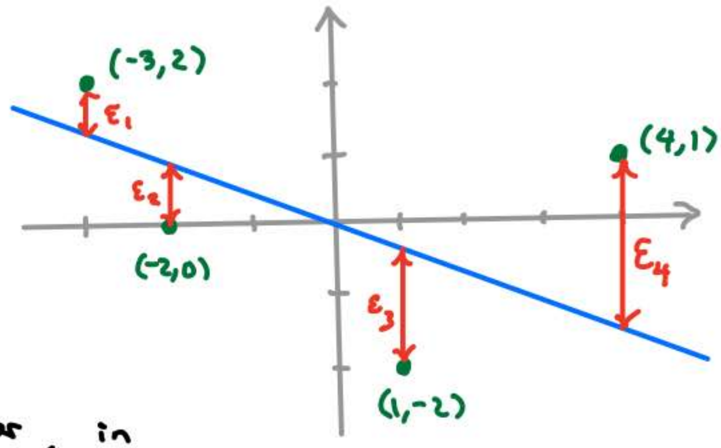
Lecture 33 : Least Squares

Data fitting 101

Say you're in charge of redeveloping a public park, and you want to run a straight footpath as close as possible to 4 giant oaks.



That is, we want to find β_0 & β_1 coming "as close as possible" to solving

$$\left.\begin{aligned} \beta_0 + \beta_1(-3) &= 2 \\ \beta_0 + \beta_1(-2) &= 0 \\ \beta_0 + \beta_1(1) &= -2 \\ \beta_0 + \beta_1(4) &= 1 \end{aligned}\right\}$$

or, in matrix form

$$\begin{array}{cc} \vec{w}_0 & \vec{w}_1 \end{array}$$
$$\begin{pmatrix} 1 & -3 \\ 1 & -2 \\ 1 & 1 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ -2 \\ 1 \end{pmatrix}$$

$$X \cdot \vec{\beta} = \vec{y}.$$

[Of course, it would be more appropriate to take perpendicular distances, but for our first example let's not overcomplicate this.]

As you can check, $\vec{y}$ does not belong to $\mathrm{Col}(X)$, and so this system can't be <u>solved</u>. Instead, we aim to

→ <u>minimize the sum of squares of the vertical errors ε_i.</u>

That is, we'd like to choose $\vec{\beta}$ to make

vector in Col(X)

$$\sqrt{\sum_{i=1}^{4}\{(\beta_0+\beta_1 x_i)-y_i\}^2} = \|X\vec{\beta}-\vec{y}\| = \mathrm{dist}(X\vec{\beta}, \vec{y})$$

as small as possible. In fact, the vector in Col(X) that does this is exactly

$$\hat{y} = \mathrm{Proj}_{\mathrm{Col}(X)}(\vec{y})$$

(valid b/c $\vec{w}_0 \perp \vec{w}_1$) $\rightarrow$

$$= \frac{\vec{w}_0\cdot\vec{y}}{\vec{w}_0\cdot\vec{w}_0}\vec{w}_0 + \frac{\vec{w}_1\cdot\vec{y}}{\vec{w}_1\cdot\vec{w}_1}\vec{w}_1$$

$$= \frac{1}{4}\begin{pmatrix}1\\1\\1\\1\end{pmatrix} + \frac{-4}{30}\begin{pmatrix}-3\\-2\\1\\4\end{pmatrix} = \frac{1}{60}\begin{pmatrix}39\\31\\7\\-17\end{pmatrix}.$$

In other words, the best-fit line will pass through $(-3, \frac{39}{60})$, $(-2, \frac{31}{60})$, $(1, 7/60)$, $(4, -17/60)$. To find its equation, we need to solve

$$X\vec{\beta} = \hat{y}$$

for $\vec{\beta}$ — i.e., find β_0 & β_1 such that $\beta_0\vec{w}_0 + \beta_1\vec{w}_1 = \hat{y}$.

But looking at the calculation, we see that we've <u>already done this</u>, and $\beta_0 = \frac{1}{4}$ & $\beta_1 = \frac{-2}{15}$! So we use the line

$$\boxed{y = \frac{1}{4} - \frac{2}{15}x.}$$

Now, how do things change if the trees are at (0,2), (1,0), (2,-2), and (3,1) instead? Then $\vec{y}$ is the same, but our matrix X becomes

$$X = \begin{array}{cc} \vec{w}_0 & \vec{w}_1 \end{array} \\ X = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix},$$

and $\vec{w}_0, \vec{w}_1$ are <u>no longer orthogonal</u>. Perhaps a QR-decomposition could come in handy?

<u>Step 1</u>: <u>Run the Gram-Schmidt algorithm</u> (on columns of X)

$$\vec{v}_0 = \vec{w}_0 = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\vec{v}_1 = \vec{w}_1 - \frac{\vec{v}_0 \cdot \vec{w}_1}{\vec{v}_0 \cdot \vec{v}_0}\vec{v}_0 = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \end{pmatrix} - \frac{6}{4}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -3/2 \\ -1/2 \\ 1/2 \\ 3/2 \end{pmatrix}$$

$$\Rightarrow \vec{u}_0 = \frac{\vec{v}_0}{\|\vec{v}_0\|} = \begin{pmatrix} 1/2 \\ 1/2 \\ 1/2 \\ 1/2 \end{pmatrix}, \quad \vec{u}_1 = \begin{pmatrix} -3/2\sqrt{5} \\ -1/2\sqrt{5} \\ 1/2\sqrt{5} \\ 3/2\sqrt{5} \end{pmatrix} \quad \text{and } Q = \begin{pmatrix} \uparrow & \uparrow \\ \vec{u}_0 & \vec{u}_1 \\ \downarrow & \downarrow \end{pmatrix}.$$

As always with Gram-Schmidt, $Col(Q) = Col(X)$.

<u>Step 2</u>: <u>Find the decomposition $X = QR$</u>

Since Q has o.n. columns,

$$Q^T Q = I_2.$$

From $QR = X$ we get

$$R = \underbrace{Q^T Q}_{I_2} R = Q^T X$$

$$\Rightarrow \qquad R = \begin{pmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ -3/2\sqrt{5} & -1/2\sqrt{5} & 1/2\sqrt{5} & 3/2\sqrt{5} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix} = \begin{pmatrix} 2 & 3 \\ 0 & \sqrt{5} \end{pmatrix}$$

Step 3: <u>Use this to solve the least-squares problem.</u>

Columns of Q provide an o.n. basis of $\mathrm{Col}(X)$.

To project $\vec{y}$ to this, we have the formula from Lec. 32:

$$\hat{y} = \mathrm{Proj}_{\mathrm{Col}(X)} \vec{y} = Q Q^T \vec{y}.$$

So we must solve

$$X \vec{\beta} = \hat{y}$$

$$Q R \vec{\beta} = Q Q^T \vec{y}$$

$$\underbrace{\cancel{Q^T Q}}_{I_2} R \vec{\beta} = \underbrace{\cancel{Q^T Q}}_{I_2} Q^T \vec{y}$$

we must be careful here about "cancelling" Q's, b/c Q is 4×2.

(*) $\boxed{R \vec{\beta} = Q^T \vec{y}.}$

Now

$$Q^T \vec{y} = \begin{pmatrix} 1/2 & 1/2 & 1/2 & 1/2 \\ -3/2\sqrt{5} & -1/2\sqrt{5} & 1/2\sqrt{5} & 3/2\sqrt{5} \end{pmatrix} \begin{pmatrix} 2 \\ 0 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -\sqrt{5}/2 \end{pmatrix}.$$

So (*) reads

$$\begin{pmatrix} 2 & 3 \\ 0 & \sqrt{5} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} = \begin{pmatrix} 1/2 \\ -\sqrt{5}/2 \end{pmatrix} \quad \Rightarrow \quad \beta_1 = -\tfrac{1}{2}$$

$$\Rightarrow \quad \beta_0 = 1$$

and the least-squares line is $y = 1 - \frac{1}{2} x$.

$\boxed{\text{Summary}}$ We found

(1) <u>How to get QR factorization of A</u> $\leftarrow$ $m \times n$, $m \geq n$

<u>Step 1</u>: Perform full Gram-Schmidt on columns of A $\rightsquigarrow Q$.
i.e. normalize the vectors at the end!

<u>Step 2</u>: Compute $R = Q^T A$ (should be upper-triangular)

(2) <u>A least-squares solution of an equation $A\vec{x} = \vec{b}$ is $\hat{x} \in \mathbb{R}^n$ such that $\|\vec{b} - A\hat{x}\| \leq \|\vec{b} - A\vec{x}\|$ for all $\vec{x} \in \mathbb{R}^n$.</u>

<u>It can be obtained by solving $A\hat{x} = \hat{b}$, where $\hat{b} = \mathrm{Proj}_{\mathrm{Col}(A)}(\vec{b})$,</u>

since

- $\mathrm{Col}(A)$ is the set of vectors that can be written $A\vec{x}$
- $\mathrm{Proj}_{\mathrm{Col}(A)}(\vec{b})$ is the closest vector to $\vec{b}$ in $\mathrm{Col}(A)$

(in particular, $A\hat{x} = \hat{b}$ is consistent).

(3) <u>If the columns of A are independent and you know a QR decomposition of A, then a least-squares solution may be found by</u>

$$\underline{R\hat{x} = Q^T\vec{b}}$$

since then

$$\underbrace{QR}_{A}\hat{x} = \underbrace{QQ^T\vec{b}}_{\hat{b}}.$$

Q1: What can you say about the least-squares solution of $A\vec{x}=\vec{b}$ when $\vec{b}$ is $\perp$ to A's columns?

Ans: It's $\vec{0}$! (More precisely, $\vec{0}$ is <u>a</u> least-squares solution.)

Q2: When is "least-squares solution" <u>unique</u>, i.e. <u>the</u> least-squares solution?

Ans: When columns of A are independent, so that $\mathrm{Nul}(A)=\{\vec{0}\}$.

Finding $\hat{x}$ w/o Gram-Schmidt or QR

Set $\hat{b} := \mathrm{Proj}_{\mathrm{Col}(A)}\vec{b}$. Then

$\hat{x}$ is a least-squares solution to $A\vec{x}=\vec{b}$ $\iff$ $A\hat{x}=\hat{b}$

$\iff A\hat{x}-\vec{b} \in (\mathrm{Col}(A))^{\perp}$

write $A=\begin{pmatrix} \uparrow & & \uparrow \\ \vec{c}_1 & \cdots & \vec{c}_n \\ \downarrow & & \downarrow \end{pmatrix}$ $\iff 0 = \vec{c}_j \cdot (A\hat{x}-\vec{b}) = c_j^T(A\hat{x}-\vec{b})$ for each j

$\iff A^T(A\hat{x}-\vec{b}) = \vec{0}$

$\iff \boxed{A^TA\hat{x} = A^T\vec{b}}$ "Normal equations"

to be discussed next time

Claim: A^TA is invertible $\iff$ columns of A are independent.

In this case, $\hat{x} = (A^TA)^{-1}A^T\vec{b}$

is the <u>unique</u> least-squares solution.

Ex/ Find a (or the!) least-squares solution to

$$\begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix} \vec{x} = \begin{pmatrix} 2 \\ 0 \\ -2 \\ 1 \end{pmatrix} \qquad [A\vec{x} = \vec{b}]$$

in this way.

$$A^T A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 1 & 2 \\ 1 & 3 \end{pmatrix} = \begin{pmatrix} 4 & 6 \\ 6 & 14 \end{pmatrix}$$

$$A^T \vec{b} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ 0 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Normal equations: $\begin{pmatrix} 4 & 6 \\ 6 & 14 \end{pmatrix} \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$

$$\begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix} = \frac{1}{20} \begin{pmatrix} 14 & -6 \\ -6 & 4 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ -\frac{1}{2} \end{pmatrix}$$

which is the same solution as before, but from a rather different-looking 2×2 system!

//

While the absence of square-roots in the normal equation makes it appear better suited for computers, the opposite is true: taking the inverse of $A^T A$ turns out to be (numerically) a bit of a disaster for large matrices.