Solving Least Squares
Normal Equations

- Let $\tilde{A}$ have full column rank, and be size $mxn$ with $m \geq n$
- Diagonal (nonzero) weighting $A = D\tilde{A}$ does not change the rank/size
  - but changes the answer when $D \neq I$ and $m \neq n$
- Minimizing $\|r\|_2 = \|b - Ac\|_2$ leads to the normal equations $A^TAc = A^Tb$ for the critical point
- Since $A^TA$ is SPD, $A^TAc = A^Tb$ has a unique solution obtainable via fast/efficient SPD solvers
- When $b$ is in the range of $A$, the unique solution to $A^TAc = A^Tb$ makes $r = 0$, and is thus the unique solution to $Ac = b$
  - When $A$ is square ($m = n$), and full rank, then $b$ is always in the range of $A$
Condition Number

- Compare $A = U\Sigma V^T$ and $A^T A = V\Sigma^T \Sigma V^T = V\Lambda V^T$ where $\Lambda = \Sigma^T \Sigma$ is a diagonal size $n \times n$ matrix of singular values squared

- Since the singular values of $A^T A$ are the square of those in $A$, the condition number $\frac{\sigma_{\text{max}}}{\sigma_{\text{min}}}$ of $A^T A$ is also squared (compared to $A$)
  - Thus, solving the normal equations requires twice the precision (e.g. $(10^7)^2 = 10^{14}$)

- It takes twice as much precision to get the same number of significant digits!

- The normal equations are not the preferred approach (unless $A$ is extremely well conditioned)

- However, (like the SVD) it is a great tool for theoretical purposes
  - Can transform any full rank matrix into an SPD system
Understanding Least Squares

- When $A = U\Sigma V^T$ has full column rank, $\Sigma = \begin{pmatrix} \hat{\Sigma} \\ 0 \end{pmatrix}$ with $\hat{\Sigma}$ a size $nxn$ diagonal matrix of (strictly) positive singular values
  - The 0 submatrix is size $(m - n)xn$ and doesn’t exist when $m = n$
- Note: $A^T A = V\Sigma^T \Sigma V^T = V\hat{\Sigma}^2 V^T$ and $(A^T A)^{-1} = V\hat{\Sigma}^{-2} V^T$
- So $c = (A^T A)^{-1} A^T b = V\hat{\Sigma}^{-2} V^T V\Sigma^T U^T b = V(\hat{\Sigma}^{-1} \begin{pmatrix} 0 \\ 0 \end{pmatrix}) U^T b$
- $Ac = U\Sigma V^T V(\hat{\Sigma}^{-1} \begin{pmatrix} 0 \\ 0 \end{pmatrix}) U^T b = U \begin{pmatrix} \hat{\Sigma} \\ 0 \end{pmatrix}(\hat{\Sigma}^{-1} \begin{pmatrix} 0 \\ 0 \end{pmatrix}) U^T b = U \begin{pmatrix} I_{nxn} \\ 0 \\ 0 \end{pmatrix} U^T b$
- $r = b - Ac = UI_{mxm} U^T b - U \begin{pmatrix} I_{nxn} \\ 0 \\ 0 \end{pmatrix} U^T b = U \begin{pmatrix} 0 \\ 0 \\ I_{(m-n)x(m-n)} \end{pmatrix} U^T b$
Understanding Least Squares

• Recall: from SVD slides (unit 3):
  • The columns of $U$ corresponding to “nonzero” singular values form an orthonormal basis for the range of $A$
  • The remaining columns of $U$ form an orthonormal basis for the (unattainable) space perpendicular to the range of $A$

• Since $A$ only has $n$ singular values, only the first $n$ columns of $U$ (which has $m$ columns) span the range of $A$

• Writing $\begin{pmatrix} \hat{b}_r \\ \hat{b}_z \end{pmatrix} = U^T b$ separates $\hat{b}_r$ (which is size $nx1$) in the range of $A$ from $\hat{b}_z$ (which is size $(m-n)x1$) orthogonal to the range of $A$

• Then (from the last slide): $c = V \hat{\Sigma}^{-1} \hat{b}_r$, $Ac = U \begin{pmatrix} \hat{b}_r \\ 0 \end{pmatrix}$, and $r = U \begin{pmatrix} 0 \\ \hat{b}_z \end{pmatrix}$
Orthogonal Matrices (and the L2 norm)

• Recall (from unit 3):
  • **Orthogonal matrices** have orthonormal columns (an orthonormal basis), so their transpose is their inverse. They preserve inner products, and thus are rotations, reflections, and combinations thereof.

• An orthogonal $\hat{Q}$ has $\hat{Q}\hat{Q}^T = \hat{Q}^T\hat{Q} = I$

• So, $\|\hat{Q}r\|_2 = \sqrt{\hat{Q}r \cdot \hat{Q}r} = \sqrt{r^T\hat{Q}^T\hat{Q}r} = \sqrt{r^Tr} = \|r\|_2$

• Similarly, $\|\hat{Q}^Tr\|_2 = \sqrt{\hat{Q}^Tr \cdot \hat{Q}^Tr} = \sqrt{r^T\hat{Q}\hat{Q}^Tr} = \sqrt{r^Tr} = \|r\|_2$

• That is, orthogonal transformations preserve Euclidean distance
Understanding Least Squares

- \( r = U \begin{pmatrix} 0 \\ \hat{b}_z \end{pmatrix} \) with orthogonal \( U \) implies \( \|r\|_2 = \|\hat{b}_z\|_2 \)

- Consider the diagonalized SVD view of \( Ac = b \) when \( A \) has full rank:
  \[
  U \Sigma \Sigma^T c = b \quad \text{or} \quad \begin{pmatrix} \Sigma \\ 0 \end{pmatrix} \hat{c} = \begin{pmatrix} \hat{b}_r \\ \hat{b}_z \end{pmatrix}
  \]

- The first block row gives \( c = V \hat{\Sigma}^{-1} \hat{b}_r \), which is the least squares solution

- The second block row is \( 0 = \hat{b}_z \). The norm of the residual for this block row is \( \|\hat{b}_z\|_2 \), which is identical to \( \|r\|_2 \)

- The SVD approach gives the same (minimum residual) least squares solution
Recall: Gram-Schmidt (Unit 5)

- Orthogonalizes a set of vectors
- For each new vector, subtract its (weighted) dot product overlap with all prior vectors, making it orthogonal to them
- A-orthogonal Gram-Schmidt uses an A-weighted dot/inner product
- Given vector $\tilde{S}^q$, subtract out the A-overlap with $s^1$ to $s^{q-1}$ so that the resulting vector $s^q$ has $<s^q, s^{\hat{q}}>_A = 0$ for $\hat{q} \in \{1, 2, \ldots, q - 1\}$
- That is, $s^q = \tilde{S}^q - \sum_{\hat{q}=1}^{q-1} \frac{<\tilde{s}^q, s^{\hat{q}}>_A}{<s^{\hat{q}}, s^{\hat{q}}>_A} s^{\hat{q}}$ where the two non-normalized $s^{\hat{q}}$ both require division by their norm (and $<s^{\hat{q}}, s^{\hat{q}}>_A = \|s^{\hat{q}}\|_A^2$)
- Proof: $<s^q, s^{\tilde{q}}>_A = <\tilde{S}^q, s^{\tilde{q}}>_A - \frac{<\tilde{s}^q, s^{\tilde{q}}>_A}{<s^{\tilde{q}}, s^{\tilde{q}}>_A} <s^{\tilde{q}}, s^{\tilde{q}}>_A = 0$
Gram-Schmidt for QR Factorization

- From $A$, create a full rank $Q$ with orthonormal columns
- For each column $a_k$, subtract the overlap with all prior columns in $Q$ and make the result unit length:
  \[
  q_k = \frac{a_k - \sum_{\hat{k}=1}^{k-1} \langle a_k, q_{\hat{k}} \rangle q_{\hat{k}}}{\|a_k - \sum_{\hat{k}=1}^{k-1} \langle a_k, q_{\hat{k}} \rangle q_{\hat{k}}\|_2}
  \]
- Define $r_{kk} = \langle a_k, q_{\hat{k}} \rangle$ for $k > \hat{k}$, and $r_{kk} = \|a_k - \sum_{\hat{k}=1}^{k-1} \langle a_k, q_{\hat{k}} \rangle q_{\hat{k}}\|_2$
- Then $q_k = \frac{a_k - \sum_{\hat{k}=1}^{k-1} r_{\hat{k}k} q_{\hat{k}}}{r_{kk}}$, and thus $a_k = r_{kk} q_k + \sum_{\hat{k}=1}^{k-1} r_{\hat{k}k} q_{\hat{k}} = \sum_{\hat{k}=1}^{k} r_{\hat{k}k} q_{\hat{k}}$
- This gives $A = QR$ where $R$ is upper triangular and $Q^T Q = I$
Gram-Schmidt QR (Example)

- Example: \( A = QR \) with upper triangular \( R \)

\[
\begin{pmatrix}
3 & -3 & 3 \\
2 & -1 & 1 \\
2 & -1 & -1 \\
2 & -3 & 3 \\
2 & -3 & 5
\end{pmatrix}
= \begin{pmatrix}
3/5 & 0 & 0 \\
2/5 & 1/2 & 1/2 \\
2/5 & 1/2 & -1/2 \\
2/5 & -1/2 & -1/2 \\
2/5 & -1/2 & 1/2
\end{pmatrix}
\begin{pmatrix}
5 & -5 & 5 \\
0 & 2 & -4 \\
0 & 0 & 2
\end{pmatrix}
\]

- Note that \( Q^T Q = I_{3\times3} \) since the columns of \( Q \) are orthonormal
- However, \( QQ^T \neq I_{5\times5} \) and \( Q \) is only a subset of an orthogonal matrix
Not Good for Large Matrices

• Gram-Schmidt has too much numerical drift for large matrices
• Don’t use Gram-Schmidt to find $A = QR$ with upper triangular $R$ and $Q^T Q = I$
• But it does provide a good conceptual way to think about $A = QR$
QR Factorization

- Consider $A = QR$ with upper triangular $R$ and $Q^TQ = I$
- $Q$ is size $m \times n$ (just like $A$) with $n$ orthonormal columns
- Let $\tilde{Q}$ be the matrix with $m - n$ orthonormal columns that span the space perpendicular to the range of $Q$
- Then, the size $m \times m$ matrix $\hat{Q} = (Q \quad \tilde{Q})$ is orthogonal
- $\|r\|_2 = \|\hat{Q}^Tr\|_2 = \left\| \begin{pmatrix} Q^T \\ \tilde{Q}^T \end{pmatrix} \begin{pmatrix} b - QRc \end{pmatrix} \right\|_2 = \left\| \begin{pmatrix} Q^Tb - Rc \\ \tilde{Q}^Tb \end{pmatrix} \right\|_2$
- Only the first (block) row varies with $c$, so $\|r\|_2$ is minimized by solving $Rc = Q^Tb$
- Since $R$ is upper triangular, $Rc = Q^Tb$ can be solved via back-substitution
Householder Transform

- Let the unit normal $\hat{v}$ implicitly define a plane orthogonal to it
- Then, $H = I - 2\hat{v}\hat{v}^T$ reflects vectors across that plane
  $$Ha = a - 2(\hat{v}^T a) \hat{v}$$
- $H$ is orthogonal with $H = H^T = H^{-1}$
- Don’t form the full $H$
- Instead, apply it via the definition of $\hat{v}$
Householder Transform

- Choose directions $v_k = a - Ha$ in a manner that zeroes out elements.

  $$
  \begin{pmatrix}
  a_1 \\
  \vdots \\
  a_{k-1} \\
  a_k \\
  a_{k+1} \\
  \vdots \\
  a_n
  \end{pmatrix}
  -
  \begin{pmatrix}
  a_1 \\
  \vdots \\
  a_{k-1} \\
  \gamma \\
  0 \\
  \vdots \\
  0
  \end{pmatrix}
  = \hat{a}_k - \gamma \hat{e}_k
  \text{ where } \hat{a}_k =
  \begin{pmatrix}
  0 \\
  \vdots \\
  0 \\
  a_k \\
  a_{k+1} \\
  \vdots \\
  a_n
  \end{pmatrix}
  $$

- E.g. $v_k = 
  \begin{pmatrix}
  a_1 \\
  \vdots \\
  a_{k-1} \\
  a_k \\
  a_{k+1} \\
  \vdots \\
  a_n
  \end{pmatrix}
  -
  \begin{pmatrix}
  a_1 \\
  \vdots \\
  a_{k-1} \\
  \gamma \\
  0 \\
  \vdots \\
  0
  \end{pmatrix}
  = \hat{a}_k - \gamma \hat{e}_k
  \text{ where } \hat{a}_k =
  \begin{pmatrix}
  0 \\
  \vdots \\
  0 \\
  a_k \\
  a_{k+1} \\
  \vdots \\
  a_n
  \end{pmatrix}$

- $Ha$ should be the same length as $a$ (i.e. a reflection), so $\gamma = \pm \| \hat{a}_k \|_2$

- Then, $\nu_k = \hat{a}_k \pm \| \hat{a}_k \|_2 \hat{e}_k$, which is subsequently normalized to $
  \hat{v}_k = \frac{\nu_k}{\| \nu_k \|_2}$

- For robustness, $\nu_k = \hat{a}_k + S(a_k)\| \hat{a}_k \|_2 \hat{e}_k$ where $S(a_k) = \pm 1$ is the sign function.
Householder Transform (Example)

- Consider $a = \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix}$ in the formulation $v_k = \hat{a}_k + S(a_k)\|\hat{a}_k\|_2 \hat{e}_k$

- Here $\hat{a}_1 = \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix}$, $v_1 = \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix} + S(2)\sqrt{9} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix}$, $\hat{v}_1 = \frac{1}{\sqrt{30}} \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix}$

- Then, $H_1 a = a - 2(\hat{v}_1^T a) \hat{v}_1 = \begin{pmatrix} 2 \\ 1 \\ 2 \end{pmatrix} - 2 \frac{15}{\sqrt{30}} \frac{1}{\sqrt{30}} \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} -3 \\ 0 \\ 0 \end{pmatrix}$
Householder Transform for QR

• For each column of $A$, construct the Householder transform that zeroes out entries below the diagonal

• Then $H_n H_{n-1} \cdots H_2 H_1 A = \begin{pmatrix} R \\ 0 \end{pmatrix}$ and $H_n H_{n-1} \cdots H_2 H_1 b = \begin{pmatrix} \tilde{b}_r \\ \tilde{b}_z \end{pmatrix}$

• Apply $H_k$ efficiently using $\hat{v}_k$, and remember to apply it to all columns $\geq k$
  - It doesn’t affect columns $< k$ (because of all the zeros at the top of $\hat{v}_k$)

• Note: $H_n$ is required to get zeroes at the bottom of the last column

• Letting $\hat{Q}^T = H_n H_{n-1} \cdots H_2 H_1$ gives $\hat{Q}^T A = \begin{pmatrix} R \\ 0 \end{pmatrix}$ or $A = \hat{Q} \begin{pmatrix} R \\ 0 \end{pmatrix}$

• $\|r\|_2 = \|\hat{Q}^T r\|_2 = \|\hat{Q}^T (b - \hat{Q} \begin{pmatrix} R \\ 0 \end{pmatrix} c)\|_2 = \|\begin{pmatrix} \tilde{b}_r \\ \tilde{b}_z \end{pmatrix} - \begin{pmatrix} Rc \\ 0 \end{pmatrix}\|_2$

• Solve $Rc = \tilde{b}_r$ via back-substitution to minimize $\|r\|_2$ to a value of $\|\tilde{b}_z\|_2$