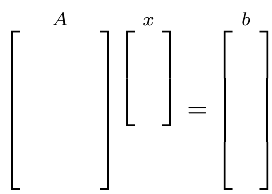
Linear Least Squares, Projection, Pseudoinverses

Cameron Musco

1 Over Determined Systems - Linear Regression

- A is a data matrix. Many samples (rows), few parameters (columns).
- b is like your y values - the values you want to predict. x is the linear coefficients in the regression.
- Overdetermined system. Can't exactly reconstruct b just from columns of A. But we want to find x that recombines columns of A to get as close to b as possible. Assume for now the best case - the columns of A are linearly independent.



The Solution - Pseudoinverse

- We can use the pseudoinverse: $A^+ = (A^\top A)^{-1} A^\top$. $x = A^+ b$.
- **The pseudoinverse takes vectors in the column space of A to vectors in the row space of A.** In this case, b might not actually be in the column space, so the pseudoinverse takes the projection of b onto the column space to a vector x in the row space.
- Technically, x might not be in the row space, if the matrix doesn't have full row rank. But remember: 'in the row space' means you are a linear combination of rows in A. You are not in the null space, the set of x such that $Ax = 0$. If in the null space, your dot product with every row is 0 so you are orthogonal to the row space. Any x can be written as a sum of its row space and null space components $x_R + x_N$. And $Ax = Ax_R + Ax_N = Ax_R$. So, if we choose the minimum length x, it will have no null space component, and will be in the row space of A.

There are multiple ways to arrive at the pseudoinverse:

Optimization Problem

$$\min_x \|b - Ax\|^2 = \min_x (b - Ax)^\top (b - Ax)$$
$$\min_x \|b - Ax\|^2 = \min_x b^\top b - 2x^\top A^\top b + x^\top A^\top Ax$$

Optimized when gradient is 0. Remember, just treat matrices as numbers:

$$\nabla_x = -2A^\top b + 2A^\top Ax$$

So need:

$$\begin{aligned} \nabla_x = 0 &= -2A^\top b + 2A^\top Ax \\ 2A^\top b &= 2A^\top Ax \\ \boldsymbol{A^\top b} &= \boldsymbol{A^\top Ax} \end{aligned}$$

This is called the normal equations. And is solved with

$$\boldsymbol{x} = \underbrace{\boldsymbol{(A^\top A)^{-1}A^\top}}_{\boldsymbol{A^+}} \boldsymbol{b}$$

Projection Onto Column Space

- Just project b onto the column space of A to get b_c and then solve $Ax = b_c$.
- Get b's similarity to each column vector by dotting it with each: $A^\top b$.
- Normalize weights by multiplying by $(A^\top A)^{-1}$. So weight vector becomes: $(A^\top A)^{-1}A^\top b$. Intuitively you can see how this works by thinking about the case when b is a column vector. $b = c_i$. Then we want our weights vector to be e_i. $(A^\top A)^{-1}A^T A = I$ so $(A^\top A)^{-1}A^T c_i$ is just the i^{th} column of I, aka e_i.
- Now actually use these weights to combine the columns of A to get b_c. $b_c = A * [(A^\top A)^{-1}A^\top b] = A(A^\top A)^{-1}A^\top b$
- Now we can actually solve:

$$\begin{aligned} Ax &= b_c \\ Ax &= A(A^\top A)^{-1}A^\top b \\ \boldsymbol{x} &= \underbrace{\boldsymbol{(A^\top A)^{-1}A^\top}}_{\boldsymbol{A^+}} \boldsymbol{b} \end{aligned}$$

Singular Value Decomposition

- Decompose $A = UDV^\top$

$$\overset{A}{\begin{bmatrix} \quad & \\ \\ \\ \\ \end{bmatrix}} = \overset{U}{\begin{bmatrix} \quad & \\ \\ \\ \\ \end{bmatrix}} \overset{D}{\begin{bmatrix} \quad & \\ \\ \end{bmatrix}} \overset{V}{\begin{bmatrix} \quad & \\ \\ \end{bmatrix}}$$

- This is the ‘truncated SVD. In general U is $m \times m$ (m is height of A), and D is $m \times n$. But since we have rank of A at most n, the $n + 1...m$ columns of u are just zero columns, and only the first n diagonal entries of D are nonzero.

$$\begin{aligned} Ax &= b \\ UDV^\top x &= b \\ (VD^{-1}U^\top)(UDV^\top)x &= (VD^{-1}U^\top)b \\ x &= (VD^{-1}U^\top)b \end{aligned}$$

since U and V are both orthogonal matrices.

- What exactly is $(VD^{-1}U^T)$? Well try the pseudoinverse:

$$
\begin{aligned}
(A^\top A)^{-1}A^\top &= (VDU^\top UDV^\top)^{-1}(VDU^\top) \\
(A^\top A)^{-1}A^\top &= (VD^{-2}V^\top)(VDU^\top) \\
\boldsymbol{(A^\top A)^{-1}A^\top} &= \boldsymbol{VD^{-1}U^\top}
\end{aligned}
$$

2 Under Determined Systems

- If A is short and fat or if A is tall but does not have full column rank. Then there are multiple x's such that $Ax = b$ or that achieve $\min \|b - Ax\|^2$.
- Then we want to solve for the x minimizing $\|x\|^2$. And yay. The pseudoinverse gives us exactly that x.

Optimization with Lagrange Multipliers

- Take the simplest case first and the analog to linear regression - A is short and fat but has full row rank. Use Lagrange multiplier to optimize.

$$
\min_{x \in \{x : Ax = b\}} \|x\|^2
$$
$$
\min_x \max_\lambda \|x\|^2 + \lambda^\top (b - Ax)
$$

Need to have $\nabla_x = 2x - A^\top \lambda = 0$ and $\nabla_\lambda = b - Ax = 0$ so $x = A^\top \lambda / 2$

$$
\begin{aligned}
0 &= b - AA^\top \lambda/2 \\
\lambda &= 2(AA^\top)^{-1}b
\end{aligned}
$$

so:

$$
\boldsymbol{x} = \underbrace{\boldsymbol{A^\top (AA^\top)^{-1}}}_{\boldsymbol{A}+} \boldsymbol{b}
$$

Projection Onto Row Space

- We want a solution x in the row space of A. So simply right x as a combination of row vectors: $x = A^\top w$.

$$
\begin{aligned}
Ax &= b \\
A(A^T w) &= b \\
w &= (AA^\top)^{-1}b \\
x &= A^\top w = \boldsymbol{A^\top (AA^\top)^{-1} b}
\end{aligned}
$$

3 Under *And* Over Determined Systems - The Golden Goose

- A has neither full row rank nor full column rank. There is no exact solution to $Ax = b$, but there are many optimal solutions.

- **Project b onto col(A)**: $A = UDV^\top$, where U provides an orthonormal basis for the column space of A. So just project b onto columns of U. $b_P = Uw$, where we can find the weights by dotting b with each of the columns of U. So $w = U^\top b$. Since U is orthonormal, no need to normalize weights - $(U^\top U)^{-1} = I$. (Remember, U is always tall and thin, or at best square, since we truncate it to only have $rank(A)$ columns. And since U has full column rank, $U^\top U = I$. So we now want to solve: $Ax = UU^\top b$.

- **Take x in rowspace of A:** To minimize the norm of x choose an x such that $x = A^\top c$.

- **Solve the new system using SVDs:** Remember we are using the truncated SVD. D has all positive diagonal entries so can be inverted, U is full column rank, and V is full row rank.

$$\begin{aligned}
A(A^\top c) &= UU^\top b \\
UDV^\top VDU^\top c &= UU^\top b \\
UD^2U^\top c &= UU^\top b \\
c &= (UD^{-2}U^\top)UU^\top b \\
c &= UD^{-2}Ub \\
x &= (VDU^\top)UD^{-2}Ub \\
\boldsymbol{x} &= \underbrace{\boldsymbol{VD^{-1}U}}_{\boldsymbol{A^+}}\boldsymbol{b}
\end{aligned}$$