

CS 450: Numerical Analysis¹

Linear Least Squares

University of Illinois at Urbana-Champaign

¹ *These slides have been drafted by Edgar Solomonik as lecture templates and supplementary material for the book “Scientific Computing: An Introductory Survey” by Michael T. Heath ([slides](#)).*

Linear Least Squares

- Find $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{Ax} - \mathbf{b}\|_2$ where $\mathbf{A} \in \mathbb{R}^{m \times n}$:

Since $m \geq n$, the minimizer generally does not attain a zero residual $\mathbf{Ax} - \mathbf{b}$. We can rewrite the optimization problem constraint via

$$\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^n} \|\mathbf{Ax} - \mathbf{b}\|_2^2 = \operatorname{argmin}_{\mathbf{x} \in \mathbb{R}^n} \left[(\mathbf{Ax} - \mathbf{b})^T (\mathbf{Ax} - \mathbf{b}) \right]$$

- Given the SVD $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ we have $\mathbf{x}^* = \underbrace{\mathbf{V}\mathbf{\Sigma}^\dagger\mathbf{U}^T}_{\mathbf{A}^\dagger} \mathbf{b}$, where $\mathbf{\Sigma}^\dagger$ contains the reciprocal of all nonzeros in $\mathbf{\Sigma}$:

- *The minimizer satisfies $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \mathbf{x}^* \cong \mathbf{b}$ and consequently also satisfies*

$$\mathbf{\Sigma} \mathbf{y}^* \cong \mathbf{d} \quad \text{where } \mathbf{y}^* = \mathbf{V}^T \mathbf{x}^* \text{ and } \mathbf{d} = \mathbf{U}^T \mathbf{b}.$$

- *The minimizer of the reduced problem is $\mathbf{y}^* = \mathbf{\Sigma}^\dagger \mathbf{d}$, so $y_i = d_i / \sigma_i$ for $i \in \{1, \dots, n\}$ and $y_i = 0$ for $i \in \{n+1, \dots, m\}$.*

Data Fitting via Linear Least Squares

Demo: Polynomial fitting with the normal equations

- ▶ Given a set of m points with coordinates x and y , seek an $n - 1$ degree polynomial p so that $p(x_i) \approx y_i$ by minimizing

$$\sum_{i=1}^m (y_i - p(x_i))^2 = \sum_{i=1}^m \left(y_i - \sum_{j=1}^n z_j x_i^{j-1} \right)^2$$

where $z \in \mathbb{R}^n$ are the unknown polynomial coefficients

- ▶ we can write this objective as a linear least squares problem

$$\|y - Az\|_2^2 \quad \text{where} \quad A = \begin{bmatrix} 1 & x_1 & \cdots & x_1^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & x_m & \cdots & x_m^{n-1} \end{bmatrix}$$

Conditioning of Linear Least Squares

- Consider a perturbation δb to the right-hand-side b

$$A(x + \delta x) \cong b + \delta b$$

- The amplification in relative perturbation magnitude (from b to x) depends on how much of b is spanned by the columns of A ,

$$(x + \delta x) = A^\dagger(b + \delta b)$$

$$\delta x = A^\dagger \delta b$$

$$\begin{aligned} \frac{\|\delta x\|_2}{\|x\|_2} &= \frac{\|A^\dagger \delta b\|_2}{\|x\|_2} \\ &\leq \frac{1}{\sigma_{\min}(A)} \frac{\|\delta b\|_2}{\|x\|_2} \\ &\leq \frac{1}{\sigma_{\min}(A)} \frac{\|\delta b\|_2}{\|Ax\|_2 / \|A\|_2} \\ &\leq \kappa(A) \frac{\|b\|_2}{\|Ax\|_2} \frac{\|\delta b\|_2}{\|b\|_2} \end{aligned}$$

Normal Equations

Demo: Normal equations vs Pseudoinverse

Demo: Issues with the normal equations

- *Normal equations* are given by solving $A^T A x = A^T b$:

If $A^T A x = A^T b$ then

$$(U \Sigma V^T)^T U \Sigma V^T x = (U \Sigma V^T)^T b$$

$$\Sigma^T \Sigma V^T x = \Sigma^T U^T b$$

$$V^T x = (\Sigma^T \Sigma)^{-1} \Sigma^T U^T b = \Sigma^\dagger U^T b$$

$$x = V \Sigma^\dagger U^T b = x^*$$

- However, solving the normal equations is a more ill-conditioned problem than the original least squares algorithm

Generally we have $\kappa(A^T A) = \kappa(A)^2$ (the singular values of $A^T A$ are the squares of those in A). Consequently, solving the least squares problem via the normal equations may be unstable because it involves solving a problem that has worse conditioning than the initial least squares problem.

Solving the Normal Equations

- ▶ If \mathbf{A} is full-rank, then $\mathbf{A}^T \mathbf{A}$ is symmetric positive definite (SPD):
 - ▶ *Symmetry is easy to check $(\mathbf{A}^T \mathbf{A})^T = \mathbf{A}^T \mathbf{A}$.*
 - ▶ *\mathbf{A} being full-rank implies $\sigma_{\min} > 0$ and further if $\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T$ we have*

$$\mathbf{A}^T \mathbf{A} = \mathbf{V}^T \mathbf{\Sigma}^2 \mathbf{V}$$

which implies that rows of \mathbf{V} are the eigenvectors of $\mathbf{A}^T \mathbf{A}$ with eigenvalues $\mathbf{\Sigma}^2$ since $\mathbf{A}^T \mathbf{A} \mathbf{V}^T = \mathbf{V}^T \mathbf{\Sigma}^2$.

- ▶ Since $\mathbf{A}^T \mathbf{A}$ is SPD we can use Cholesky factorization, to factorize it and solve linear systems:

$$\mathbf{A}^T \mathbf{A} = \mathbf{L} \mathbf{L}^T$$

QR Factorization

- ▶ If A is full-rank there exists an orthogonal matrix Q and a unique upper-triangular matrix R with a positive diagonal such that $A = QR$
 - ▶ Given $A^T A = LL^T$, we can take $R = L^T$ and obtain $Q = AL^{-T}$, since $\underbrace{L^{-1}A^T}_{Q^T} \underbrace{AL^{-T}}_Q = I$ implies that Q has orthonormal columns.
- ▶ A reduced QR factorization (unique part of general QR) is defined so that $Q \in \mathbb{R}^{m \times n}$ has orthonormal columns and R is square and upper-triangular. A full QR factorization gives $Q \in \mathbb{R}^{m \times m}$ and $R \in \mathbb{R}^{m \times n}$, but since R is upper triangular, the latter $m - n$ columns of Q are only constrained so as to keep Q orthogonal. The **reduced QR** factorization is given by taking the first n columns Q and \hat{Q} the upper-triangular block of R , \hat{R} giving $A = \hat{Q}\hat{R}$.
- ▶ We can solve the normal equations (and consequently the linear least squares problem) via reduced QR as follows

$$A^T A x = A^T b \quad \Rightarrow \quad \hat{R}^T \underbrace{\hat{Q}^T \hat{Q}}_I \hat{R} x = \hat{R}^T \hat{Q}^T b \quad \Rightarrow \quad \hat{R} x = \hat{Q}^T b$$

Gram-Schmidt Orthogonalization

Demo: Gram-Schmidt–The Movie
Demo: Gram-Schmidt and Modified Gram-Schmidt

► Classical Gram-Schmidt process for QR:

The Gram-Schmidt process orthogonalizes a rectangular matrix, i.e. it finds a set of orthonormal vectors with the same span as the columns of the given matrix. If \mathbf{a}_i is the i th column of the input matrix, the i th orthonormal vector (i th column of Q) is

$$\mathbf{q}_i = \mathbf{b}_i / \underbrace{\|\mathbf{b}_i\|_2}_{r_{ii}}, \quad \text{where} \quad \mathbf{b}_i = \mathbf{a}_i - \sum_{j=1}^{i-1} \underbrace{\langle \mathbf{q}_j, \mathbf{a}_i \rangle}_{r_{ji}} \mathbf{q}_j.$$

► Modified Gram-Schmidt process for QR:

Better numerical stability is achieved by orthogonalizing each vector with respect to each previous vector in sequence (modifying the vector prior to orthogonalizing to the next vector), so $\mathbf{b}_i = \text{MGS}(\mathbf{a}_i, i-1)$, where $\text{MGS}(\mathbf{d}, 0) = \mathbf{d}$ and

$$\text{MGS}(\mathbf{d}, j) = \text{MGS}(\mathbf{d} - \langle \mathbf{q}_j, \mathbf{d} \rangle \mathbf{q}_j, j-1)$$

Householder QR Factorization

- ▶ **A Householder transformation $Q = I - 2uu^T$ is an orthogonal matrix defined to annihilate entries of a given vector z , so $\|z\|_2 Qe_1 = z$:**
 - ▶ *Householder QR achieves unconditional stability, by applying only orthogonal transformations to reduce the matrix to upper-triangular form.*
 - ▶ *Householder transformations (reflectors) are orthogonal matrices, that reduce a vector to a multiple of the first elementary vector, $\alpha e_1 = Qz$.*
 - ▶ *Because multiplying a vector by an orthogonal matrix preserves its norm, we must have that $|\alpha| = \|z\|_2$.*
 - ▶ *As we will see, this transformation can be achieved by a rank-1 perturbation of identity of the form $Q = I - 2uu^T$ where u is a normalized vector.*
 - ▶ *Householder matrices are both symmetric and orthogonal implying that $Q = Q^T = Q^{-1}$.*
- ▶ **Imposing this form on Q leaves exactly two choices for u given z ,**

$$u = \frac{z \pm \|z\|_2 e_1}{\|z \pm \|z\|_2 e_1\|_2}$$

Applying Householder Transformations

- ▶ The product $x = Qw$ can be computed using $O(n)$ operations if Q is a Householder transformation

$$x = (I - 2uu^T)w = w - 2\langle u, w \rangle u$$

- ▶ Householder transformations are also called *reflectors* because their application reflects a vector along a hyperplane (changes sign of component of w that is parallel to u)
 - ▶ $I - uu^T$ would be an elementary projector, since $\langle u, w \rangle u$ gives component of w pointing in the direction of u and

$$x = (I - uu^T)w = w - \langle u, w \rangle u$$

subtracts it out.

- ▶ *On the other hand, Householder reflectors give*

$$y = (I - 2uu^T)w = w - 2\langle u, w \rangle u = x - \langle u, w \rangle u$$

which reverses the sign of that component, so that $\|y\|_2 = \|w\|_2$.

Givens Rotations

- ▶ Householder reflectors reflect vectors, Givens rotations rotate them
 - ▶ *Householder matrices reflect vectors across a hyperplane, by negating the sign of the vector component that is perpendicular to the hyperplane (parallel to u)*
 - ▶ *Any vector can be reflected to a multiple of an elementary vector by a single Householder rotation (in fact, there are two rotations, resulting in a different sign of the resulting vector)*
 - ▶ *Givens rotations instead rotate vectors by an axis of rotation that is perpendicular to a hyperplane spanned by two elementary vectors*
 - ▶ *Consequently, each Givens rotation can be used to zero-out (annihilate) one entry of a vector, by rotating it so that the component of the vector pointing in the direction of the axis corresponding to that entry, points into a different axis*
- ▶ Givens rotations are defined by orthogonal matrices of the form $\begin{bmatrix} c & s \\ -s & c \end{bmatrix}$
 - ▶ *Given a vector $\begin{bmatrix} a \\ b \end{bmatrix}$ we define c and s so that $\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \sqrt{a^2 + b^2} \\ 0 \end{bmatrix}$*
 - ▶ *Solving for c and s , we get $c = \frac{a}{\sqrt{a^2 + b^2}}$, $s = \frac{b}{\sqrt{a^2 + b^2}}$*

QR via Givens Rotations

- ▶ We can apply a Givens rotation to a pair of matrix rows, to eliminate the first nonzero entry of the second row

$$\begin{bmatrix} \mathbf{I} & & & \\ & c & & s \\ & & \mathbf{I} & \\ & -s & & c \\ & & & & \mathbf{I} \end{bmatrix} \begin{bmatrix} \vdots \\ a \\ \vdots \\ b \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \sqrt{a^2 + b^2} \\ \vdots \\ 0 \\ \vdots \end{bmatrix}$$

- ▶ Thus, $n(n-1)/2$ Givens rotations are needed for QR of a square matrix
 - ▶ Each rotation modifies two rows, which has cost $O(n)$
 - ▶ Overall, Givens rotations cost $2n^3$, while Householder QR has cost $(4/3)n^3$
 - ▶ Givens rotations provide a convenient way of thinking about QR for sparse matrices, since nonzeros can be successively annihilated, although they introduce the same amount of fill (new nonzeros) as Householder reflectors

Rank-Deficient Least Squares

- ▶ Suppose we want to solve a linear system or least squares problem with a (nearly) rank deficient matrix A
 - ▶ A rank-deficient (singular) matrix satisfies $Ax = 0$ for some $x \neq 0$
 - ▶ Rank-deficient matrices must have at least one zero singular value
 - ▶ Matrices are said to be deficient in **numerical rank** if they have extremely small singular values
 - ▶ The solution to both linear systems (if it exists) and least squares is not unique, since we can add to it any multiple of x
- ▶ Rank-deficient least squares problems seek a minimizer x of $\|Ax - b\|_2$ of minimal norm $\|x\|_2$
 - ▶ If A is a diagonal matrix (with some zero diagonal entries), the best we can do is $x_i = b_i/a_{ii}$ for all i such that $a_{ii} \neq 0$ and $x_i = 0$ otherwise
 - ▶ We can solve general rank-deficient systems and least squares problems via $x = A^\dagger b$ where the pseudoinverse is

$$A^\dagger = V \Sigma^\dagger U^T \quad \sigma_i^\dagger = \begin{cases} 1/\sigma_i & : \sigma_i > 0 \\ 0 & : \sigma_i = 0 \end{cases}$$

Truncated SVD

- ▶ After floating-point rounding, rank-deficient matrices typically regain full-rank but have nonzero singular values on the order of $\epsilon_{\text{mach}}\sigma_{\text{max}}$
 - ▶ *Very small singular values can cause large fluctuations in the solution*
 - ▶ *To ignore them, we can use a pseudoinverse based on the **truncated SVD** which retains singular values above an appropriate threshold*
 - ▶ *Alternatively, we can use Tychonov regularization, solving least squares problems of the form $\min_{\mathbf{x}} \|\mathbf{Ax} - \mathbf{b}\|_2^2 + \alpha \|\mathbf{x}\|^2$, which are equivalent to the augmented least squares problem*

$$\begin{bmatrix} \mathbf{A} \\ \sqrt{\alpha}\mathbf{I} \end{bmatrix} \mathbf{x} \cong \begin{bmatrix} \mathbf{b} \\ \mathbf{0} \end{bmatrix}$$

- ▶ By the **Eckart-Young-Mirsky theorem**, truncated SVD also provides the best low-rank approximation of a matrix (in 2-norm and Frobenius norm)
 - ▶ *The SVD provides a way to think of a matrix as a sum of outer-products $\sigma_i \mathbf{u}_i \mathbf{v}_i^T$ that are disjoint by orthogonality and the norm of which is σ_i*
 - ▶ *Keeping the r outer products with largest norm provides the best rank- r approximation*

QR with Column Pivoting

- ▶ QR with column pivoting provides a way to approximately solve rank-deficient least squares problems and compute the truncated SVD
 - ▶ *We seek a factorization of the form $QR = AP$ where P is a permutation matrix that permutes the columns of A*
 - ▶ *For $n \times n$ matrix A of rank r , the bottom $r \times r$ block of R will be 0*
 - ▶ *To solve least squares, we can solve the rank-deficient triangular system $Ry = Q^T b$ then compute $x = Py$*
- ▶ A pivoted QR factorization can be used to compute a rank- r approximation
 - ▶ *To compute QR with column pivoting,*
 - 1. pivot the column of largest norm to be the leading column,*
 - 2. form and apply a Householder reflector H so that $HA = \begin{bmatrix} \alpha & b \\ \mathbf{0} & B \end{bmatrix}$,*
 - 3. proceed recursively (go back to step 1) to pivot the next column and factorize B*
 - ▶ *Computing the SVD of the first r columns of AP^T generally (but not always) gives the truncated SVD*
 - ▶ *Halting after r steps leads to a cost of $O(n^2r)$*