MAT 653: Statistical Simulation

Instructor: Dr. Wei Li

2023-08-27

Back **Updated:** 2023-08-27 Statistical Simulation, Wei Li

Solving Linear Systems

Suppose we have a linear system $Ax = b$, where $A \in \mathbb{R}^{m\times n}$ known, $b \in \mathbb{R}^{m\times 1}$ known , and $x \in \mathbb{R}^{n\times 1}$ unknown. Our goal is: given A and b, want to find x.

We may encounter the following questions

1. (Existence) Is there any solution?

2. (Uniqueness) When there is an unique solution? (i.e. under what condition?)

3. What can one do if there are multiple solutions (implying infinitely many).

4. What can one do if there is no solution?

To answer the **Question 1**, we introduce the following result:

Theorem: $Ax = b$ has a solution $x \iff b \in Col(A)$.

Check:

Based on this, we have the following result to answer the **Question 2**:

Theorem: Suppose $b \in Col(A)$, then $Ax = b$ has a unique solution $\iff$ columns of A are all linearly independent ($m \geq n$).

Proof:

Special case: If A is square matrix that has linearly independent columns, then there exists a unique solution to $Ax = b$ for any $b \in \mathbb{R}^m$, which is $x = A^{-1}b$.

Turning to question 3, if there are multiple solutions, one may characterize the solution space completely, or when a unique solution is desired, regularization technique can be applied or additional constraint may be imposed. This depends on the applications and will not be further discussed here. An alternative is to cast the problem as a least square minimization problem and then solve for the solution with some regularization, which will be covered under the question 4.

Sovling a linear system

Given $A \in \mathbb{R}^{m\times n}$, **assuming there exists one unique solution**, then **how to solve for** x. The well-known methods include:

- Gaussian elimination
- Cholesky factorization
- QR decomposition
- SVD (singular value decomposition)

We are not going to discuss Gaussian elimination here. You can find more details in any algebra textbook.

Cholesky factorization

Theorem (Cholesky factorization/decomposition): If symmetric matrix $A > 0$ (positive-definite), then there exists a unique upper triangular matrix U s.t. $A = U^\top U$.

Application: Given $A > 0$ (hence invertible), solve for $Ax = b$. There are three steps in the algorithm.

1. Compute $A = U^\top U$. (Then we have $U^\top Ux = b$, denote $Ux = y$, now $U^\top y = b$)
2. Forward solve $U^\top y = b$ to get $\hat{y}$. (Here $U^\top$ is a lower triangular matrix)
3. Back solve $Ux = \hat{y}$ to get $\hat{x}$. (Here U is an upper triangular matrix)

Usage: In R, we use *chol(A)* to implement Cholesky decomposition to matrix A. If we have $Ux = b$ where U is an upper triangular matrix, we use $x \leftarrow backsolve(U, b)$. And if we have $Lx = b$ where L is a lower triangular matrix, we use $x \leftarrow forwardsolve(L, b)$

QR decomposition

Theorem (*Thin/Reduced* QR decomposition): If $A \in \mathbb{R}^{m\times n}$ $(m \geq n)$ has n linearly independent columns, then $A = Q_1R$, where $Q_1 \in \mathbb{R}^{m\times n}$ has orthonormal columns (hence $Q_1^\top Q_1 = I$), and $R \in \mathbb{R}^{n\times n}$ is an upper triangular positive definite matrix.

Proof:

Remark: The matrix Q_1 provides an orthonormal basis for $Col(A)$, i.e., the columns of A are linear combinations of the columns of Q_1. In fact, we have $Col(A) = Col(Q_1)$, and any partial set of columns satisfy the same property, i.e.,

$$span\{q_1, \cdots, q_k\} = span\{a_1, \cdots, a_k\} \quad k = 1, 2, \cdots, n$$

And from the proof of the theorem we can tell $q_k \perp \{a_1, a_2, \cdots, a_{k-1}\}, k \geq 2$.

Theorem (*Full* QR decomposition): With $A = Q_1R$ as above, write

$$A = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$$

where $Q \in \mathbb{R}^{m\times m}$ is orthogonal matrix (i.e. $Q^\top Q = QQ^\top = I$), Q_2 is a matrix containing columns all orthogonal to that of Q_1 and $\begin{bmatrix} R \\ 0 \end{bmatrix} \in \mathbb{R}^{m\times n}$.

Remark: Actually,

$$A = Q \begin{bmatrix} R \\ 0 \end{bmatrix} = Q_1R + Q_2 0$$

Note that to make sure $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$ is orthogonal, we are appending additional orthonormal columns to Q_1 to get Q, which means columns of Q_2 are orthonormal, orthogonal to Q_1. Then we also know columns of Q_2 orthogonal to that of A, since $A = Q_1 R$.

Example: Consider $A = \begin{bmatrix} a_1 & a_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$, find thin and full QR decomposition of A, i.e. $A = Q_1 R = Q \begin{bmatrix} R \\ 0 \end{bmatrix}$.

$$q_1 = \frac{a_1}{\|a_1\|} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \qquad q_2 = \frac{a_2 - \langle a_2, q_1 \rangle q_1}{\|a_2 - \langle a_2, q_1 \rangle q_1\|} == \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$A = \begin{bmatrix} a_1 & a_2 \end{bmatrix} = \begin{bmatrix} q_1 & q_2 \end{bmatrix} R = \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} \|a_1\| & q_1^\top a_2 \\ 0 & \|a_2 - \langle a_2, q_1 \rangle q_1\| \end{bmatrix}$$

$$A = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 \\ 0 & 1 \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & 1 \end{bmatrix} \qquad \textit{(Thin QR)}$$

To get full QR, we want to find normalized $q_3 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$ s.t. $q_3 \perp q_1$, $q_3 \perp q_2$.

From $\langle q_3, q_1 \rangle = 0$, we have $\frac{1}{\sqrt{2}} x + \frac{1}{\sqrt{2}} z = 0$, and from $\langle q_3, q_2 \rangle = 0$, we have $y = 0$. Then we can fix $x = 1$ to get $y = 0$, $z = -1$, normalize the vector to get $q_3 = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ -\frac{1}{\sqrt{2}} \end{bmatrix}$. Then we have

$$A = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix} \begin{bmatrix} R \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & 0 \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \qquad \textit{(Full QR)}$$

Application: Solving linear system $Ax = b$, where $A \in \mathbb{R}^{m \times n}$ has linearly independent columns (hence the system has a unique solution). There are three steps in the algorithm.

1. Do QR decomposition (thin version) to get $A = Q_1 R$.
2. Then $Q_1 R x = b \implies Rx = Q_1^\top b$. Denote $y = Q_1^\top b$, we can compute y.
3. Back solve $Rx = y$ to get $\hat{x}$.

Solving Linear Least Squares

Goal: $Ax = b$, given A and b, to find x.

When there is no solution: $b \notin \mathrm{Col}(A)$

In this case, a natural problem to consider is the linear least square problem:

$$\min_x \|Ax - b\|$$

Our strategy is to find a $\hat{x}$ such that $A\hat{x} = \hat{b}$ is a projection of b on Col(A). Then $\hat{x}$ is an approximate solution of the original problem $Ax = b$, but only in the sense that Ax is close to b in L_2 norm. Note however, this is from the point of view of optimization. From statistical point of view, least square problem is primarily driven by the goal to minimize some "expected" error – mean-squared error between random objects.

Consider $\hat{b} = A\hat{x}$. Obtain $\hat{b}$ by dropping a perpendicular line from b to Col(A), that is to say $\langle r, \hat{a}\rangle = 0, \hat{a} \in$ Col(A), where $r = b - \hat{b}$ is the residual vector. In particular,

$$\begin{aligned} \langle r, a_i\rangle = 0, i = 1, 2, \cdots, n &\iff A^T r = 0 \\ &\iff A^T(b - A\hat{x}) = 0 \\ &\iff A^T A\hat{x} = A^T b \end{aligned}$$

Now $\hat{x} = (A^T A)^{-1} A^T b$ if $A^T A$ is invertible, and then $\hat{b} = A(A^T A)^{-1} A^T b$ (even if it exists).

Note: $A^T A\hat{x} = A^T b$ is called Normal Equation.

Fact 1: Normal equation always has at least one solution.

Fact 2: $A^T A$ is invertible if and only if A has full-column rank.

Fact 3: The solution to the normal equation is not necessary a solution to the original problem $Ax = b$.

Example:

Regularization (ridge regression): Solve $(A^T A + \lambda I)x = A^T b$ for $\lambda > 0$ is called. In this case, $A^T A + \lambda I$ is always invertible, and we have

$$\hat{x} = (A^T A + \lambda I)^{-1} A^T b.$$

Application of QR to solve a linear least square problem: $\min_x \|Ax - b\|$.

A columns are linearly independent. $A^T Ax = A^T b$. Decompose matrix A by using QR decomposition:

$$\begin{aligned} A = Q_1 R \Rightarrow R^T Q_1^T Q_1 R x &= R^T Q_1^T b \\ \Leftrightarrow R^T R x &= R^T Q_1^T b \\ \Leftrightarrow (R^T)^{-1} R^T R x &= (R^T)^{-1} R^T Q_1^T b \\ \Leftrightarrow R x &= Q_1^T b = y \end{aligned}$$

Algorithm:

1. Compute thin QR: $A = Q_1 R$.
2. Compute the vector: $y = Q_1^T b$.
3. Backsolve the $Rx = y$ to obtain $\hat{x}$.

Note: R command: solve(A,b), qr.solve(A,b)