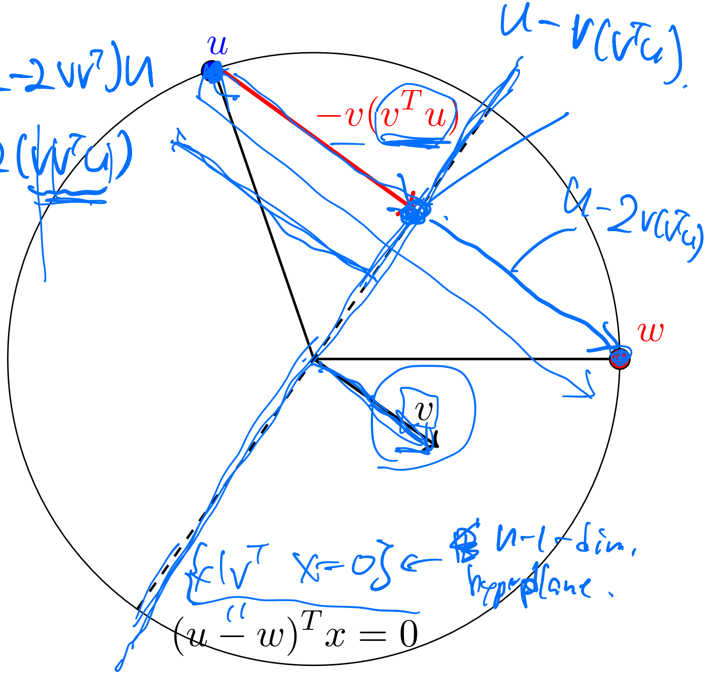


Householder reflectors

$$H = I - 2vv^T, \qquad \|v\| = 1$$

- H orthogonal and symmetric: $H^T H = H^2 = I$, eigvals 1 ($n-1$ copies) and -1 (1 copy)
- For any given $u, w \in \mathbb{R}^n$ s.t. $\|u\| = \|w\|$ and $u \neq v$, $H = I - 2vv^T$ with $v = \frac{w-u}{\|w-u\|}$ gives $Hu = w$ ($\Leftrightarrow u = Hw$, thus 'reflector')
- We'll use this mostly for $w = [*, 0, 0, \dots, 0]^T$

$Hu = (I - 2vv^T)u$
$= u - 2v(v^T u)$



$$w = \begin{bmatrix} * \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Householder reflectors for QR

$v = \frac{w-u}{\|w-u\|_2}$

Householder reflectors:

$$H = I - 2vv^T, \qquad v = \frac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2}, \qquad e = [1, 0, \ldots, 0]^T$$

satisfies $Hx = [\|x\|, 0, \ldots, 0]^T$

$$H \begin{bmatrix} x \\ \vdots \\ x \end{bmatrix} = \begin{bmatrix} \|x\| \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

since $\|Qy\|_2 = \|y\|_2$

Householder reflectors for QR

Householder reflectors:

$$H = I - 2vv^T, \qquad v = \frac{x - \|x\|_2 e}{\|x - \|x\|_2 e\|_2}, \qquad e = [1, 0, \ldots, 0]^T$$

satisfies $Hx = [\|x\|, 0, \ldots, 0]^T$

$\Rightarrow$ To do QR, find H_1 s.t. $H_1 a_1 = \begin{bmatrix} \|a_1\|_2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$,

$H_n \cdots H_2 H_1 A =$

repeat to get $H_n \cdots H_2 H_1 A = R$ upper triangular, then

$A = (H_1 \cdots H_{n-1} H_n) R = QR$

$H_i^2 = I$

$H_i^{-1} = H_i^T = H_i$

Householder QR factorisation, diagram

$$A = \begin{bmatrix} * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \\ * & * & * & * \end{bmatrix}$$

Apply sequence of Householder reflectors

different

$$H_1 A = (I - 2v_1 v_1^T)A = \begin{bmatrix} * & * & * & * \\ & * & * & * \\ & * & * & * \\ & * & * & * \\ & * & * & * \end{bmatrix}, \qquad H_2 H_1 A = (I - 2v_2 v_2^T) H_1 A = \begin{bmatrix} * & * & * & * \\ & * & * & * \\ & & * & * \\ & & * & * \\ & & * & * \end{bmatrix},$$

$\hat{a}_2$

$v_2 = \hat{a}_2 - \begin{bmatrix} \|\hat{a}_2\| \\ 0 \\ \vdots \\ 0 \end{bmatrix}$

$H_i = I$ if already in desired form.

$$H_3 H_2 H_1 A = \begin{bmatrix} * & * & * & * \\ & * & * & * \\ & & * & * \\ & & & * \\ & & & * \end{bmatrix}, \qquad H_n \cdots H_3 H_2 H_1 A = \begin{bmatrix} * & * & * & * \\ & * & * & * \\ & & * & * \\ & & & * \\ & & & \end{bmatrix},$$

$v_3 = \begin{bmatrix} 0 \\ 0 \\ * \end{bmatrix}$, $\hat{a}_3 - \begin{bmatrix} \|\hat{a}_3\| \\ 0 \\ 0 \end{bmatrix}$

m, n, 0

Note $v_k = [\underbrace{0, 0, \ldots, 0}_{k-1 \text{ 0's}}, *, *, \ldots, *]^T$

Householder QR factorisation

$$H_n \cdots H_2 H_1 A = \begin{bmatrix} * & * & * & * \\ & * & * & * \\ & & * & * \\ & & & * \\ & & & \\ & & & \end{bmatrix} = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

full SVD

$[u_1 \cdots u_n]$ U $[U \; U_\perp]$

Householder QR for U gives $U = [Q_F]\begin{bmatrix} R \\ 0 \end{bmatrix}$

$$\Leftrightarrow A = (H_1^T \cdots H_{n-1}^T H_n^T) \begin{bmatrix} R \\ 0 \end{bmatrix} =: Q_F \begin{bmatrix} R \\ 0 \end{bmatrix}$$ (**full** QR; Q_F is square orthogonal)

Writing $Q_F = [Q \; Q_\perp]$ where $Q \in \mathbb{R}^{m\times n}$ orthonormal, $A = QR$ (**'thin'** QR or just QR)

n $m-n$

Properties

- Cost $\frac{4}{3}n^3$ flops with Householder-QR (twice that of LU) $n\times n$ $\frac{2}{3}n^3 + O(n^2)$
- Unconditionally backward stable: $\hat{Q}\hat{R} = A + \Delta A$, $\|\hat{Q}^T\hat{Q} - I\|_2 = \epsilon$ (next lec) $\frac{\|\Delta A\|}{\|A\|} = \epsilon$
- Constructive proof for $A = QR$ existence
- To solve $Ax = b$, solve $Rx = Q^T b$ via triangle solve.

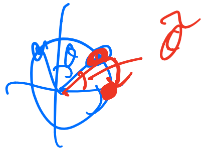
  $\rightarrow$ Excellent method, but twice slower than LU (so rarely used)

  $QRx = b$

Givens rotation

$Q_k \cdots Q_2 Q_1 A \to$ (simple form).

$$G = \begin{bmatrix} c & s \\ -s & c \end{bmatrix}, \quad c^2 + s^2 = 1$$



Designed to 'zero' one element at a time. E.g. QR for upper Hessenberg matrix

$$\begin{bmatrix} G_1 & 0 \\ 0 & I \end{bmatrix}$$

$$A = \begin{bmatrix} * & * & * & * & * \\ * & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & & * & * \end{bmatrix}, \quad G_1 A = \begin{bmatrix} * & * & * & * & * \\ & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & & * & * \end{bmatrix}, \quad G_2 G_1 A = \begin{bmatrix} * & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & * & * & * \\ & & & * & * \end{bmatrix},$$

updated

intact!

$$\begin{bmatrix} G & \\ & I \end{bmatrix}$$

$$G_3 G_2 G_1 A = \begin{bmatrix} * & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & & * & * \\ & & & * & * \end{bmatrix}, \quad G_4 G_3 G_2 G_1 A = \begin{bmatrix} * & * & * & * & * \\ & * & * & * & * \\ & & * & * & * \\ & & & * & * \\ & & & & * \end{bmatrix} =: R$$

$\to O(n^2)$ flops, not n^3

$\Leftrightarrow A = G_1^T G_2^T G_3^T G_4^T R$ is the QR factorisation.

$\iff A = G_1^T G_2^T G_3^T G_4^T R$.

QR!

- G acts locally on two rows (two columns if right-multiplied)
- Non-neighboring rows/cols allowed

AG

$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} \; \end{bmatrix}$ intact.

Least-squares problem

Given $A \in \mathbb{R}^{m\times n}, m \geq n$ and $b \in \mathbb{R}^m$, find $x \in \mathbb{R}^n$ s.t.

lin sys

$\boxed{A}x = b$

det(A) ≠ 0

$\sigma_{min}(A) > 0$

$$\min_x \left\| \boxed{A} \, \boxed{x} - \boxed{b} \right\|_2$$

m

n

m>n

- More data than degrees of freedom
- 'Overdetermined' linear system; $Ax = b$ usually impossible
- Thus minimise $\|Ax - b\|$; usually $\|Ax - b\|_2$ but sometimes e.g. $\|Ax - b\|_1$ of interest (we focus on $\|Ax - b\|_2$)

residual

$\sum_i |Ax-b|_i$ compressed sensing

- Assume full rank rank$(A) = n$; this makes solution unique

$\iff \sigma_{min}(A) = \sigma_n(A) > 0$

$\cancel{\sigma_n(A) = \sigma_m(A) = 0}$

Least-squares problem via QR

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, m \ge n$$

$$A_m = \underbrace{[Q \; Q_\perp]}_{m} \begin{bmatrix} R \\ 0 \end{bmatrix} \quad \text{(e.g. Householder QR).}$$

$$\|Ax - b\|_2 = \left\| [Q \; Q_\perp] \begin{bmatrix} R \\ 0 \end{bmatrix} x - [b] \right\|_2$$

Left mult. $[Q \; Q_\perp]^T$

$$= \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x - \begin{bmatrix} Q^T b \\ Q_\perp^T b \end{bmatrix} \right\|_2$$

vec.

for any x,

$$= \left\| \begin{bmatrix} Rx - Q^T b \\ -Q_\perp^T b \end{bmatrix} \right\|_2$$

$$\min_x \|Ax - b\|_2 = \min_x \left\| \begin{matrix} Rx - Q^T b \\ -Q_\perp^T b \end{matrix} \right\|_2$$

no x in it!

$Rx - Q^T b = 0$ via $Rx = Q^T b$

nonsing. upper triang.

$$A = \boxed{Q}\boxed{R} = \boxed{Q} U_R \Sigma_R V_R^T = U \Sigma V^T$$

$\Sigma_R = \Sigma$

$(\sigma_{min}(A) > 0$

$\sigma_{min}(R) > 0$

Least-squares problem via QR

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, m \geq n$$

Let $A = [Q\ Q_\perp]\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right] = Q_F\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right]$ be 'full' QR factorization. Then

$$\|Ax - b\|_2 = \|Q_F^T(Ax - b)\|_2 = \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x - \begin{bmatrix} Q^T b \\ Q_\perp^T b \end{bmatrix} \right\|_2$$

so $\boxed{x = R^{-1}Q^T b}$ is solution. This also gives algorithm:

$A = \boxed{Q}R$ is enough!

full QR vs "thin" QR cheaper

$O(m^3)$ $O(mn^2)$

m n

Least-squares problem via QR

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, m \geq n$$

Let $A = [Q\ Q_\perp]\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right] = Q_F\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right]$ be 'full' QR factorization. Then

$$\|Ax - b\|_2 = \|Q_F^T(Ax - b)\|_2 = \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x - \begin{bmatrix} Q^T b \\ Q_\perp^T b \end{bmatrix} \right\|_2$$

so $x = R^{-1}Q^T b$ is solution. This also gives algorithm:

1. Compute **thin** QR factorization $A = QR$
2. Solve linear system $Rx = Q^T b$.

Least-squares problem via QR

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, m \geq n$$

Let $A = [Q\ Q_\perp]\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right] = Q_F\left[\begin{smallmatrix} R \\ 0 \end{smallmatrix}\right]$ be 'full' QR factorization. Then

$$\|Ax - b\|_2 = \|Q_F^T(Ax - b)\|_2 = \left\| \begin{bmatrix} R \\ 0 \end{bmatrix} x - \begin{bmatrix} Q^T b \\ Q_\perp^T b \end{bmatrix} \right\|_2$$

LU useless

so $x = R^{-1}Q^T b$ is solution. This also gives algorithm:

1. Compute **thin** QR factorization $A = QR$
2. Solve linear system $Rx = Q^T b$.

- This is backward stable: computed $\hat{x}$ solution for $\min_x \|(A + \Delta A)x + (b + \Delta b)\|_2$ (see Higham's book Ch.20)
- Unlike square system $Ax = b$, one really needs QR: LU won't do the job

Normal equation: Cholesky-based least-squares solver

$$\min_x \|Ax - b\|_2, \qquad A \in \mathbb{R}^{m\times n}, m \geq n$$

$x = R^{-1}Q^Tb$ is the solution $\Leftrightarrow$ x solution for $n \times n$ **normal equation**

$$(A^TA)x = A^Tb \Leftrightarrow A^T(Ax-b)=0$$

- $A^TA \succeq 0$ (always) and $A^TA \succ 0$ if $\text{rank}(A) = n$; then PD linear system; use Cholesky to solve. $\frac{1}{3}n^3$
- Fast! but NOT backward stable; $\kappa_2(A^TA) = (\kappa_2(A))^2$ where $\kappa_2(A) = \frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}$ **condition number** (next lecture)

Gram matrix

$10^6 \to 10^{12}$

$2 \to 4$

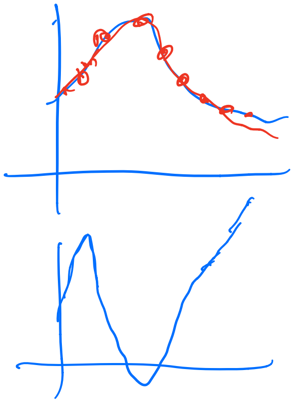
$A = U\Sigma V^T \quad A^TA = V\Sigma^2 V^T$

$\sigma_1^2 \cdots \sigma_n^2$

$\frac{\sigma_1^2}{\sigma_n^2} = \left(\frac{\sigma_1}{\sigma_n}\right)^2$

fine if $\frac{\sigma_1}{\sigma_n} = O(1)$

Application: regression/function approximation



Given function $f : [-1, 1] \to \mathbb{R}$,

Consider approximating via polynomial $f(x) \approx p(x) = \sum_{i=0}^{n} c_i x^i$.

Very common technique: **Regression**

1. Sample f at points $\{z_i\}_{i=1}^{m}$, and prescribed.
2. Find coefficients c defined by Vandermonde system $Ac \approx f$,

$$\min \left\| \begin{bmatrix} f(z_i) - p(z_i) \end{bmatrix}_{i=1}^{i=m} \right\|_2 \qquad \begin{bmatrix} 1 & z_1 & \cdots & z_1^n \\ 1 & z_2 & \cdots & z_2^n \\ \vdots & \vdots & & \vdots \\ 1 & z_m & \cdots & z_m^n \end{bmatrix} \begin{bmatrix} c_0 \\ \vdots \\ c_n \end{bmatrix} \approx \begin{bmatrix} f(z_1) \\ f(z_2) \\ \vdots \\ f(z_m) \end{bmatrix}.$$

$m \geq n+1$

$\min_x \| A x - b \|_2$

$x = R^{-1} Q^T b$

- Numerous applications, e.g. in statistics, numerical analysis, approximation theory, data analysis!