NUMERICALLY EFFICIENT METHODS FOR SOLVING LEAST SQUARES PROBLEMS

DO Q LEE

Abstract. Computing the solution to Least Squares Problems is of great importance in a wide range of fields ranging from numerical linear algebra to econometrics and optimization. This paper aims to present numerically stable and computationally efficient algorithms for computing the solution to Least Squares Problems. In order to evaluate and compare the stability and efficiency of our proposed algorithms, the theoretical complexities and numerical results have been analyzed.

Contents

1. Introduction: The Least Squares Problem 2
2. Existence and Uniqueness 2
2.1. Least Squares Solution from Normal Equations 3
3. Norms and Conditioning 4
3.1. Norms: Quantifying Error and Distance 4
3.2. Sensitivity and Conditioning: Perturbations to b 5
3.3. Sensitivity to perturbations in A 6
4. Normal Equations Method 7
4.1. Cholesky Factorization 7
4.2. Flop: Complexity of Numerical Algorithms 7
4.3. Algorithm: Computing the Cholesky Factorization 8
4.4. Shortcomings of Normal Equations 8
5. Orthogonal Methods - The QR Factorization 8
5.1. Orthogonal Matrices 9
5.2. Triangular Least Squares Problems 9
5.3. The QR Factorization in Least Squares Problems 10
5.4. Calculating the QR-factorization - Householder Transformations 10
5.5. Rank Deficiency: Numerical Loss of Orthogonality 12
6. Singular Value Decomposition (SVD) 12
6.1. The Minimum Norm Solution using SVD 13
6.2. Computing the SVD of Matrix A 14
7. Comparison of Methods 14
8. Acknowledgements 15
References 15

Date: August 24, 2012.

1. Introduction: The Least Squares Problem

Suppose we are given a set of observed values β and $\alpha_1, \dots, \alpha_n$. Suppose the variable β is believed to have a linear dependence on the variables $\alpha_1, \dots, \alpha_n$. Then we may postulate a linear model

$$\beta = x_1\alpha_1 + \ldots + x_n\alpha_n.$$

Our goal is to determine the unknown coefficients $x_1, \dots, x_n$ so that the linear model is a best fit to our observed data. Now consider a system of m linear equations with n variables:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mn}x_n = b_m \end{cases}$$

Then we obtain an overdetermined linear system $Ax = b$, with $m \times n$ matrix A, where $m > n$. Since equality is usually not exactly satisfiable when $m > n$, the Least Squares Solution x minimizes the squared Euclidean norm of the residual vector $r(x) = b - Ax$ so that

$$\min \|r(x)\|_2^2 = \min \|b - Ax\|_2^2 \tag{1.1}$$

In this paper, numerically stable and computationally efficient algorithms for solving Least Squares Problems will be considered. Sections 2 and 3 will introduce the tools of orthogonality, norms, and conditioning which are necessary for understanding the numerical algorithms introduced in the following sections. The Normal Equations Method using Cholesky Factorization will be discussed in detail in section 4. In section 5, we will see that the QR Factorization generates a more accurate solution than the Normal Equations Method, making it one of the most important tools in computing Least Squares Solutions. Section 6 will discuss the Singular Value Decomposition (SVD) and its robustness in solving rank-deficient problems. Finally, we will see that under certain circumstances the Normal Equations Method and the SVD may be more applicable than the QR approach.

2. Existence and Uniqueness

In this section, we will see that the linear Least Squares Problem $Ax = b$ always has a solution, and this solution is unique if and only if the columns of A are linearly independent, i.e., $rank(A) = n$, where A is an $m \times n$ matrix. If $rank(A) < n$, then A is rank-deficient, and the solution is not unique. As such, the numerical algorithms in the later sections of this paper will be based on the assumption that A has full column rank n.

Definition 2.1. Let $S \subset \mathbb{R}^n$. The orthogonal complement of S, denoted as $S_\perp$, is the set of all vectors $x \in \mathbb{R}^n$ that are orthogonal to S.

One important property of orthogonal complements is the following:

$$\mathbb{R}^n = V \oplus V_\perp,$$

where $\oplus$ is the direct sum, which means that any vector $x \in \mathbb{R}^n$ can be uniquely represented as

$$x = p + o,$$

where $p \in V$ and $o \in V_\perp$. Then p is called the orthogonal projection of the vector x onto the subspace V. As a result, we have the following lemma:

Lemma 2.2. *Let $V \subset \mathbb{R}^n$. Let p be the orthogonal projection of a vector $x \in \mathbb{R}^n$ onto V. Then, $\|x - v\| > \|x - p\|$ for any $v \neq p \in V$.*

Proof. Let $o = x - p$, $o' = x - v$, and $v' = p - v$. Then $o' = o + v'$, $v' \in V$, and $v' \neq 0$. Since $o \perp V$, it follows that $o \cdot v' = 0$. Thus,

$$\begin{aligned}\|o'\|^2 = o' \cdot o' = (o + v') \cdot (o + v') &= o \cdot o + v' \cdot o + o \cdot v' + v' \cdot v' \\ &= o \cdot o + v' \cdot v' = \|o\|^2 + \|v'\|^2 > \|o\|^2.\end{aligned}$$

Thus $\|x - p\| = \min_{v \in V} \|x - v\|$ is the shortest distance from the vector x to V. □

2.1. **Least Squares Solution from Normal Equations.** Recall from (1.1) that the Least Squares Solution x minimizes $\|r(x)\|^2$, where $r(x) = b - Ax$ for $x \in \mathbb{R}^n$. The dimension of $span(A)$ is at most n, but if $m > n$, b generally does not lie in $span(A)$, so there is no exact solution to the Least Squares Problem. In our next theorem, we will use Lemma 2.2 to see that $Ax \in span(A)$ is closest to b when $r = b - Ax$ is orthogonal to $span(A)$, giving rise to the system of Normal Equations $A^T Ax = A^T b$.

Theorem 2.3. *Let A be an $m \times n$ matrix and $b \in \mathbb{R}^m$. Then $\hat{x}$ is a Least Squares Solution of the system $Ax = b$ if and only if it is a solution of the associated normal system $A^T Ax = A^T b$.*

Proof. Let $x \in \mathbb{R}^n$. Then Ax is an arbitrary vector in the column space of A, which we write as $R(A)$. As a consequence of Lemma 2.2, $r(x) = b - Ax$ is minimum if Ax is the orthogonal projection of b onto $R(A)$. Since $R(A)^\perp = Null(A^T)$, $\hat{x}$ is a Least Squares Solution if and only if

$$A^T r(\hat{x}) = A^T(b - A\hat{x}) = 0,$$

which is equivalent to the system of Normal Equations

$$A^T A\hat{x} = A^T b.$$

□

For this solution to be unique, the matrix A needs to have full column rank:

Theorem 2.4. *Consider a system of linear equations $Ax = b$ and the associated normal system $A^T Ax = A^T b$. Then the following conditions are equivalent:*

1. *The Least Squares Problem has a unique solution*
2. *The system $Ax = 0$ only has the zero solution*
3. *The columns of A are linearly independent.*

Proof. Let $\hat{x}$ be the unique Least Squares Solution and $x \in \mathbb{R}^n$ is such that $A^T Ax = 0$. Then

$$A^T A(\hat{x} + x) = A^T A\hat{x} = A^T b.$$

Thus $\hat{x} + x = \hat{x}$ i.e., $x = 0$ since the normal system has a unique solution. Also, this means that $A^T Ax = 0$ only has the trivial solution, so $A^T A$ is nonsingular ($A^T A$ is nonsingular if and only if $A^T Av \neq 0$ for all non-zero $v \in \mathbb{R}^n$).

Now it is enough to prove that $(A^T A)$ is invertible if and only if $rank(A) = n$. For $v \in \mathbb{R}^n$, we have $v^T(A^T A)v = (v^T A^T)(Av) = (Av)^T(Av) = \|Av\|_2^2$, so $A^T Av \neq 0$

if and only if $Av \neq 0$. For $1 \leq i \leq n$, denote the i-th component of v by v_i and the i-th column of A by a_i. Then Av is a linear combination of the columns of A, the coefficients of which are the components of v. It follows that $Av \neq 0$ for all non-zero v if and only if the n columns of A are linearly independent, i.e., A is full-rank. □

3. Norms and Conditioning

This section will first justify the choice of the Euclidean 2-norm from our introduction in (1.1). Then we will see how this choice of norm is closely linked to analyzing the sensitivity and conditioning of the Least Squares Problem.

3.1. **Norms: Quantifying Error and Distance.** A norm is a function that assigns a positive length to all the vectors in a vector space. The most common norms on $\mathbb{R}^n$ are

(1) The Euclidean norm: $\|x\|_2 = (\sum_{i=1}^{n} |x|^2)^{1/2}$,
(2) The p-norm: $\|x\|_p = (\sum_{i=1}^{n} |x|^p)^{1/p}$ for $p \geq 1$, and
(3) The Infinite norm: $\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|$.

Given so many choices of norm, it is natural to ask whether they are in any way comparable. It turns out that in finite dimensions, all norms are in fact equivalent to each other. The following theorem allows us to generalize Euclidean spaces to vector spaces of any finite dimension.

Theorem 3.1. *If S is a vector space of finite dimension, then all norms are equivalent on S i.e., for all pairs of norms $\|\cdot\|_n$, $\|\cdot\|_m$, there exist two constants c and C such that $0 < c \leq C$, and $\forall x \in S$, we have*

$$c\|x\|_n \leq \|x\|_m \leq C\|x\|_n$$

Proof. It is enough to show that any two norms are equivalent on the unit sphere of a chosen norm, because for a general vector $x \in \mathbb{R}^n$, we can write $x = \gamma x_0$, where $\gamma = \|x\|_n$ and x_0 is a vector on the unit sphere. We first prove that any norm is a continuous function: Suppose that $x_0 \in \mathbb{R}^n$. Then for every $x \in \mathbb{R}^n$, the triangle inequality gives $|\|x\| - \|x_0\|| \leq \|x - x_0\|$. Thus $|\|x\| - \|x_0\|| < \epsilon$ whenever $\|x - x_0\| < \epsilon$.

For simplicity, we work in the case $n = 2$. For the second inequality, let $\{e_i\}$ be the canonical basis in $\mathbb{R}^n$. Then write $x = \sum x_i e_i$ so that

$$\|x\|_m \leq \sum |x_i| \|e_i\|_m \leq \sqrt{\sum {x_i}^2} \sqrt{\sum {\|e_i\|_m}^2} = C\|x\|_2$$

For the first inequality, by using continuity on the unit sphere, it can be shown that the function $x \to \|x\|_m$ has a minimum value c on the unit sphere. Now write $x = \|x\|_2 \hat{x}$ so that

$$\|x\|_m = \|x\|_2 \|\hat{x}\|_m \geq c\|x\|_2.$$

□

With the equivalence of norms, we are now able to choose the 2-norm as our tool for the Least Squares Problem: solutions in the 2-norm are equivalent to solutions in any other norm.

The 2-norm is the most convenient one for our purposes because it is associated with an inner product. Once we have an inner product defined on a vector space, we can define both a norm and distance for the inner product space:

Definition 3.2. Suppose that V is an inner product space. The norm or length of a vector $u \in V$ is defined as

$$\|u\| = \langle u, u \rangle^{\frac{1}{2}}.$$

The equivalence of norms also implies that any properties of accuracy and stability are independent of the norm chosen to evaluate them. Now that we have a means of quantifying distance and length, we have the tools for quantifying the error for our Least Squares Solutions.

3.2. **Sensitivity and Conditioning: Perturbations to b.** In general, a non-square $m \times n$ matrix A has no inverse in the usual sense. But if A has full rank, a pseudoinverse can be defined as

$$A^+ = (A^T A)^{-1} A^T, \tag{3.3}$$

and condition number $\kappa(A)$ by

$$\kappa(A) = \|A\| \cdot \|A^+\|. \tag{3.4}$$

Combined with Theorem 2.3, the Least Squares Solution of $Ax = b$ can be given by $x = A^+ b$. We will soon see that if $\kappa(A) \gg 1$, small perturbations in A can lead to large errors in the solution.

Systems of linear equations are sensitive if a small perturbation in the matrix A or in the right-hand side b causes a significant change in the solution x. Such systems are called ill-conditioned. The following is an example of an ill-conditioned matrix with respect to perturbations on b.

Example 3.5. The linear system $Ax = b$, with

$$A = \begin{bmatrix} 1 & 1 \\ 1 & 1+\epsilon \end{bmatrix}, b = \begin{bmatrix} 2 \\ 2 \end{bmatrix},$$

where $0 < \epsilon \ll 1$ has the solution $x = \begin{bmatrix} 2 & 0 \end{bmatrix}^T$. We claim this system is ill-conditioned because small perturbations in b alter the solution significantly. If we compute the system $A\hat{x} = \hat{b}$, where

$$\hat{b} = \begin{bmatrix} 2 \\ 2+\epsilon \end{bmatrix}$$

we see this has the solution $\hat{x} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$, which is completely different from x.

One way to determine sensitivity to perturbations of b is to examine the relationship between error and the residual. Consider the solution of the system $Ax = b$, with expected answer x and computed answer $\hat{x}$. We will write the error e and the residual r as

$$e = x - \hat{x},\ r = b - A\hat{x} = b - \hat{b}.$$

Since x may not be obtained immediately, the accuracy of the solution is often evaluated by looking at the residual

$$r = b - A\hat{x} = Ax - A\hat{x} = Ae$$

We take the norm of e to get a bound for the absolute error

$$\|e\|_2 = \|x - \hat{x}\|_2 = \|A^+(b - \hat{b})\|_2 \leq \|A^+\|_2 \|b - \hat{b}\|_2 = \|A^+\|_2 \|r\|_2$$

so

$$\|e\|_2 \leq \|A^+\|_2 \|r\|_2.$$

Using this we can derive a bound for the relative error $\|e\|_2/\|x\|_2$ and $\|r\|_2/\|b\|_2$:

$$\|e\|_2 \leq \|A^+\|_2 \|r\|_2 \frac{\|Ax\|_2}{\|b\|_2} \leq \|A^+\|_2 \|A\|_2 \|x\|_2 \frac{\|r\|_2}{\|b\|_2}.$$

Thus

$$\frac{\|e\|_2}{\|x\|_2} \leq \kappa(A) \frac{\|r\|_2}{\|b\|_2}, \tag{3.6}$$

If $\kappa(A)$ is large (i.e., the matrix is ill-conditioned), then relatively small perturbations of the right-hand side b (and therefore the residual) may lead to even larger errors. For well-conditioned problems ($\kappa(A) \approx 1$) we can derive another useful bound:

$$\|r\|_2 \|x\|_2 = \|Ae\|_2 \|x\|_2 = \|Ae\|_2 \|A^+ b\|_2 \leq \|A\|_2 \|e\|_2 \|A^+\|_2 \|b\|_2$$

so that

$$\frac{1}{\kappa(A)} \frac{\|r\|_2}{\|b\|_2} \leq \frac{\|e\|_2}{\|x\|_2}. \tag{3.7}$$

Finally, combine (3.6) and (3.7) to obtain

$$\frac{1}{\kappa(A)} \frac{\|r\|_2}{\|b\|_2} \leq \frac{\|x - \hat{x}\|_2}{\|x\|_2} \leq \kappa(A) \frac{\|r\|_2}{\|b\|_2}.$$

These bounds are true for any A, but show that the residual is a good indicator of the error only if A is well-conditioned. The closer $\kappa(A)$ is to 1, the smaller the bound can become. On the other hand, an ill-conditioned A would allow for large variations in the relative error.

3.3. **Sensitivity to perturbations in A.** Small perturbations on ill-conditioned matrices can lead to large changes in the solution.

Example 3.8. For the linear system $Ax = b$, let

$$A = \begin{bmatrix} 1+\epsilon & 1-\epsilon \\ 1-\epsilon & 1+\epsilon \end{bmatrix}, \quad \Delta A = \begin{bmatrix} -\epsilon & \epsilon \\ \epsilon & -\epsilon \end{bmatrix}$$

with $0 < \epsilon \ll 1$. Then consider the perturbed matrix

$$\hat{A} = A + \Delta A = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}.$$

$\hat{A}$ is singular, so $\hat{A}\hat{x} = b$ has no solution.

Consider the linear system $Ax = b$, but now A is perturbed to $\hat{A} = A + \Delta A$. Denote by x the exact solution to $Ax = b$, and by $\hat{x}$ the solution to the perturbed Least Squares Problem $\hat{A}\hat{x} = b$. Now we can write $\hat{x} = x + \Delta x$ so that

$$\hat{A}\hat{x} = (A + \Delta A)(x + \Delta x) = b$$

so that

$$Ax - b + (\Delta A)x + A(\Delta x) + (\Delta A)(\Delta x) = 0$$

The term $(\Delta A)(\Delta x)$ is negligibly small compared to the other terms, so we get

$$(\Delta x) = -A^+(\Delta A)x.$$

Taking norms leads to the following result:

$$\|\Delta x\|_2 \leq \|A^+\|_2\|\Delta A\|_2\|x\|_2 = \|A^+\|_2\|A\|_2\frac{\|\Delta A\|_2}{\|A\|_2}\|x\|_2$$

or

$$\frac{\|x-\hat{x}\|_2}{\|x\|_2} \leq \kappa(A)\frac{\|A-\hat{A}\|_2}{\|A\|_2}.$$

4. Normal Equations Method

From Theorem 2.3, we have seen that finding the Least Squares Solution can be reduced to solving a system of Normal Equations $A^TAx = A^Tb$. The Normal Equations Method computes the solution to the Least Squares Problem by transforming the rectangular matrix A into triangular form.

4.1. **Cholesky Factorization.**

If A has full column rank, then the following holds for A^TA:

(1) A^TA is symmetric ($(A^TA)^T = A^T(A^T)^T = A^TA$)
(2) A^TA is positive definite ($x^TA^TAx = (Ax)^TAx = \|Ax\|_2^2 > 0$ if $x \neq 0$).

Thus, if an $m \times n$ matrix A has rank n, then A^TA is $n \times n$, symmetric, and positive definite. In this case it is favorable to use the Cholesky Factorization, which decomposes any positive definite symmetric matrix into two triangular matrices so that A^TA can be expressed as

$$A^TA = LL^T$$

where L is an $n \times n$ lower triangular matrix. See [2] for a detailed analysis.

4.2. **Flop: Complexity of Numerical Algorithms.**

Triangular matrices are used extensively in numerical algorithms such as the Cholesky or the QR Factorization because triangular systems are one of the simplest systems to solve. By deriving the flop count for triangular system solving, this section introduces flop counting as a method of evaluating the performance of an algorithm.

A flop is a floating point operation $(+, -, \times, /)$. In an $n \times n$ unit lower triangular system $Ly = b$, each y_k in y is obtained by writing

$$y_k = b_k - \sum_{j=1}^{k-1} l_{kj}y_j,$$

which requires $k-1$ multiplications and $k-1$ additions. Thus y requires $n^2 - n$ flops to compute. Since n is usually sufficiently large to ignore lower order terms, we say that an n-by-n forward substitution costs $\sim n^2$ flops.

When a linear system is solved by this algorithm, the arithmetic associated with solving triangular system is often dominated by the arithmetic required for the factorization. We only worry about the leading-order behaviors when counting flops; i.e., we assume m, n are large. Keeping this in mind, we will examine the Cholesky Factorization used for solving systems involving symmetric, positive definite matrices.

4.3. **Algorithm: Computing the Cholesky Factorization.** For a matrix A, define $A_{i:i',j:j'}$ to be the $(i'-i+1)\times(j'-j+1)$ submatrix of A with upper left corner a_{ij} and lower right corner $a_{i'j'}$.

Algorithm 4.1.
$R = A$
for $k = 1$ to m
  for $j = k+1$ to m
    $R_{j,j:m} = R_{j,j:m} - R_{k,j:m}R_{kj}/R_{kk}$
  $R_{k,k:m} = R_{k,k:m}/\sqrt{R_{kk}}$

The 4-th line dominates the operation count for this algorithm, so its flop count can be obtained by considering

$$\sum_{k=1}^{m}\sum_{j=k+1}^{m} 2(m-j) \sim 2\sum_{k=1}^{m}\sum_{j=1}^{k} j \sim \sum_{k=1}^{m} k^2 \sim m^3/3$$

Thus we have the following algorithm for the Normal Equations Method:

(1) Calculate $C = A^TA$ (C is symmetric, so $\approx mn^2$ flops)
(2) Cholesky Factorization $C = LL^T$ ($n^3/3$ flops)
(3) Calculate $d = A^Tb$ ($2mn$ flops)
(4) Solve $Lz = d$ by forward substitution (n^2 flops)
(5) Solve $L^Tx = z$ by back substitution (n^2 flops)

This gives us the cost for large $m, n : mn^2 + (1/3)n^3$ flops

4.4. **Shortcomings of Normal Equations.** The Normal Equations Method is much quicker than other algorithms but is in general more unstable.

Example 4.2. Consider the matrix

$$A = \begin{bmatrix} 1 & 1 \\ \epsilon & 0 \\ 0 & \epsilon \end{bmatrix},$$

where $0 < \epsilon \ll 1$. Then for very small ϵ,

$$A^TA = \begin{bmatrix} 1+\epsilon^2 & 1 \\ 1 & 1+\epsilon^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix},$$

which is singular.

Conditioning of the system is also worsened: $\kappa(A^TA) = [\kappa(A)]^2$, so the Normal Equations have a relative sensitivity that is squared compared to the original Least Squares Problem $Ax = b$. Thus we can conclude that any numerical method using the Normal Equations will be unstable, since the rounding errors will correspond to $(\kappa(A))^2$ instead of $\kappa(A)$.

5. Orthogonal Methods - The QR Factorization

The QR Factorization is an alternative approach that avoids the shortcomings of Normal Equations. For a matrix $A \in \mathbb{R}^{m\times n}$ with full rank n, let $A = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix}$. We wish to produce a sequence of orthonormal vectors $q_1, q_2, \ldots$ spanning the same space as the columns of A:

$$\langle q_1, \ldots, q_j \rangle = \langle a_1, \ldots, a_j \rangle \tag{5.1}$$

for $j = 1, 2, \dots, n$. This way, for $k = 1, \dots, n$, each a_k can be expressed as a linear combination of $q_1, \dots, q_k$. This is equivalent to the matrix form

$$A = QR$$

where $Q \in \mathbb{R}^{m \times n}$ has orthonormal columns, and $R \in \mathbb{R}^{n \times n}$ is upper triangular. To understand the factorization we will first introduce some special classes of matrices and their properties.

5.1. **Orthogonal Matrices.** A matrix $Q \in \mathbb{R}^{m \times n}$ with $m \geq n$, has orthonormal columns if all columns in Q are orthogonal to every other column and are normalized. When $m = n$ so that Q is a square matrix, we can define an orthogonal matrix:

Definition 5.2. A square matrix with orthonormal columns is referred to as an orthogonal matrix.

For an orthogonal matrix Q, it holds that $Q^T Q = QQ^T = I_n$. From this it is clear that $Q^{-1} = Q^T$ and if Q is an orthogonal matrix, then Q^T is also orthogonal. Using these properties, we will prove the following lemma which shows that multiplying a vector by an orthogonal matrix preserves its Euclidean norm:

Lemma 5.3. *Multiplying a vector with an orthogonal matrix does not change its 2-norm. Thus if Q is orthogonal it holds that*

$$\|Qx\|_2 = \|x\|_2.$$

Proof. $\|Qx\|_2^2 = (Qx)^T Qx = x^T Q^T Qx = x^T x = \|x\|_2^2$ □

Norm preservation implies no amplification of numerical error. The greatest advantage of using orthogonal transformations is in its numerical stability: if Q is orthogonal then $\kappa(Q) = 1$. It is also clear that multiplying both sides of Least Squares Problem (1.1) by an orthogonal matrix does not change its solution.

5.2. **Triangular Least Squares Problems.** The upper triangular overdetermined Least Squares Problem can be rewritten as

$$\begin{bmatrix} R \\ O \end{bmatrix} x = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix},$$

where R is an $n \times n$ upper triangular partition, the entries of O are all zero, and $b = \begin{bmatrix} b_1 & b_2 \end{bmatrix}^T$ is partitioned similarly. Then the residual becomes

$$\|r\|_2^2 = \|b_1 - Rx\|_2^2 + \|b_2\|_2^2.$$

Although $\|b_2\|_2^2$ does not depend on x, the first term $\|b_1 - Rx\|_2^2$ can be minimized when x satisfies the $n \times n$ triangular system

$$Rx = b_1,$$

which can be easily solved by back substitution. In this case, x is the Least Squares Solution with the minimum residual

$$\|r\|_2^2 = \|b_2\|_2^2.$$

Recall that solving Least Squares Problems by Normal Equations squares the condition number, i.e., $\kappa(A^T A) = [\kappa(A)]^2$. Combined with Lemma 5.3, we can conclude that the QR approach enhances numerical stability by avoiding this squaring effect.

5.3. **The QR Factorization in Least Squares Problems.**

Theorem 5.4. *Given the matrix $A \in \mathbb{R}^{m\times n}$ and the right hand side $b \in \mathbb{R}^m$, the solution set of the Least Squares Problem*

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

is identical to the solution set of

$$Rx = Q^T b$$

where the $m \times n$ matrix Q with orthonormal columns and the upper triangular $n \times n$ matrix R, is a QR-factorization of A.

Proof. Given $m \times n$ matrix A with $m > n$, we need to find an $m \times m$ orthogonal matrix Q such that

$$A = Q \begin{bmatrix} R \\ O \end{bmatrix},$$

so that when applied to the Least Squares Problem, $Ax = b$ becomes equivalent to

$$Q^T Ax = \begin{bmatrix} R \\ O \end{bmatrix} x = \begin{bmatrix} \hat{b}_1 \\ \hat{b}_2 \end{bmatrix} = Q^T b. \tag{5.5}$$

where R is $n \times n$ and upper triangular. If we partition Q as an $m \times m$ orthogonal matrix $Q = \begin{bmatrix} Q_1 & Q_2 \end{bmatrix}$, where Q_1 is $m \times n$, then

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

which is called the reduced QR Factorization of A. Then the solution to the Least Squares Problem $Ax = b$ is given by solution to square system

$$Q_1{}^T Ax = Rx = \hat{b}_1 = Q_1{}^T b$$

So the minimum value of $\|b - Ax\|_2^2$ is realized when $\|Q_1{}^T b - Rx\|_2^2 = 0$, i.e. when $Q^T b - Rx = 0$ □

Although we have shown that QR is an efficient, accurate way to solve Least Squares, exhibiting the QR Factorization is quite a different matter. A standard method of obtaining a QR Factorization is via Gram-Schmidt orthogonalization. Although this can be implemented numerically, it turns out not to be as stable as QR via Householder reflectors, and thus we choose to focus on the latter method. Finally, the Least Squares Problem can be solved by the following algorithm:

(1) QR Factorization of A (Householder or Gram-Schmidt $\sim 2mn^2$ flops)
(2) Form $d = Q^T b$ ($2mn$ flops)
(3) Solve $Rx = d$ by back substitution (n^2 flops)

This gives us the cost for large $m, n : 2mn^2$ flops

5.4. **Calculating the QR-factorization - Householder Transformations.** This section gives a rough sketch on how to calculate the QR-factorization in a way that is numerically stable. The main idea is to multiply the matrix A by a sequence of simple orthogonal matrices Q_k in order to "shape" A into an upper triangular matrix $Q_n \cdots Q_2 Q_1$. In the k-th step, the matrix Q_k introduces zeroes below the diagonal in the k-th column while keeping the zeroes in previous rows. By the end of the n-th step, all the entries below the diagonal become zero, making $Q^T = Q_n \cdots Q_2 Q_1 A = R$ upper triangular.

In order to construct the matrix Q^T, choose each orthogonal matrix Q_k such that

$$Q_k = \begin{bmatrix} I & 0 \\ 0 & F \end{bmatrix},$$

where I is a $k-1 \times k-1$ identity matrix and F is an $m-k+1 \times m-k+1$ matrix.The identity matrix in Q_k allows us to preserve the zeroes already introduced in the first $k-1$ columns, while F introduces zeroes below the diagonal in the k-th column. Now let $x \in \mathbb{R}^{m-k+1}$. We define F as the Household Reflector, which is chosen as

$$F = I - 2\frac{vv^T}{v^T v},$$

where $v = \|x\|e_1 - x$. This achieves

$$Fx = \begin{bmatrix} \|x\| \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \|x\|e_1.$$

After A has been reduced to upper triangular form $Q^T A = R$, the orthogonal matrix Q is constructed by directly computing the formula

$$Q^T = Q_n \cdots Q_1 \rightarrow Q = Q_1 \cdots Q_n.$$

As this process is repeated n times, each process works on smaller blocks of the matrix while deleting the entries below the diagonal in the next column. Each Householder transformation is applied to entire matrix, but does not affect prior columns, so zeros are preserved.

In order to achieve the operation count, we take advantage of the matrix-vector multiplications rather than the full use of matrix multiplication. In other words, we perform

$$(I - 2\frac{vv^T}{v^T v})A = A - 2\frac{v(v^T A)}{v^T v}$$

which consists of a matrix-vector multiplication and an outer product. Compared to the Cholesky Factorization, applying the Householder transformation this way is much cheaper because it requires only vector v, rather than the full matrix F. Repeating this process n times gives us the Householder algorithm:

Algorithm 5.6.
for $k = 1$ to n
  $v = A_{k:m,k}$
  $u_k = v - \|v\|e_1$
  $u_k = u_k/\|u_k\|$
  $A_{k:m,k:n} = A_{k:m,k:n} - 2u_k({u_k}^T A_{k:m,k:n})$

Most work is done by the last line in the loop: $A_{k:m,k:n} = A_{k:m,k:n} - 2u_k({u_k}^T A_{k:m,k:n})$, which costs $4(m-k+1)(n-k+1)$ flops. Thus we have

$$Total\ flops \sim \sum_{k=1}^{n} 4(m-k+1)(n-k+1) = 2mn^2 - 2n^3/3$$

5.5. **Rank Deficiency: Numerical Loss of Orthogonality.** We have assumed so far that the matrix A has full rank. But if A is rank-deficient, the columns A would be linearly dependent, so there would be some column a_j in A such that $a_j \in span\{q_1, \dots, q_{j-1}\} = span\{a_1, \dots, a_{j-1}\}$. In this case, from 5.1 we can see that the QR Factorization will fail. Not only that, when a matrix is close to rank-deficient, it could lose its orthogonality due to numerical loss. Consider a 2×2 matrix

$$A = \begin{bmatrix} 0.70000 & 0.70711 \\ 0.70001 & 0.70711 \end{bmatrix}$$

It is easy to check that A has full rank. But with a 5-digit accuracy of the problem, after computing the QR Factorization we see that

$$Q = \begin{bmatrix} 0.70710 & 0.70711 \\ 0.70711 & -0.70710 \end{bmatrix}$$

is clearly not orthogonal. If such a Q is used to solve the Least Squares Problem, the system would be highly sensitive to perturbations. Nevertheless, a minimum norm solution can still be computed with the help of Singular Value Decomposition (SVD), which will be covered in the next section.

6. Singular Value Decomposition (SVD)

If the QR Factorization used orthogonal transformations to reduce the Least Squares Problem to a triangular system, the Singular Value Decomposition uses orthogonal transformations to reduce the problem into a diagonal system. We first introduce the Singular Value Decomposition.

Theorem 6.1. *Let A be an arbitrary $m \times n$ matrix with $m \geq n$. Then A can be factorized as $A = U\Sigma V^T$, where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices, and $\Sigma = diag(\sigma_1, \dots, \sigma_n)$, where $\sigma_1 \geq \dots \geq \sigma_n \geq 0$.*

Proof. Let $\sigma_1 = \|A\|_2$. Choose $v_1 \in \mathbb{R}^n$ and ${u_1}' \in \mathbb{R}^m$ such that $\|v_1\|_2 = 1$, $\|{u_1}'\|_2 = \sigma_1$ and ${u_1}' = Av_1$. Then normalize ${u_1}'$ as $u_1 = {u_1}'/\|{u_1}'\|_2$. To form the orthogonal matrices U_1 and V_1, extend v_1 and u_1 to some orthonormal bases $\{v_i\}$ and $\{u_i\}$ as the orthogonal columns of each matrix. Then we have

$$ {U_1}^T A V_1 = S = \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix}. \tag{6.2}$$

Then it suffices to prove that $w = 0$. Consider the inequality

$$\left\| \begin{bmatrix} \sigma_1 & w^T \\ 0 & B \end{bmatrix} \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \right\|_2 \geq {\sigma_1}^2 + w^2 = ({\sigma_1}^2 + w^2)^{1/2} \| \begin{bmatrix} \sigma_1 \\ w \end{bmatrix} \|_2$$

which gives us $\|S\|_2 \geq ({\sigma_1}^2 + w^2)^{1/2}$. But since U_1 and V_1 are orthogonal, from (6.2) we have $\|S\|_2 = \sigma_1$, which makes $w = 0$.
Now proceed by induction. If $n = 1$ or $m = 1$ the case is trivial. In other cases, the submatrix B has an SVD $B = U_2\Sigma_2{V_2}^T$ by the induction hypothesis. Thus we obtain the SVD of A in the following form:

$$A = U_1 \begin{bmatrix} 1 & 0 \\ 0 & U_2 \end{bmatrix} \begin{bmatrix} \sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & V_2 \end{bmatrix}^T V_1^T$$

□

The main idea behind the SVD is that any matrix can be transformed into a diagonal matrix if we choose the right orthogonal coordinate systems for its domain and range. So using the orthogonality of V we can write the decomposition in the form

$$AV = U\Sigma.$$

6.1. **The Minimum Norm Solution using SVD.** If $rank(A) < n$, Theorem 2.4 tells us that the solution to the Least Squares Problem may not be unique, i.e., multiple vectors x give the minimum residual norm. Of course, rank deficiency should not happen in a well-formulated Least Squares Problem: the set of variables used to fit the data should be independent in the first place. In the case when A is rank-deficient, we can still compute the Least Squares Solution by selecting the solution $\hat{x}$ that has the smallest norm among the solutions.

We begin with the pseudoinverse A^+ from (3.3) redefined in terms of the SVD:

$$A^+ = V\Sigma^+U^T$$

where $\Sigma^+ = diag({\sigma_1}^{-1}, \dots, {\sigma_n}^{-1})$. Note that this pseudoinverse exists regardless of whether the matrix is square or has full rank. To see this, suppose $A = U\Sigma V^T$ is the SVD of $A \in \mathbb{R}^n$. For rank-deficient matrices with $rank(A) = r < n$, we have

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad \Sigma_1 = diag(\sigma_1, \dots, \sigma_r), \quad \sigma_1 \geq \cdots \geq \sigma_r > 0,$$

with U and V partitioned accordingly as $U = \begin{bmatrix} U_1 & U_2 \end{bmatrix}$ and $V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}$. The pseudoinverse A^+ is given by

$$A^+ = V\Sigma^+U^T = V_1{\Sigma_1}^{-1}{U_1}^T, \quad where \quad \Sigma^+ = \begin{bmatrix} {\Sigma_1}^{-1} & 0 \\ 0 & 0 \end{bmatrix}$$

Since Σ_1 is nonsingular, A^+ always exists, and it can be easily confirmed that $AA^+ = A^+A = I$. Now write c and y as

$$c = U^Tb = \begin{bmatrix} {U_1}^Tb \\ {U_2}^Tb \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, \quad y = V^Tx = \begin{bmatrix} {V_1}^Tx \\ {V_2}^Tx \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

Apply the orthogonality of U and the SVD of A to the Least Squares Problem:

$$\|b - Ax\|_2^2 = \|UU^Tb - U\Sigma V^Tx\|_2^2 = \left\| \begin{bmatrix} c_1 - \Sigma_1 y_1 \\ c_2 \end{bmatrix} \right\|_2^2 = \|c_1 - \Sigma_1 y_1\|_2^2 + \|c_2\|_2^2.$$

so that we have $\|b - Ax\|_2 \geq \|c_2\|_2 \; \forall x \in \mathbb{R}^n$. Equality holds when

$$x = Vy = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} \Sigma_1^{-1}c_1 \\ y_2 \end{bmatrix} = A^+b + V_2y_2$$

for any $y_2 \in \mathbb{R}^{n-r}$, making x the general solution of the Least Squares Problem. Note that the solution is unique when $r = rank(A) = n$, in which case $V_1 = V$. On the other hand, if A is rank-deficient ($r < n$), we write z as

$$z = Vy = \begin{bmatrix} V_1 & V_2 \end{bmatrix} \begin{bmatrix} \Sigma_1^{-1}c_1 \\ y_2 \end{bmatrix} = A^+b + V_2y_2$$

for some nonzero y_2. Let $\hat{x} = A^+b$, i.e., the solution when $y_2 = 0$. Then

$$\|z\|_2^2 = \|A^+b\|_2^2 + \|y_2\|_2^2 > \|x\|_2^2.$$

so that the vector $x = A^+b = V_1\Sigma_1^{-1}U_1^Tb$ is the minimum norm solution to the Least Squares Problem. Thus the minimum norm solution to the Least Squares Problem can be obtained through the following algorithm:

(1) Compute $A = U\Sigma V^T = U_1\Sigma_1 V_1^T$, the SVD of A
(2) Compute U^Tb
(3) Solve $\Sigma w = U^Tb$ for w (Diagonal System)
(4) Let $x = Vw$

The power of the SVD lies in the fact that it always exists and can be computed stably. The computed SVD will be well-conditioned because orthogonal matrices preserve the 2-norm. Any perturbation in A will not be amplified by the SVD since $\|\delta A\|_2 = \|\delta\Sigma\|_2$.

6.2. **Computing the SVD of Matrix A.** There are several ways to compute the SVD of the matrix A. First, the nonzero Singular Values of A can be computed by an eigenvalue computation for the normal matrix A^TA. However, as we have seen in the conditioning of Normal Equations, this approach is numerically unstable. Alternatively, we can transform A to bidiagonal form using Householder reflections, and then transform this matrix into diagonal form using two sequences of orthogonal matrices. See [3] for an in-depth analysis. While there are other versions of this method, the work required for computing the SVD is typically evaluated to be

$$\sim 2mn^2 + 11n^3.$$

[3] The computation of the SVD is an interesting subject in itself. Various alternative approaches are used in practice. Some methods emphasize speed (such as the divide-and-conquer methods), while others focus on the accuracy of small Singular Values (such as some QR implementations) [3]. But in any case, once it has been computed, the SVD can be a powerful tool in itself.

7. Comparison of Methods

So far, we have seen three algorithms for solving the Least Squares Problem:

(1) Normal equations by Cholesky Factorization costs $\sim mn^2 + n^3/3$ flops.
It is the fastest method but at the same time numerically unstable.
(2) QR Factorization costs $\sim 2mn^2 - 2n^3/3$ flops.
It is more accurate and broadly applicable, but may fail when A is nearly rank-deficient
(3) SVD costs $\sim 2mn^2 + 11n^3$ flops.
It is expensive to compute, but is numerically stable and can handle rank deficiency

If $m \approx n$, (1) and (2) require about same amount of work. If $m \gg n$, the QR method requires about twice as much work as Normal Equations. But the error for the Normal Equations Method produces solutions whose relative error is proportional to $[\kappa(A)]^2$. The Householder QR method is more accurate and more widely used than the Normal Equations, but these advantages may not be worth the additional cost when the problem is well conditioned.

The SVD is more stable and robust than the QR approach, but requires more computational work. If $m \gg n$, then the work for QR and SVD are both dominated by the first term, $2mn^2$, so the two methods cost about the same amount of work.

However, when $m \approx n$ the cost of the SVD is roughly 10 times that of the QR-factorization.

8. Acknowledgements

It is a pleasure to thank my mentor Jessica Lin for her guidance and expertise on numerical analysis. Her time and efforts in guiding me throughout my research has been most helpful. Finally, I owe great thanks to the University of Chicago REU program for funding my research.

References

1. Charles L. Lawson and Richard J. Hanson, *Solving least squares problems*, Prentice-Hall Inc., Englewood Cliffs, N.J., 1974, Prentice-Hall Series in Automatic Computation. MR 0366019 (51 #2270)
2. Endre Süli and David F. Mayers, *An introduction to numerical analysis*, Cambridge University Press, Cambridge, 2003. MR 2006500
3. Lloyd N. Trefethen and David Bau, III, *Numerical linear algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997. MR 1444820 (98k:65002)