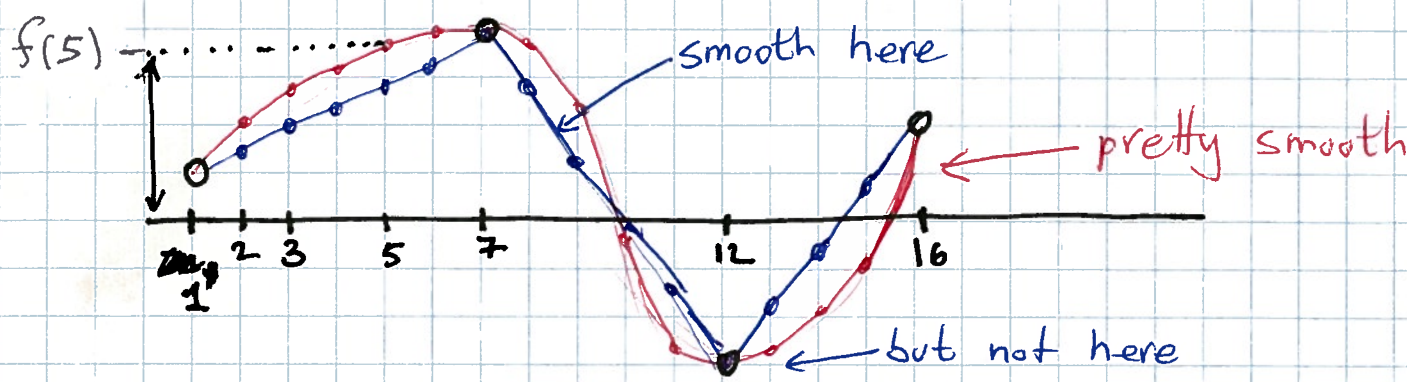
6.006 Lecture 24: Linear Equations & Least Squares

- Creating smooth meshes
- Soft and hard ~~constraint~~ constraints
- Linear least squares
- Matrix representation & upper triangular form

CLRS Ch. 28 covers this lecture & thursday's, but using a more complex and less accurate algorithm.

Reconstructing Smooth Functions

We want to find the smoothest piecewise linear function that satisfies $f(1)=1$, $f(7)=4$, $f(12)=-3$, $f(16)=2$ (or any constraints of this type)



We will model smooth as "$f(k)$ is close to the average of $f(k-1)$ and $f(k+1)$".

$$f(2) \approx \frac{f(1)+f(3)}{2} \quad \rightarrow \quad -f(1) + 2f(2) - f(3) \approx 0$$

$$f(3) \approx \frac{f(2)+f(4)}{2} \quad \rightarrow \quad -f(2) + 2f(3) - f(4) \approx 0$$

$$\vdots$$

we know $f(7)$ but want smoothness $\rightarrow$

$$f(7) \approx \frac{f(6)+f(8)}{2}$$

$$\vdots$$

"soft" smoothness constraints
We have more constraints than unknowns.

$$\left.\begin{array}{l} f(1)=1 \\ f(7)=4 \\ f(12)=-3 \\ f(16)=2 \end{array}\right\} \text{"hard" equality constraints}$$

Types of systems of linear equations

$$\left.\begin{array}{l}\left.\begin{aligned} x_1 + x_2 &= 2 \\ x_2 &= 1 \end{aligned}\right\} \text{consistent \& determines all unknowns} \\ \begin{aligned} x_1 \quad &= 7 \end{aligned}\end{array}\right\} \text{overdetermined \& inconsistent}$$

$$\left.\begin{array}{l}\left.\begin{aligned} x_1 + x_2 + x_3 &= 3 \\ x_2 + x_3 &= 7 \end{aligned}\right\} \text{underdetermined} \\ \begin{aligned} x_1 + 2x_2 + 2x_3 &= 10 \end{aligned}\end{array}\right\} \text{still underdetermined!}$$

overdetermined systems are usefull: just write down all the equations you know to be true, or even just hope to make true.

- We substitute the known $f(k)$'s to eliminate them:

$$-f(1) + 2f(2) - f(3) \approx 0 \qquad f(1) = 1$$
$$-1 + 2f(2) - f(3) \approx 0$$
$$2f(2) - f(3) \approx 1$$

- We can't satisfy all the constraints (too many) but we can try to minimize the discrepancy

$$r_1 = 2f(2) - f(3) \qquad -1$$
$$r_2 = -f(2) + 2f(3) - f(4) - 0$$
$$\vdots$$

- We bring the problem to a standard form by labeling the unkowns $x_1 \cdots x_n$, the constants $b_1 \cdots b_m$ ($m = \#$ constraints) and the coefficients $a_{11} \cdots a_{1n}, a_{21} \cdots a_{2n}, \cdots a_{mn}$.

$x_1 \equiv f(2) \cdots \quad x_5 = f(6), x_6 = f(8) \cdots \quad x_{12} = f(15)$

$b_1 = 1,\ b_2 = 0,\ b_3 = 0, \ldots$

$a_{11} = 2 \quad a_{12} = -1 \quad a_{1,3 \cdots n} = 0$

$a_{21} = -1 \quad a_{22} = 2 \quad a_{2,3} = -1 \quad a_{2,4 \cdots n} = 0$

- Next, we'll see how to minimize the r_i's for any a_{ij}'s, and b_i's.

Linear Least Squares Problems

- We need to minimize

$$r_1 = a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n - b_1$$
$$r_2 = a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n - b_2$$
$$\vdots \qquad\qquad\qquad\qquad \vdots$$
$$r_n = a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n - b_n$$
$$\vdots \qquad\qquad\qquad\qquad \vdots$$
$$r_m = a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n - b_m$$

by assigning values to the x_j's (unkowns).

- Matrix notation

$$\underbrace{\begin{bmatrix} r_1 \\ \vdots \\ r_n \\ \vdots \\ r_m \end{bmatrix}}_{r} = \underbrace{\begin{bmatrix} \sum_j a_{1j}x_j \\ \vdots \\ \sum_j a_{nj}x_j \\ \vdots \\ \sum_j a_{mj}x_j \end{bmatrix}}_{Ax} - \underbrace{\begin{bmatrix} b_1 \\ \vdots \\ b_n \\ \vdots \\ b_m \end{bmatrix}}_{b} = \underbrace{\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & & & \\ a_{n1} & \cdots & \cdots & a_{nn} \\ \vdots & & & \\ a_{m1} & \cdots & \cdots & a_{mn} \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}}_{x} - \underbrace{\begin{bmatrix} b_1 \\ \vdots \\ b_n \\ \vdots \\ b_m \end{bmatrix}}_{b}$$

or simply $Ax - b$

- How to measure how small is a vector $r = Ax - b$ with many components? Possible answers:
  - $\max_i |r_i|$
  - $\sum_i r_i^2$ ← easy to minimize & sensible with respect to a few large r_i's.

    $\boxed{\text{least squares}}$
  - $\sum_i |r_i|$

- One solution strategy is to find a solution assuming that A has a specific pattern of zeros and non-zeros, and then to find an algorithm to transform any A to this pattern (There is really only one other strategy; not in 6.006).

Consider the following structure:

$$
\begin{array}{llllll}
a_{11}x_1 + & a_{12}x_2 + & a_{13}x_3 + \cdots + & a_{1n}x_n & - & b_1 \\
0\,x_1 + & a_{22}x_2 + & a_{23}x_3 + \cdots + & a_{2n}x_n & - & b_2 \\
0\,x_1 + & 0\,x_2 + & a_{33}x_3 + \cdots + & a_{3n}x_n & - & b_3 \\
\vdots & & & & & \vdots \\
0\,x_1 + & 0\,x_2 + & 0\,x_3 + \cdots + & a_{nn}x_n & - & b_n \\
0\,x_1 + & 0\,x_2 + & 0\,x_3 + \cdots + & 0\,x_n & - & b_{n+1} \\
\vdots & & & & & \vdots \\
0\,x_1 + & 0\,x_2 + & 0\,x_3 + \cdots + & 0\,x_n & - & b_m
\end{array}
$$

- Clearly, $r_{n+1} = b_{n+1} \cdots r_m = b_m$ no matter how we set the x_j's.
- We can usually make all the other r_i's zero.

Set $x_n = \frac{b_n}{a_{nn}}$ so $r_n = 0$ (we ignore for now the case $a_{nn} = 0$).

Now substitute x_n in the other expressions

$$x_{n-1} = \frac{b_{n-1} - a_{n-1,n}\,x_n}{a_{n-1,n-1}} \leftarrow \text{now known}$$