Lecture 2: Solving linear systems of equations

Thursday, August 27, 2015 9:30 AM

Reading: Chapter 2 of Meyer

Outline:
- Linear equations
- Geometry
- Solution stability
- Gaussian elimination
- Experiments in Matlab/Mathematica
  - timing
  - sparse matrices
  - 1D and 2D lattices
- LU decomposition
- Homogeneous and non-homogeneous systems

Linear equations

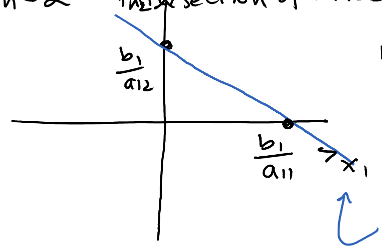
Recall the problem:

$$\underbrace{A}_{m\times n \text{ matrix}} \underbrace{x}_{n \text{ unknowns}} = \underbrace{b}_{m\text{-dim vector}} \iff \sum_{j=1}^{n} a_{ij} x_j = b_i \quad \text{for } i = 1, \dots, n$$

Geometric picture

$n=1$ one variable x_1 (trivial)

$n=2$ intersection of lines in $\mathbb{R}^2$



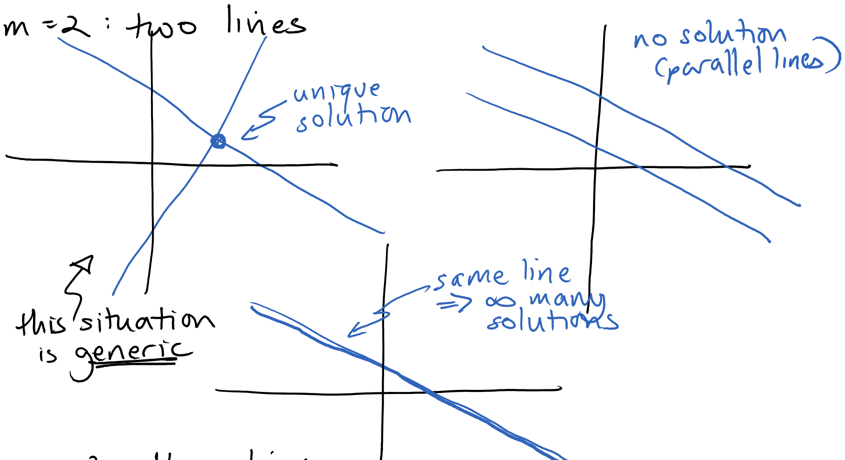
$m=1$:

$a_{11}x_1 + a_{12}x_2 = b_1$

equation of a line

all solutions (infinitely many)

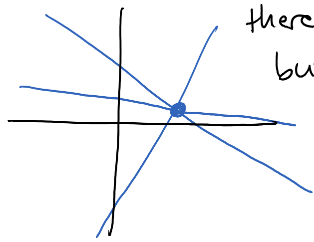
$m=2$: two lines

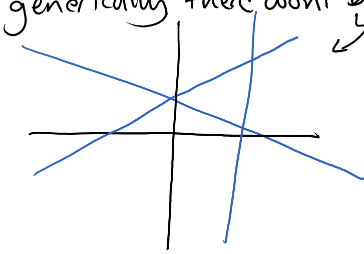


$m=3$: three lines

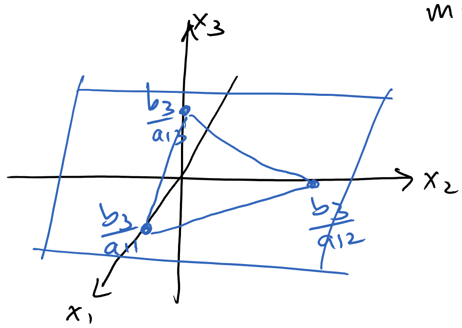
$m=3$: three lines

there could be a solution, but generically there won't be





$n=3$ variables:



$\underline{m=1 \text{ equation}}$

$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_3$

determines a 2D plane!

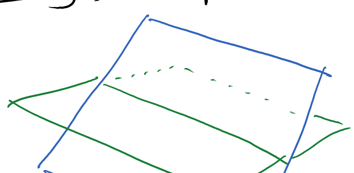
special case: $a_{11} = 0$

$\Rightarrow x_1$ doesn't matter

$\Rightarrow$ plane is parallel to x_1 axis

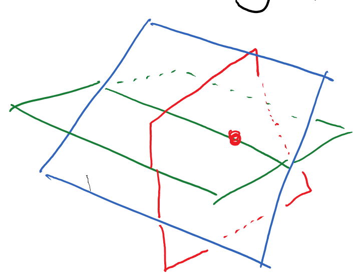
$\underline{m=2 \text{ equations}}$:

generically, two planes intersect in a line



but they could also be parallel, intersecting nowhere (or everywhere if they are the same)

$\underline{m=3 \text{ equations}}$: a plane and a line generically intersect in a single point



n variables $x_1, x_2, \ldots, x_n$

equation $\longleftrightarrow$ $(n-1)$-dimensional hyperplane

eg., $\{(x_1, \ldots, x_n) : x_n = 0\}$

generically, two intersect in an $(n-2)$-dim hyperplane, and so on ...

n should intersect in a single point

(I'm not proving this, but just relying on your intuition, and from analogy to the $n=2,3$ cases)

We'll say much more about this geometry later...

We'll say much more about this geometry later...

(Geometrically, though, what does Gaussian elimination correspond to?
– This is more easily seen when we introduce linear transformations: adding and multiplying rows leaves the range unchanged.)

<u>It is important!</u>

① Stability — very important

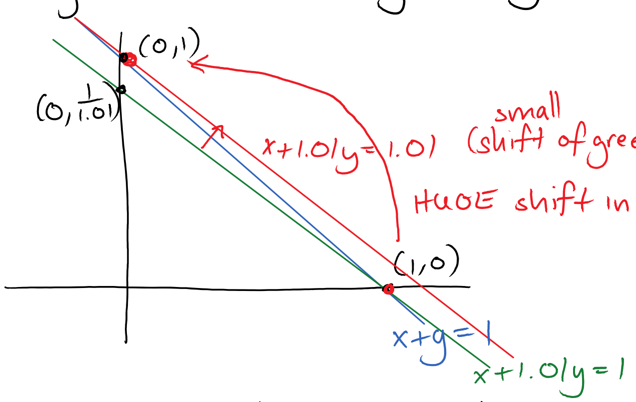
Example: Consider the equations

$$\begin{cases} x + y = 1 \\ x + 1.01y = 1 \end{cases} \Rightarrow (x,y) = (1,0)$$

$$\begin{cases} x + y = 1 \\ x + 1.01y = 1.01 \end{cases} \Rightarrow (x,y) = (0,1)$$

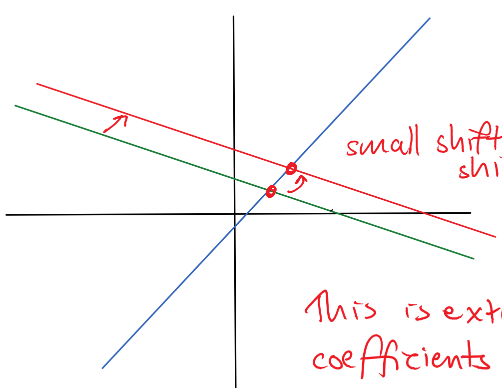
A small change to the equations led to a HUGE change in the solutions!

Why? Observe the geometry:



– because the lines are almost parallel

this wouldn't have happened if the lines were at a greater ∡



This is extremely important when coefficients are given numerically.

<u>Note</u>: In <u>higher dimensions</u>, you have the same problem with planes that are nearly parallel

Note: In higher dimensions, you have the same problem with planes that are nearly parallel. 
But they don't have to be nearly parallel; it is more complicated than that. (Eg., think of above green and blue lines as intersections of planes in 3D with $\{z=0\}$ plane. They needn't be nearly parallel.)

We'll study this more later.

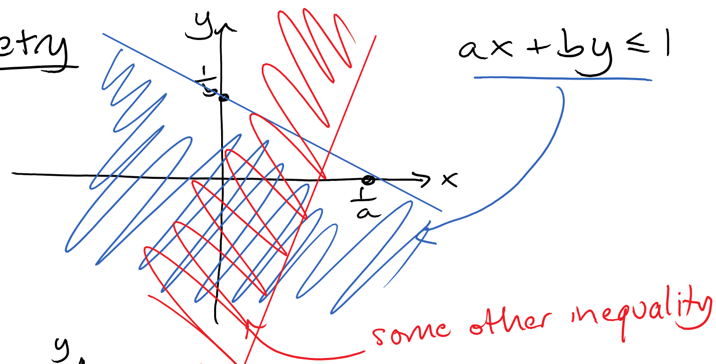
② Linear programming

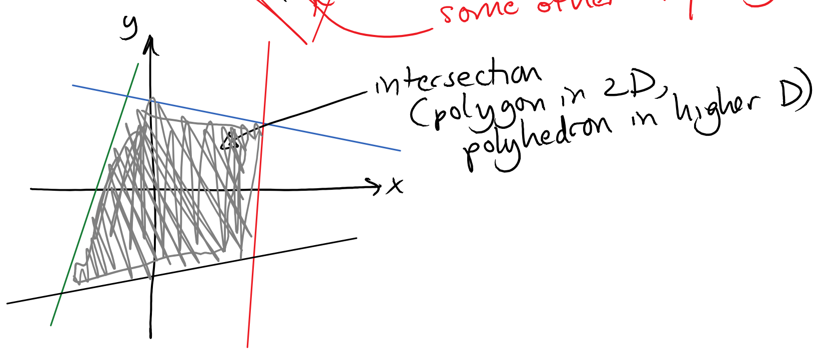
Problem: Solve a system of linear inequalities.

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2$$
$$\vdots \qquad\qquad \vdots$$

* Very useful in practice — inequalities often capture resource constraints
  Eg., say you have a \$100 budget for rice & beans
  \$3/lb \$5/lb
  $$3r + 5b \le 100$$
  $$\vdots$$
* Generalize linear systems of equations
  $$\sum_j a_{ij}x_j \le b_i \;\; \& \;\; \sum_j a_{ij}x_j \ge b_i \iff \sum_j a_{ij}x_j = b_i$$

Geometry





How to Solve Systems of Linear Equations

- Gaussian elimination
  - LU decomposition
- Inverting the matrix A (if it is invertible)
- Iterative methods
  - preconditioners

- conjugate gradient (for a positive semi-definite matrix)
- multigrid methods ...

• Methods for solving SDD systems

⋮

Recall:

Gaussian elimination:

Observe: 1. Multiplying an equation by a nonzero number doesn't change the solution set.

2. With two equations, adding one to the other doesn't change the solution set.

(x solves E_1, E_2 $\iff$ x solves E_1, E_1+E_2)

Therefore: Add multiples of one equation to the others to eliminate one variable from them.

Then repeat. Solve for the last variable, back substitute. That's it!

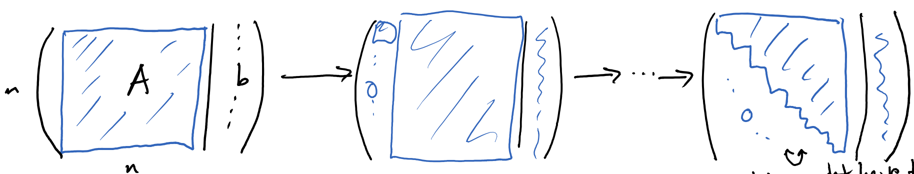
$O(n^3)$ complexity: n variables

for an $n\times n$ system: add $O(n)$ pairs of equations to eliminate a variable

$O(n)$ coefficients to update in an equation.

$\therefore$ doubling n $\Rightarrow$ $4\times$ the memory (to store A)

$8\times$ the running time!



Note: might have to rearrange columns (variables) for this form

Example: this works with any number of equations m

$$\begin{cases} x_1 + 2x_2 + 2x_3 + 3x_4 = 0 \\ 2x_1 + 4x_2 + x_3 + 3x_4 = 0 \\ 3x_1 + 6x_2 + x_3 + 4x_4 = 0 \end{cases}$$

Note: $(0,0,0,0)$ is one solution — are there more?

$$\left(\begin{array}{cccc|c} 1 & 2 & 2 & 3 & 0 \\ 2 & 4 & 1 & 3 & 0 \\ 3 & 6 & 1 & 4 & 0 \end{array}\right) \xrightarrow{-2,\ -3} \left(\begin{array}{cccc|c} 1 & 2 & 2 & 3 & 0 \\ 0 & 0 & -3 & -3 & 0 \\ 0 & 0 & -5 & -5 & 0 \end{array}\right) \xrightarrow{+\frac{5}{3}}$$

(Note: To minimize accumulation of floating point errors, it is often best to pivot on the largest magnitude entry in the column (3))

$$\left(\begin{array}{cccc|c} 1 & 2 & 2 & 3 & 0 \\ 0 & 0 & -3 & -3 & 0 \\ \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} & \cancel{0} \end{array}\right) \xrightarrow{+1} \quad \leftarrow 0=0$$

$$\left(\begin{array}{cccc|c} 1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -3 & -3 & 0 \end{array}\right)$$

$$\hookrightarrow \left(\begin{array}{cccc|c} 1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -3 & -3 & 0 \end{array}\right)$$

$x_1 + 2x_2 - x_3 = x_3 + x_4 = 0$

$\Rightarrow x_3 = -x_4$

$\quad x_1 = -2x_2 - x_4$

$\Rightarrow$ solutions $= \{(x_1, x_2, x_3, x_4) \in \mathbb{R}^4 : \begin{array}{l} x_1 = -2x_2 - x_4 \\ x_3 = -x_4 \end{array}\}$

x_2 & x_4 are free — can be arbitrary

— the solution set is a 2D plane in $\mathbb{R}^4$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -2x_2 - x_4 \\ x_2 \\ -x_4 \\ x_4 \end{pmatrix} = x_2 \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + x_4 \begin{pmatrix} -1 \\ 0 \\ -1 \\ 1 \end{pmatrix}$$

everything expressed in the free variables

eg.

(x_2, x_4)	(x_1, x_2, x_3, x_4)
$(0,0)$	$(0,0,0,0)$
$(1,0)$	$(-2,1,0,0)$
$(0,1)$	$(-1,0,-1,1)$
$(1,1)$	$(-3,1,-1,1)$

⋮

Stepping back...

We started with a "homogeneous" system of equations (right-hand sides all 0)

$\Rightarrow \vec{0} = (0,0,0,0)$ is a solution

$\Rightarrow$ any multiple of a solution is a solution

$\vec{x} = (x_1, x_2, x_3, x_4)$ solution $\Rightarrow \alpha\vec{x} = (\alpha x_1, \alpha x_2, \alpha x_3, \alpha x_4)$ is, too

b/c $\displaystyle\sum_{j=1}^{n} a_{ij}(\alpha x_j) = \alpha \cdot \sum_{j=1}^{n} a_{ij} x_j = \alpha \cdot 0 = 0$ ✓

$\Rightarrow$ the sum of two solutions is a solution

$\vec{x}, \vec{x}'$ solutions $\Rightarrow$ so is $\vec{x} + \vec{x}'$

b/c $\displaystyle\sum_{j=1}^{n} a_{ij}(x_j + x_j') = \left(\sum_j a_{ij} x_j\right) + \left(\sum_j a_{ij} x_j'\right) = 0 + 0 = 0$ ✓

— this is the formal definition of a subspace, in this case a plane parameterized by x_2 and x_4

Definition: A system of equations $Ax = b$ is
- homogeneous if $b = (0, 0, \ldots, 0)$
- non-homogeneous if $b \neq 0$

Gaussian elimination works just as well for non-homogeneous systems...

Example:

$x_1 + 2x_2 + 2x_3 + 3x_4 = 1$

$2x_1 + 4x_2 + x_3 + 3x_4 = -1$

$3x_1 + 6x_2 + x_3 + 4x_4 = \cancel{0}\ -2$

$$\underbrace{\qquad\qquad\qquad\qquad}_{\text{same coefficients } A \text{ as before}} \qquad \underbrace{\qquad}_{\text{different } b \neq \vec{0}}$$

$$\left(\begin{array}{cccc|c} 1 & 2 & 2 & 3 & 1 \\ 2 & 4 & 1 & 3 & -1 \\ 3 & 6 & 1 & 4 & \cancel{0}\ -2 \end{array}\right) \begin{array}{l} \\ -2 \\ -3 \end{array} \longrightarrow \left(\begin{array}{cccc|c} 1 & 2 & 2 & 3 & 1 \\ 0 & 0 & -3 & -3 & -3 \\ 0 & 0 & -5 & -5 & \cancel{-3}\ -5 \end{array}\right)$$

contradiction $\Rightarrow$ no solution!

$x_1 + 2x_2 - x_3 = -2$

$x_3 + x_4 = 1$

$\Rightarrow$ again, x_2 and x_4 are free

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} -2-2x_2+(1-x_4) \\ x_2 \\ 1-x_4 \\ x_4 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \underbrace{x_2 \begin{pmatrix} -2 \\ 1 \\ 0 \\ 0 \end{pmatrix} + x_4 \begin{pmatrix} -1 \\ 0 \\ -1 \\ 1 \end{pmatrix}}_{\text{this part is the same as for the homogeneous system}}$$

(*) $\Rightarrow$ The solution set for the nonhomogeneous equations is just a translation of the subspace (2D plane) of solutions to the corresponding homogeneous equations!

This makes sense: If $Ax = b$ n-h sol.

$Ay = 0$ h sol.

$\Rightarrow A(x+y) = b + 0 = b$ n-h sol.

Example: Solving differential equations numerically

Thursday, August 27, 2015 9:30 AM

(§1.4)

Problem:

Find a function f on $[0,1]$ with

$$f''(t) = \sqrt{t}$$

$$f(0) = f(1) = 0$$

Answer: $f''(t) = \sqrt{t}$

$$\Rightarrow f'(t) = \tfrac{2}{3} t^{3/2} + c$$

$$\Rightarrow f(t) = \tfrac{4}{15} t^{5/2} + ct + d$$

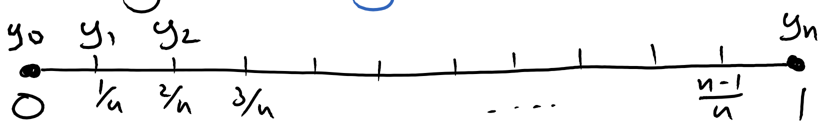
for some constants c, d

$$f(0) = 0 \Rightarrow d = 0$$

$$f(1) = \tfrac{4}{15} + c = 0 \Rightarrow c = -\tfrac{4}{15}$$

$$\therefore \boxed{f(t) = \tfrac{4}{15}\left(t^{5/2} - t\right)}$$

Now let's try solving this numerically.
by discretizing the interval $[0,1]$



Goal: Solve for f only at these discrete points

variables $y_j = f(j/n)$ for $j = 1, \ldots, n-1$

$$f'(t) = \frac{d}{dt} f(t) \approx \frac{f(t + 1/n) - f(t)}{1/n}$$

$f''(t)$ is the change in the derivative

$$\approx \frac{f'(t) - f'(t - 1/n)}{1/n} \approx \frac{1}{(1/n)^2}\Big(\big(f(t+1/n) - f(t)\big) - \big(f(t) - f(t - 1/n)\big)\Big)$$

$$= n^2 \left(f(t + \tfrac{1}{n}) - 2f(t) + f(t - 1/n)\right)$$

equations

$$y_0 = 0$$

$$y_n = 0$$

$$y_{j-1} - 2y_j + y_{j+1} = \frac{1}{n^2}\sqrt{\frac{j}{n}}$$

Solving in Mathematica:

```
n = 50;
```

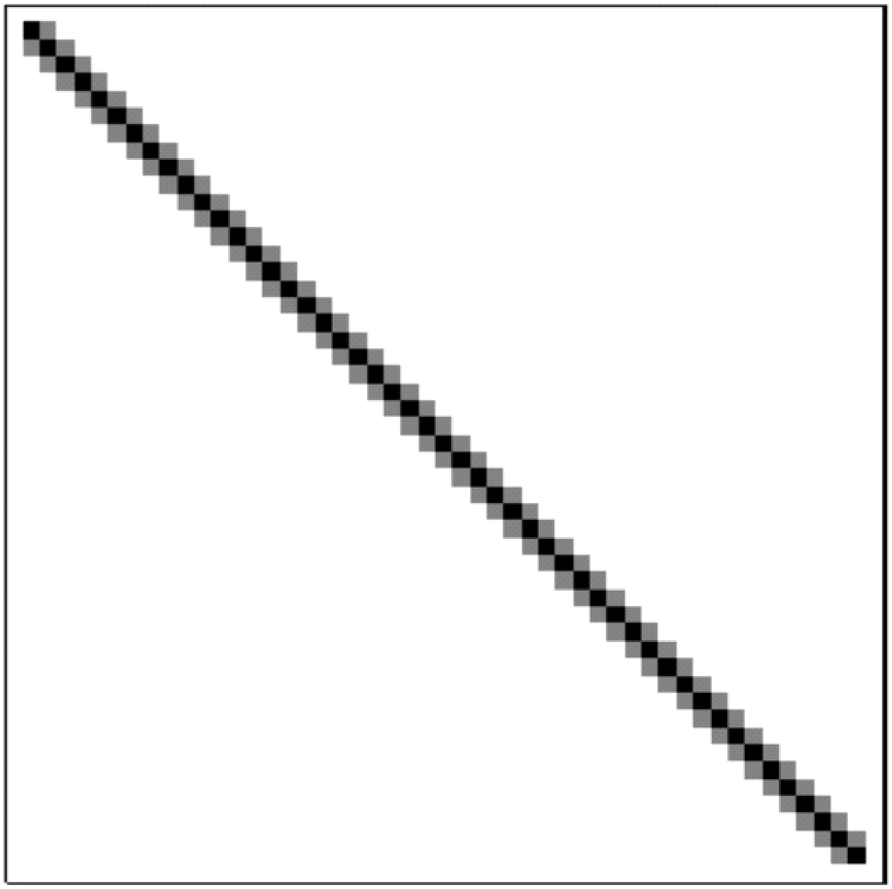
Sparse Array

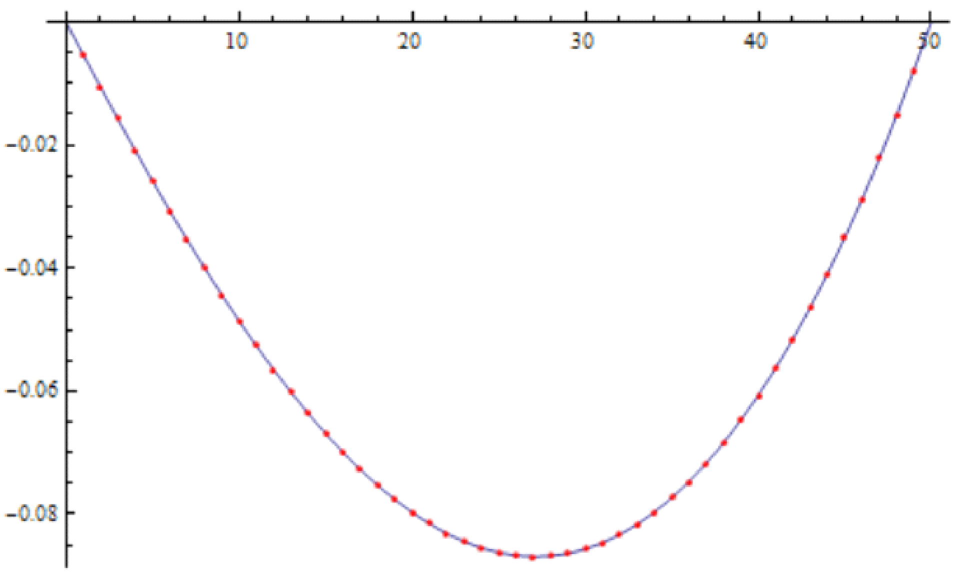
```
n = 50;
A = SparseArray[Join[
    Table[{j, j} → -2, {j, 1, n - 1}],
    Table[{j, j - 1} → 1, {j, 2, n - 1}],
    Table[{j, j + 1} → 1, {j, 1, n - 2}]
  ]];
b = Table[1/n^2 Sqrt[j/n], {j, 1, n - 1}] // N;
Dimensions[A]
Dimensions[b]

ArrayPlot[A]

plot1 = LinearSolve[A, b] // ListPlot[#, PlotStyle → RGBColor[1, 0, 0]] &;
plot2 = Plot[4/15 ((t/n)^(5/2) - t/n), {t, 0, n}];
Show[plot2, plot1]
```

Sparse Array means that we store only the nonzero entries of the matrix!

compare the effects of changing n

Also, use Normal[A] to change it to a dense matrix and compare the timing.





Extension: What if we had a differential equation in two variables, x and y? (on, e.g., the square $[0,1]\times[0,1]$) What would the matrix A look like?