Class 02, July 10, 2021

Submission - Class 01

- .py, .m, .tex, .pdf
- No late submission
- Due now

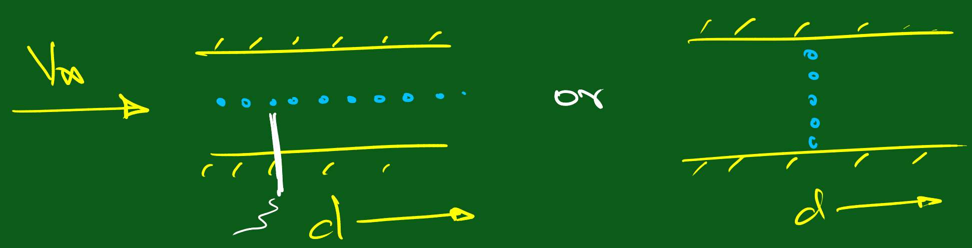
- "This is shown . . . "

Fig. ~10
Table ~4
100 ~m/s

- x^2 x**2   x*x   x/4 $\longrightarrow$ 0.25*x

  x^3 = x*x*x

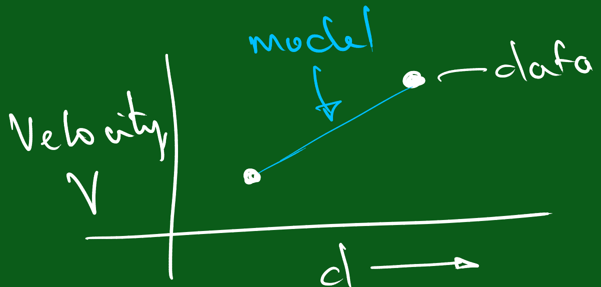
- Let us assume we are doing an experiment



How to find velocity?

- Hot-wire anemometer
  - ↳ Thermal transducer
  - ↳ Single point
  - ↳ Velocity - resistance - voltage
- Pitot static tube
- Venturimeter
- PIV - . . .

?



Eqn
↓
Tweak
↓
Least error

- linear eqn
- polynomial
- exponential

× Let us assume

$$V = \underbrace{m d + c}_{\text{to be estimated}}$$

N observations

$$V_1 = m d_1 + c$$
$$V_2 = m d_2 + c$$
$$\vdots$$
$$V_N = m d_N + c$$

$$\underbrace{\begin{bmatrix} d_1 & 1 \\ d_2 & 1 \\ \vdots & \vdots \\ d_N & 1 \end{bmatrix}}_{N\times 2} \underbrace{\begin{Bmatrix} m \\ c \end{Bmatrix}}_{2\times 1} = \underbrace{\begin{Bmatrix} V_1 \\ V_2 \\ \vdots \\ V_N \end{Bmatrix}}_{N\times 1}$$

$$\begin{bmatrix} d_1 & 1 \\ d_2 & 1 \\ \vdots & \vdots \\ d_N & 1 \end{bmatrix} \begin{Bmatrix} m \\ c \end{Bmatrix} = \begin{Bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{Bmatrix}$$

$$A \qquad x \quad = \bar{b}$$

* What does $Ax = \bar{b}$ mean in this context?

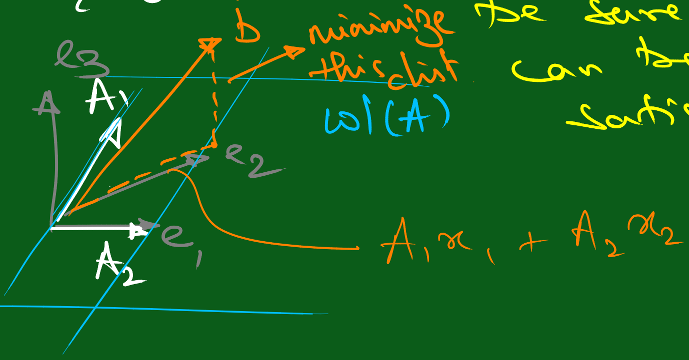
$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \vdots & \vdots \\ a_{N1} & a_{N2} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{Bmatrix}$$

$$\underbrace{\begin{Bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{N1} \end{Bmatrix}}_{A_1} \underline{x_1} + \underbrace{\begin{Bmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{N2} \end{Bmatrix}}_{A_2} \underline{x_2} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{Bmatrix}$$

* $\bar{b}$ is a linear combination of the columns of A!
* Example

$$\begin{Bmatrix} 0 \\ 1 \\ 1 \end{Bmatrix} x_1 + \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix} x_2 \approx \bar{b}$$

How can you be sure this can be satisfied?



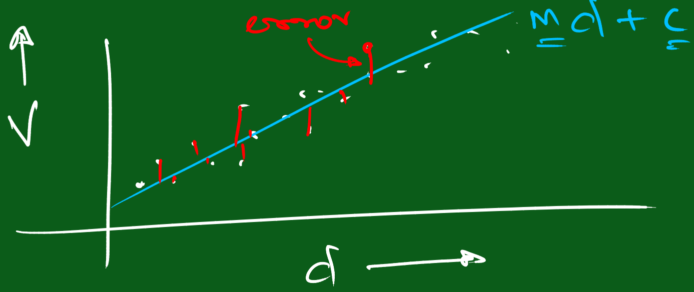
* $A_{N\times 2}\, x_{2\times 1} = b_{N\times 1}$

$$\underset{2\times 2}{(A^T A)}\, \underset{2\times 1}{x} = (A^T b)_{2\times 1}$$

How can you be sure this matrix is invertible?

$$x = (A^T A)^{-1} A^T b$$

solution for x!



m & c obtained such that it "best fits" the data

Linear Regression

$Ax = b$

find $\hat{x} = \arg\min_x \|b - Ax\|$

$Ax - b = \epsilon$ ← minimize this term

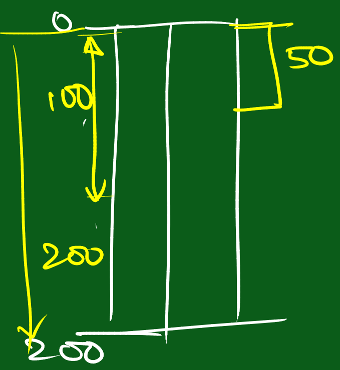
$\hat{x} = (A^T A)^{-1} A^T b$!

* $v = md^c \longrightarrow \log v = \underline{\log m} + \underline{c} \log d$

$v = md + c + kd^4 \longrightarrow$ valid?

* **Task**: data/scripts - class 02 - data 1.txt, 2.txt, 3.txt

200 (x, y)



- Perform linear regression using $y = mx + c$
- Using first 50, 100, 200 points
-

Data1	m	c
50		
100		
200		

D2		

D3		

files to be submitted

- .m
- .tex
- .pdf

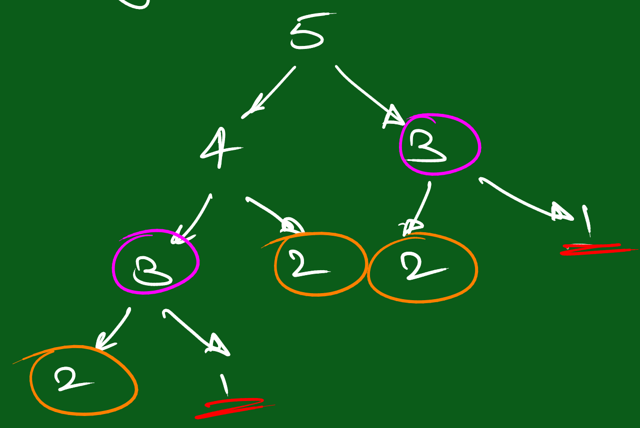
- Based on the behaviour of m & c, infer the type of data.
- Do you think a better model would work instead of $y = mx + c$? What? Why?
- Plot the data a scatter plot & a line for the model. Support previous inference
- Use octave
- Invert matrix using Gaussian-Elimination

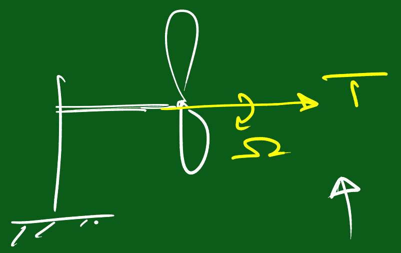
do not use inv(A) or A\b !

Class 03, July 17 2021

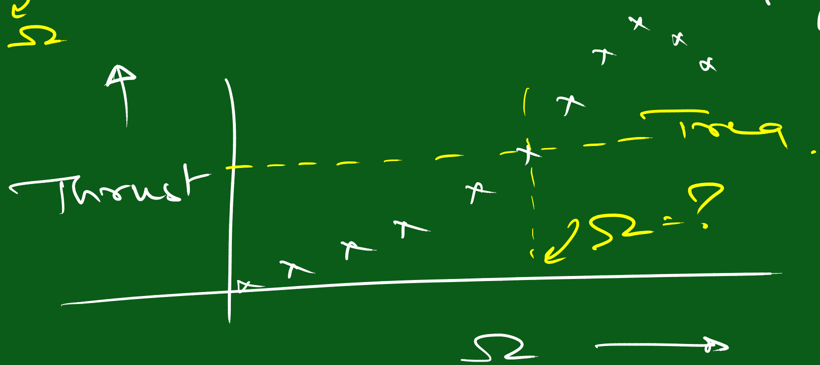
Submission Class 02
- .m, .tex, .pdf
- Due now.

× Demo on Python
× Root Finding Algorithms





What is the Ω so that $T = T_{req}$?



$$T(\Omega) = T_{req}$$

$$F(\Omega) = T(\Omega) - T_{req} = 0$$

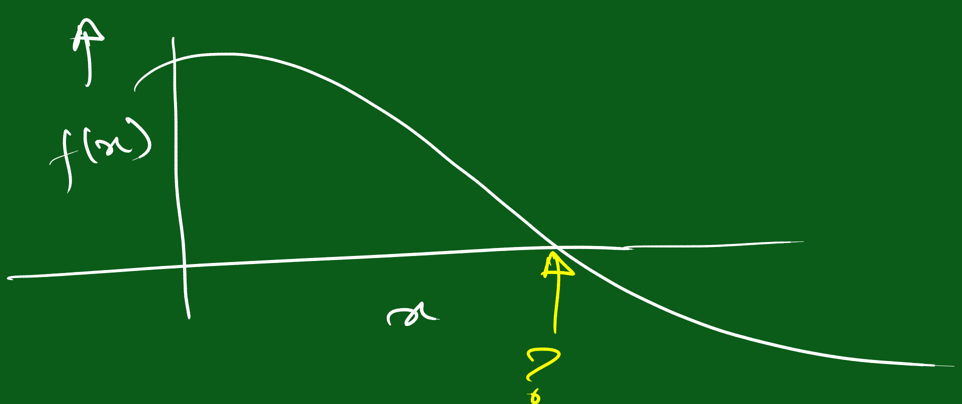
× In general we are interested in solving for

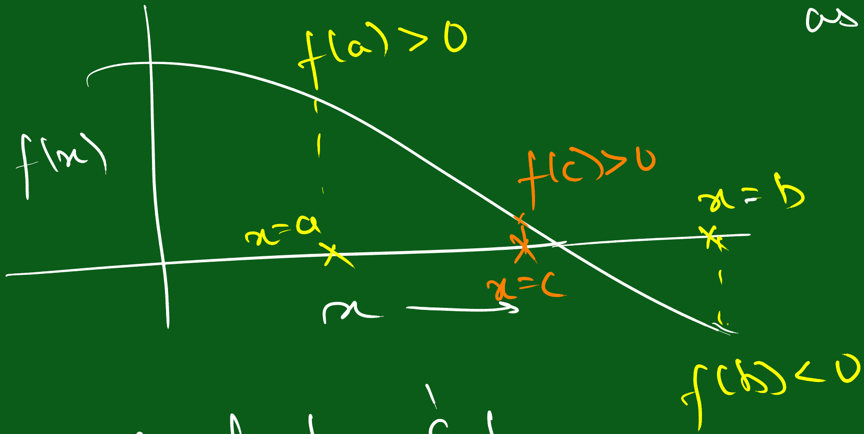
$$f(x) = 0$$

↳ scalar or vector

× Look at three methods — solving them numerically

1. Bisection method.





assume $f(x)$ continuous

$x = a,\ f(a) \geq 0$

$x = b,\ f(b) < 0$

At least one root

$x = c,\ c \in (a, b)$

Step 1: find a & b

Step 2: $c = \frac{a+b}{2}$ → if $f(c) > 0$ then $a = c$

→ if $f(c) < 0$ then $b = c$

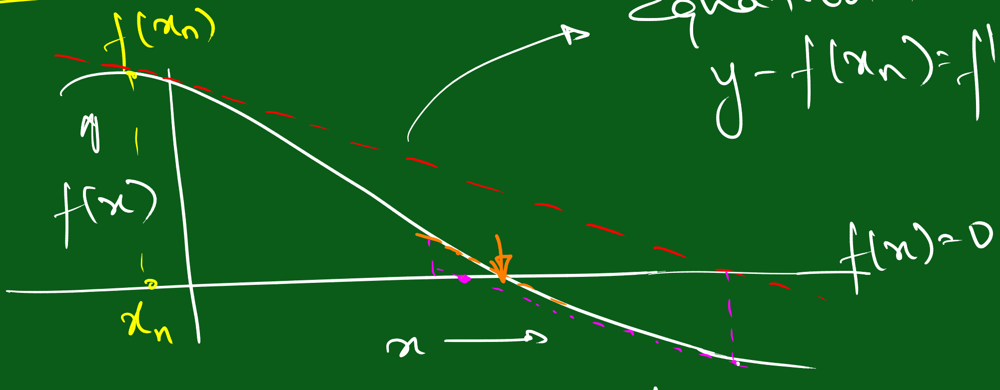
repeat till $f(c) = 0$

- limited precision
- we are okay with $|f(c)| < \epsilon$ $\sim 10^{-6}$

def bisector(f, a, b, ϵ, N_{max})

f: function; a: left; b: right; ϵ: precision; N_{max}: max step count

2. Newtons method



Equation to tangent

$$y - f(x_n) = f'(x_n)(x - x_n)$$

- we want to find $y = 0$

$$0 - f(x_n) = f'(x_n)(x_{n+1} - x_n)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

unbounded ← x_{n+1}

can be 0 ← $f'(x_n)$

also need derivative information

Start: x_0

x_1

$\vdots$

x_2

$\vdots$

$|f(x)| < \epsilon$

3. Secant Method →

Everything in Python

Report:
- Introduction (theory section with figs & equations) → must have a caption!
- Results & Analysis
- Conclusion
- References

Choose two unique starting sets

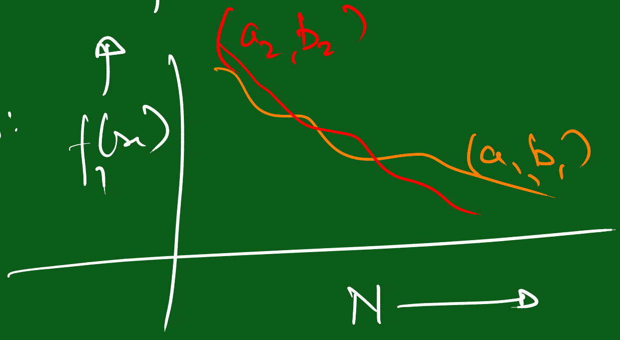
$$f_1(x) = x^3 - 3x^2 - x + 9$$

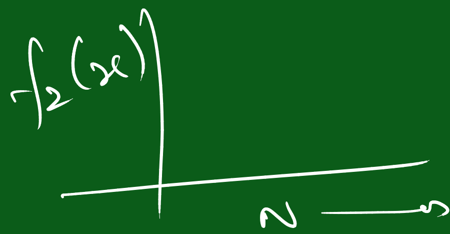
$$f_2(x) = e^x f_1(x)$$

Bisection:

Newton

Secant





$$f_3(x) = x^3 - 2x + 2$$

Start at $x = 0$