Least squares with QR factorization

Note Title 2024-11-08

For any $A \in \mathbb{R}^{m \times n}$ $(m \geq n)$, there exist Q orthogonal, R triangular such that

$$A = QR = Q_0 R_0 \qquad Q = \begin{bmatrix} \overbrace{Q_0}^{n} & \overbrace{Q_c}^{m-n} \end{bmatrix} \qquad R = \begin{bmatrix} R_0 \\ 0 \end{bmatrix} \begin{matrix} \} n \\ \} m-n \end{matrix}$$

Least squares problem: $\min_{x \in \mathbb{R}^n} \|Ax - y\|_2$

$A \in \mathbb{R}^{m \times n}$, $y \in \mathbb{R}^m$ given, $\quad x = \underbrace{(A^T A)^{-1} A^T}\, y$

A^+, pseudoinverse

We obtain another solution method from $A = QR$

$$\|Ax - y\| = \|QRx - y\| = \|Q^T(QRx - y)\| = \|Rx - Q^T y\|$$

$$\left\| \begin{bmatrix} R_0 \\ 0 \end{bmatrix} x - \begin{bmatrix} Q_0^T \\ Q_c^T \end{bmatrix} y \right\| = \left\| \begin{bmatrix} R_0 x - Q_0^T y \\ -Q_c^T y \end{bmatrix} \right\|$$ } bottom part does not depend on x

$\left\| \begin{bmatrix} R_0 x - Q_0^T y \\ -Q_c^T y \end{bmatrix} \right\|^2$ is always greater or equal than $\|-Q_c^T y\|^2$

It's always strictly larger, unless $R_0 x - Q_0^T y = 0$

We can choose $x = R_0^{-1} Q_0^T y$, this must give the minimum possible norm, since the first block is 0

- the minimum value of $\|Ax - y\|$ is $\|Q_c^T y\|$
- the minimizer is $x = R_0^{-1} Q_0^T y$. We can compute it with only the thin QR factorization.

Algorithm:

1. Compute thin QR $A = Q_0 R_0$ $O(mn^2)$
2. Compute $Q_0^T y$ $O(mn)$
3. Solve the $n \times n$ triangular system $R_0 x = Q_0^T y$ $O(n^2)$

Total complexity: $O(mn^2)$

the expensive part 1 depends on A only

$\Rightarrow$ savings for multiple problems with the same A.

Also: $A^+ = R_0^{-1} Q_0^T$, formula for the pseudo inverse.

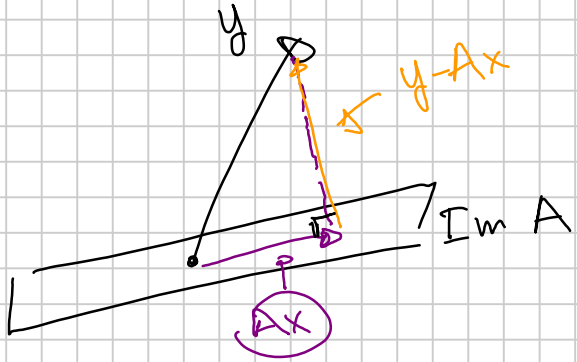
Result: R_0 is invertible if and only if A has full column rank.

A has full col. rank $\Leftrightarrow$ $A^T A$ is invertible

$\Leftrightarrow (Q_0 R_0)^T (Q_0 R_0) = R_0^T \underbrace{Q_0^T Q_0}_{I} R_0 = R_0^T R_0$ is invertible

$\Leftrightarrow$ R_0 invertible

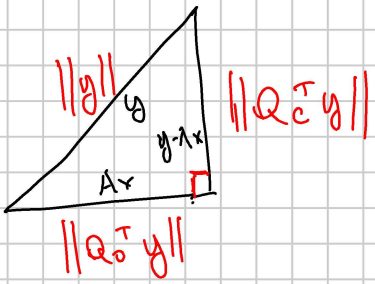
[**Remark**: the factorization $A^T A = R_0^T R_0 =$  is known as Cholesky factorization]



The closest vector Ax inside $\text{Im } A$ to y is the orthogonal projection of y onto the plane, and

it makes a right angle.

$$Ax = Q_0 R_0 R_0^{-1} Q_0^T y = Q_0 Q_0^T y \qquad \|Ax\|_2 = \|Q_0^T y\|_2$$



We can obtain a similar algorithm using the SVD of the matrix A.

$$A = USV^T = U_0 S_0 V^T$$

We replay the same steps as the QR version:

$$\|Ax-y\| = \|US\underbrace{V^T x}_{z} - y\| = \|U^T(USz-y)\| = \|Sz - U^T y\|$$

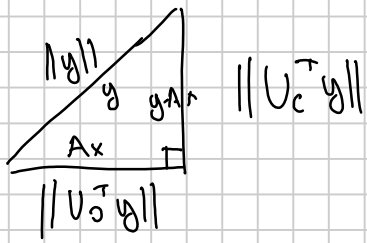
$$\left\| \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \\ \hline 0 & & \end{bmatrix} z - \begin{bmatrix} U_0^T \\ U_c^T \end{bmatrix} y \right\| = \left\| \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_n \end{bmatrix} z - U_0^T y \right\| \quad \} \text{ can become zero}$$
$$\begin{bmatrix} -U_c^T y \end{bmatrix} \quad \} \text{ constant}$$

the first block becomes 0 if $z = S_0^{-1} U_0^T y$

$$z = V^T x \iff x = Vz = V S_0^{-1} U_0^T y$$

This gives an expression for the solution x in terms of the thin SVD.

$$A^+ = V S_0^{-1} U_0^T$$



$Ax = U_O U_O^T y$ has norm $\|U_O^T y\|$ $\quad S_O = \begin{bmatrix} \sigma_1 & & 0 \\ & \ddots & \\ 0 & & \sigma_n \end{bmatrix}$

$$x = V S_O^{-1} U_O^T y = \begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{bmatrix} \begin{bmatrix} \frac{1}{\sigma_1} & & 0 \\ & \frac{1}{\sigma_2} & \\ 0 & & \ddots \\ & & & \frac{1}{\sigma_n} \end{bmatrix} \begin{bmatrix} u_1^T \\ u_2^T \\ \vdots \\ u_n^T \end{bmatrix} y$$

$$= v_1 \frac{1}{\sigma_1} u_1^T y + v_2 \frac{1}{\sigma_2} u_2^T y + \cdots + v_n \frac{1}{\sigma_n} v_n^T y$$

When is S_O invertible?

A has full col. rank $\Leftrightarrow$ $A^T A$ is invertible $\Leftrightarrow$

$$(U_O S_O V^T)^T (U_O S_O V^T) = V S_O^T \underbrace{U_O^T U_O}_{I} S_O V^T = V \begin{bmatrix} \sigma_1^2 & & 0 \\ & \sigma_2^2 & \\ 0 & & \ddots \\ & & & \sigma_n^2 \end{bmatrix} V^T$$

is invertible $\Leftrightarrow$ All the σ_i's are different from 0.

$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$, so this is equivalent to $\sigma_n \neq 0$.

Rank-deficient case

$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \sigma_{r+2} = \cdots = \sigma_n = 0$ $\quad S_O = \begin{bmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & \ddots & \\ & & & & 0 \end{bmatrix}$

$x = Vz$

$$\|Ax-y\| = \|Sz - U^T y\| = \left\| \begin{bmatrix} \sigma_1 & & & \\ & \ddots & & \\ & & \sigma_r & \\ & & & 0 \\ & & & & \ddots \\ & & & & & 0 \end{bmatrix} \begin{bmatrix} z_1 \\ \vdots \\ z_n \end{bmatrix} - \begin{bmatrix} u_1^T y \\ u_2^T y \\ \vdots \\ u_n^T y \\ \vdots \\ u_m^T y \end{bmatrix} \right\|$$

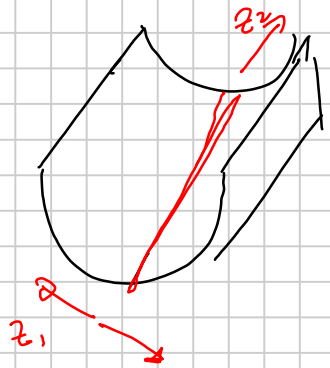
$$= \left\| \begin{bmatrix} \sigma_1 z_1 - u_1^T y \\ \sigma_2 z_2 - u_2^T y \\ \vdots \\ \sigma_r z_r - u_r^T y \\ 0 - u_{r+1}^T y \\ \vdots \\ 0 - u_n^T y \\ -u_{n+1}^T y \\ \vdots \\ -u_m^T y \end{bmatrix} \right\|$$

independent of x

This norm is minimized whenever

$$z_1 = \frac{1}{\sigma_1} u_1^T y$$
$$\vdots$$
$$z_r = \frac{1}{\sigma_r} u_r^T y$$
$$z_{r+1} = \text{anything}$$
$$\vdots$$
$$z_n = \text{anything}$$

$x = Vz$



Infinite solutions:

$z_{r+1}, \dots z_n$ can be chosen arbitrarily.

Among all solutions, the ones with the smallest $\|x\| = \|z\|$ is obtained by taking $z_{r+1} = \dots = z_n = 0$.

Result: Suppose $A = USV^T$ has zero singular values $\sigma_{r+1} = \dots = \sigma_n = 0$. Then, the solution of

$$\min_{\{x : \|Ax-y\| \text{ is minimum}\}} \|x\|$$

is given by

$$x = V \begin{bmatrix} \frac{1}{\sigma_1} u_1^T y \\ \vdots \\ \frac{1}{\sigma_r} u_r^T y \\ 0 \\ \vdots \\ 0 \end{bmatrix} = v_1 \frac{1}{\sigma_1} u_1^T y + \cdots + v_r \frac{1}{\sigma_r} u_r^T y$$

$$= \sum_{i=1}^{r} v_i \frac{1}{\sigma_i} u_i^T y$$

(Same formula as the solution of the full col. rank problem, only that we stop the sum at r instead of n.

(One can generalize the def. of the pseudoinverse to

$$A^+ = V \begin{bmatrix} \frac{1}{\sigma_1} & & & \\ & \ddots & & \\ & & \frac{1}{\sigma_r} & \\ & & & 0 \ddots 0 \end{bmatrix} U_0^T \text{, so that } x = A^+ y \Big)$$

$A =$  ← data non-maximum rank $\Leftrightarrow$ linear dependence between data columns.

$$A = \begin{bmatrix} 1 & 0.1 \\ 2 & 0.2 \\ 3 & 0.29999 \end{bmatrix}$$ ← very sensitive to noise!

- noise
- inexact computation ($\Leftrightarrow$ (relative) perturbation of size $u \approx 10^{-16}$, if you are using a backward stable algorithm).

Theorem: if $\sigma_1, \dots \sigma_n$ are the sing. values of $A \in \mathbb{R}^{m \times n}$,

and $\tilde{\sigma}_1, \dots \tilde{\sigma}_n$ are those of $A+E$, for an error $E \in \mathbb{R}^{m \times n}$ then

$$\|\tilde{\sigma}_i - \sigma_i\| \leq \|E\|_2$$

(Matlab example)

$$x = \sum v_i \frac{1}{\sigma_i} u_i^T y$$

small singular values have larger relative error due to noise, and this has a larger impact on the solution

Possible solution: if some singular values are very small (under a certain threshold), we stop the sum early:

$$x_{trunc} = \sum_{i=1}^{k} v_i \frac{1}{\sigma_i} u_i^T y$$ truncated SVD solution.

This works well in practical problems, because small singular values $\Leftrightarrow$ small features, indistinguishable from errors.

(threshold $\approx$ noise or via grid search).

[Matlab example: computing svd(A) via eig(A^TA)$^{1/2}$ is much less accurate, because the eigenvalue computation incurs in an error $\approx u \cdot \|A^TA\|$, larger than the error $\approx u \cdot \|A\|$ incurred with SVD].

```
>> svd(A)
ans =
     2.603199427398562e+04
     3.136819638199822e+03
     1.113915022155416e+03
     1.532139580657542e-12
>> eig(A'*A) .^ (1/2)
ans =
     2.780555342303793e-04
     1.113915022155438e+03
     3.136819638199822e+03
     2.603199427398563e+04
>> norm(A)
ans =
     2.603199427398562e+04
>> norm(A'*A)
ans =
     6.776647258808204e+08
```