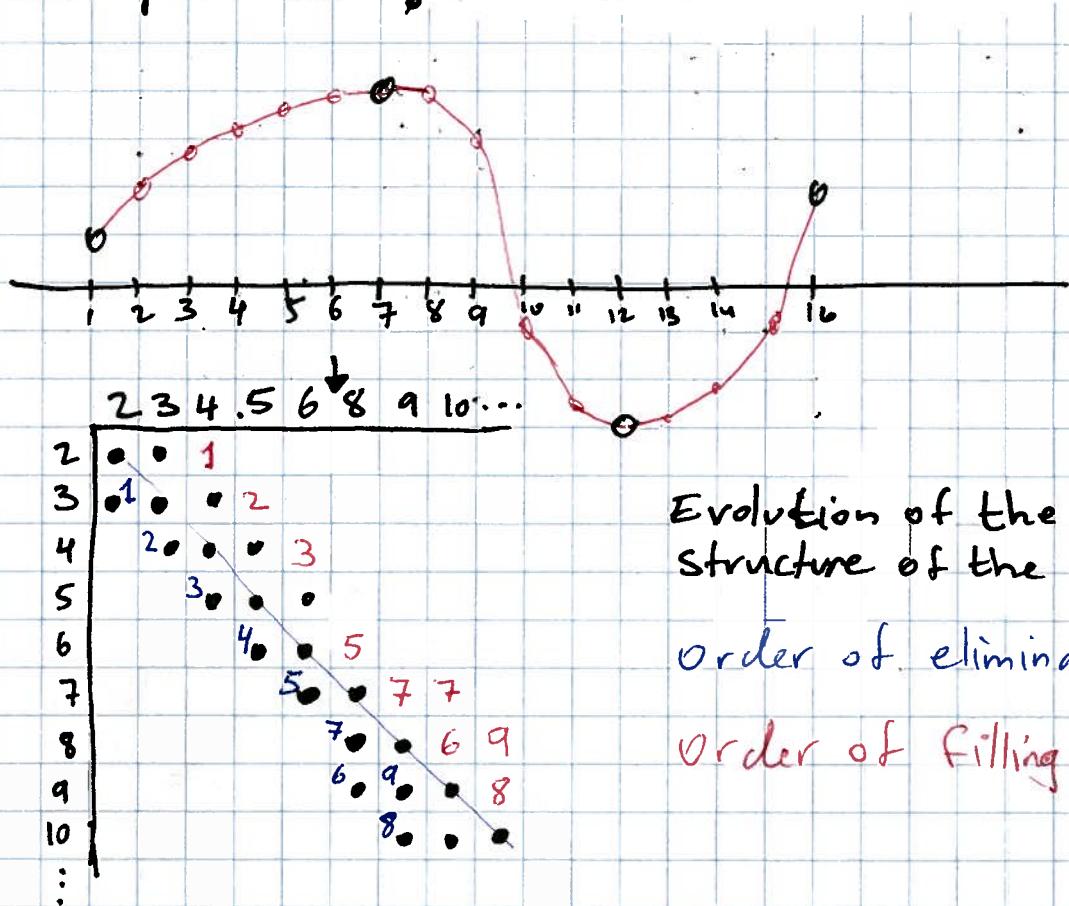


## 6.006 Lecture 25: Least squares III, numerics

- sparse least squares,
- Numerics

## Solving Sparse Least Squares Problems

In the smooth-function reconstruction problem,  $A$  has plenty of zeros to begin with; we can exploit that.



Evolution of the nonzero structure of the matrix

Order of elimination

Order of filling in

For this problem, if we only constrain the values of  $f(x)$  at a constant # points, total cost is  $O(N)$ .

There are clever elimination-ordering algs for other sparsity patterns, for general sparse matrices, and for other factorizations.

## Accuracy & Stability in numerical algorithms

The issue: try in Python (or most other languages) the following code:

$$x = 1.0$$

$$y = 1e-17 \quad \text{mathematically wrong}$$

$$x + y - x \Rightarrow 0.0 \quad \left. \begin{array}{l} \text{floating point arithmetic} \\ \text{is not commutative} \end{array} \right\}$$

$$x - x + y \Rightarrow 1e-17 \quad \left. \begin{array}{l} \text{floating point arithmetic} \\ \text{is not commutative} \end{array} \right\}$$

Floating point arithmetic: approximation of real arithmetic. Fixed precision + exponent

$2^{1000}$  has an exact representation

$1 + \underbrace{0.00\dots 0}_\text{16 zeros} 1$  does not; rounded to 1

How does this affect us?

we get approx answers, e.g., in shortest paths.

Solutions:

- Use a set closed under operations

e.g.  $+, -, \times$ ; integers for shortest paths  
(make sure  $\rightarrow$  to avoid overflows)

Hard to use a closed set if we divide  
(or take  $\sqrt{\cdot}$ ); rational rep gets really  
large

- Accept approximations, make sure  
they make sense quantitatively.

$$1.000\cdots 01 \rightarrow 1.0 \text{ accurate}$$

$$1 + 1e-17 - 1 = 1e-17 \rightarrow 0.0 \text{ not accurate but stable}$$

One definition of stability: correct answer  
to a small perturbation of input

$$1 + 1e-17 - 1.00\cdots 01 = 0.0$$

perturbation, small relative to 1

Intuition about stable algorithms:

- Information is lost mostly in subtractions/ additions
- When the terms are part of the input, that's fine: answer is correct to a perturbed input
- NOT FINE when alg generates huge or tiny terms.

Back to ~~Givens~~ Givens rotations

$$S^2 + C^2 = 1$$

$$C = S \frac{a_{m-1,1}}{a_{m,1}} \quad \text{or} \quad S = C \frac{a_{m,1}}{a_{m-1,1}}$$

$$\text{Substitute } C \text{ or } S \Rightarrow x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

a bit more analysis required to understand accuracy of answer.  
rigorous treatment in 6.336

Which root to take? make  $-b$  and  $\pm\sqrt{b^2 - 4ac}$  have same sign; otherwise we may generate a tiny # which will not be accurate.

$\Rightarrow$  This observation is enough to make our alg state-of-the-art stable.

Arithmetic vs. Verbal

If computation is not quantitative, don't round.  
e.g., cryptographic hashing

|               |                              |                              |
|---------------|------------------------------|------------------------------|
| text          | $f(\text{text})$             | $h(f(\text{text}))$          |
| "my password" | $\longrightarrow 5673097531$ | $\longrightarrow 3756199312$ |

$\uparrow$   
crypto hash fn:  
can't go  $\leftarrow$

website only stores hash, not the password

"my password"  $\rightarrow \dots \rightarrow 9375612363$

$\uparrow$   
completely  
different

Must do arithmetic in it exactly.