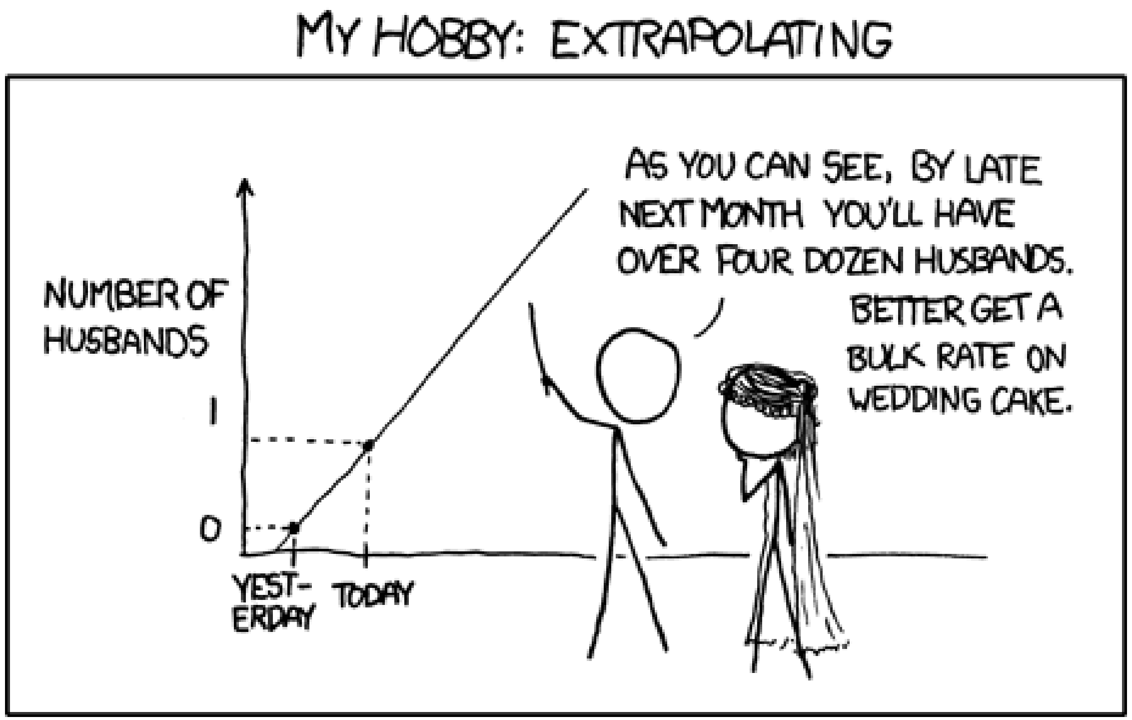
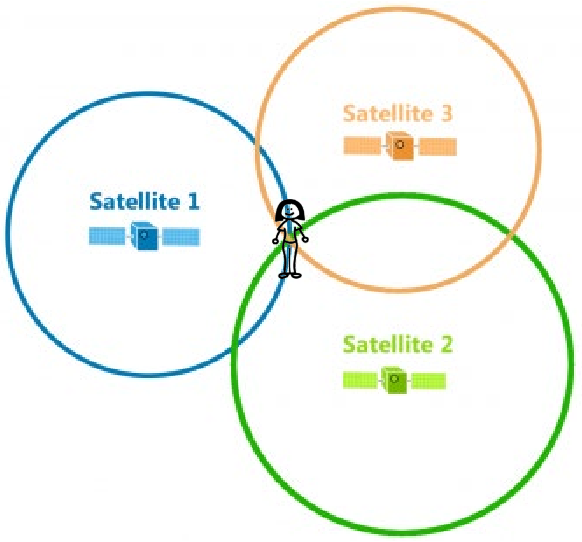


If she loves you more each and every day,
by linear regression she hated you before you met.

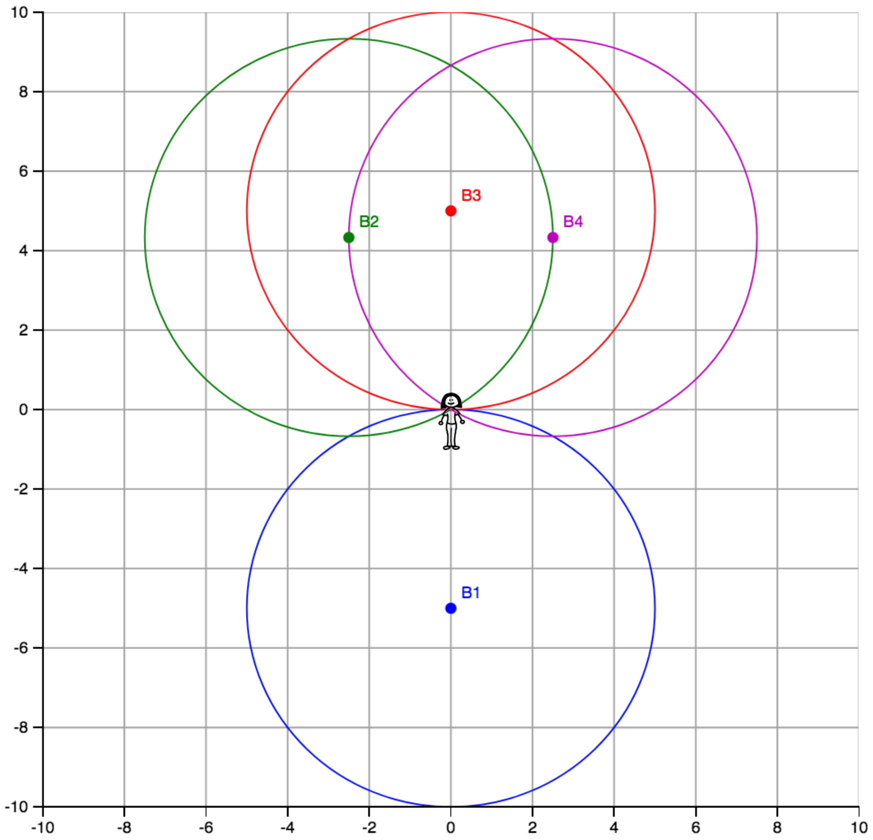
EECS 16A
Least Squares Algorithm

Last lecture: Trilateration



Finding my 2D position by calculating distances to 3 satellites with known positions:

Case 1: No noise gives unique solution

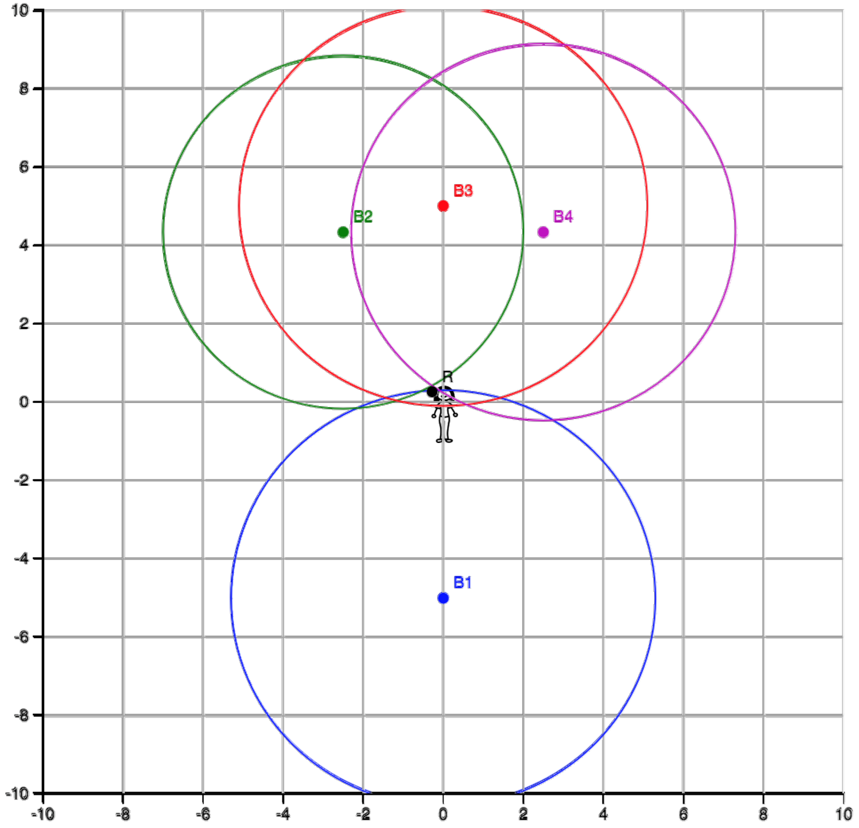


Correct measurements:

B1: 5m
B2: 5m
B3: 5m
B4: 5m

Least squares estimate:
(0,0)

Case 2: Noisy measurements



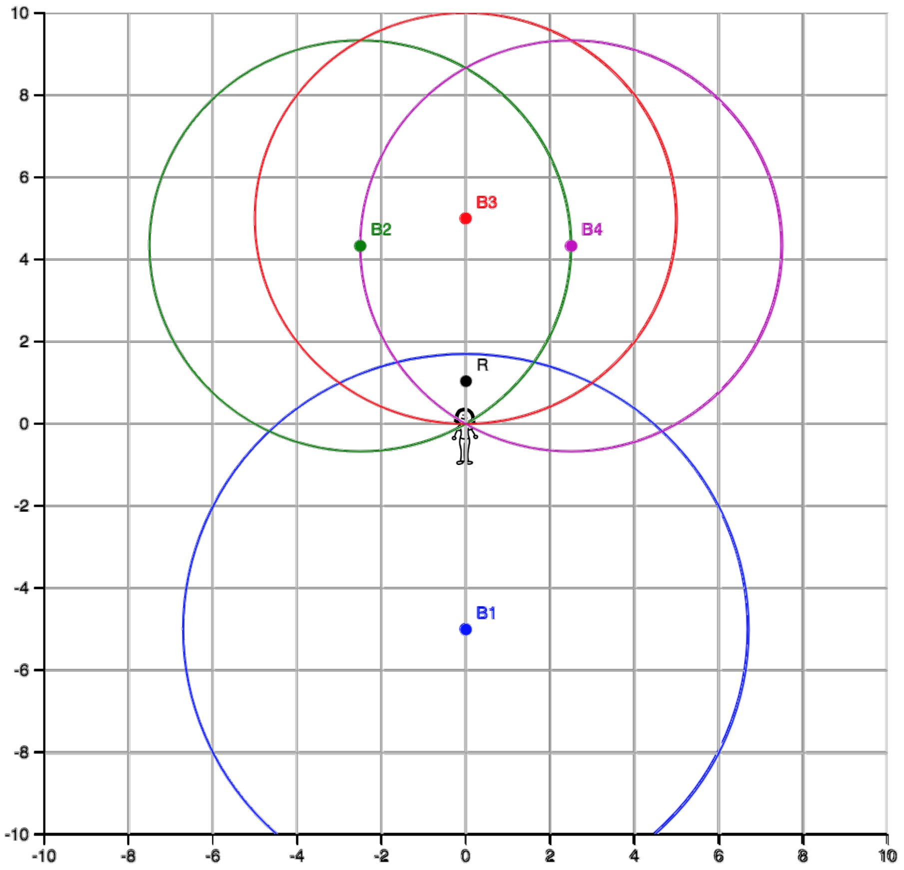
All measurements have error:

B1: 5.3m
B2: 4.5m
B3: 5.1m
B4: 4.8m

Least squares estimate:
(-0.28,0.26)

Estimate has some error, but will get smaller with more measurements (if error is random)

Case 3: Some noisy measurements



All measurements have error:

B1: 6.8m
B2: 5m
B3: 5m
B4: 5m

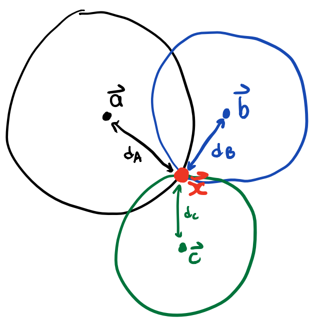
Least squares estimate:
(0,1.04)

Error is not spread evenly (random), if I knew 3 were correct, I would have gotten answer correct…

Last lecture:

Trilateration

Let's find my coordinates in 2D world, $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ from known distances d_A, d_B, d_C to 3 sattelites with known positions $\vec{a}, \vec{b}, \vec{c}$



$$\left.\begin{aligned} \|\vec{x}-\vec{a}\|^2 &= d_A^2 \\ \|\vec{x}-\vec{b}\|^2 &= d_B^2 \\ \|\vec{x}-\vec{c}\|^2 &= d_C^2 \end{aligned}\right\}$$

3 equations
2 unknowns
Problem: not linear!

Some math tricks

$$\underbrace{\begin{bmatrix} -2a_1+2b_1 & -2a_2+2b_2 \\ -2a_1+2c_1 & -2a_2+2c_2 \end{bmatrix}}_{\text{knowns}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\text{to solve for}} = \underbrace{\begin{bmatrix} d_A^2 - d_B^2 - \|\vec{a}\|^2 + \|\vec{b}\|^2 \\ d_A^2 - d_C^2 - \|\vec{a}\|^2 + \|\vec{c}\|^2 \end{bmatrix}}_{\text{knowns}}$$

2 eq'ns,
2 unknowns.
Linear ☺

Now we want to solve this $A\vec{x}=\vec{b}$ problem in the presence of measurement noise, and possibly for many satellite measurements (>3):

$$A\vec{x} = \vec{b} + \vec{e}$$

error due to noise (unknown)

$A\vec{x}=\vec{b}$ might have more equations than unknowns: $\underset{m\times n}{\boxed{A}}\ \underset{n\times 1}{\boxed{\vec{x}}} = \underset{m\times 1}{\boxed{\vec{b}}}$

Overdetermined System

*least-squares sol'n effectively does averaging

Least Squares Algorithm → finds the best estimate $\hat{x}$ such that $A\vec{x}$ is as close as possible to $\vec{b}$ (i.e. minimizes $\vec{e}$)

Want min $\|\vec{e}\|^2 = \|\vec{b}-\hat{b}\|^2 = \|\vec{b} - A\hat{x}\|^2$

$= \|A\vec{x} - \hat{b}\|^2$

estimate of $\vec{x}$

} solution will be given by projection of $\vec{b}$ onto $\vec{a}$ (for 1D case)

$\hat{b} = \alpha\vec{a}$
find α!

Projections

→ key idea in ML/SP, used for classification, etc.

→ find the component along a particular direction

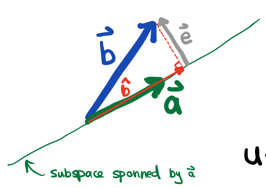
↳ What does it have to do w̄ inner product?

↳ looking for collinear component (largest inner product) or perpendicular component (smallest inner product)

(orthogonal)

"project $\vec{b}$ onto subspace spanned by $\vec{a}$" ← may not be in col(A)

scary ☹ let's go to geometry pic:



The projected vector $\hat{b}$ is collinear with $\vec{a}$ ($\hat{b} = \alpha\vec{a}$) ← in col(A) by design

and perpendicular to $\vec{e} = b - \hat{b}$ ($\vec{e} \perp \hat{b}$, $\vec{e} \perp \vec{a}$)

Using the property that perpendicular vectors have 0 inner prod:

$$\langle \vec{e}, \vec{a} \rangle = 0$$

$$\langle \vec{b} - \hat{b}, \vec{a} \rangle = 0$$

$$\langle \vec{b}, \vec{a} \rangle - \langle \hat{b}, \vec{a} \rangle = 0$$

$$\hat{b} = \alpha\vec{a}$$

$$\langle \vec{b}, \vec{a} \rangle = \langle \alpha\vec{a}, \vec{a} \rangle$$

$$\langle \vec{b}, \vec{a} \rangle = \alpha \langle \vec{a}, \vec{a} \rangle$$

$$\langle \vec{b}, \vec{a} \rangle = \alpha \|\vec{a}\|^2$$

scalar relating $\hat{b}, \vec{a}$

$$\boxed{\alpha = \frac{\langle \vec{b}, \vec{a} \rangle}{\|\vec{a}\|^2}}$$

$$\boxed{\hat{b} = \frac{\langle \vec{b}, \vec{a} \rangle}{\|\vec{a}\|^2}\vec{a}}$$

$$\|\hat{b}\| = \|\alpha\vec{a}\|$$
$$= \|\alpha\| \cdot \|\vec{a}\|$$
$$= \frac{|\langle \vec{a}, b \rangle|}{\|\vec{a}\|^2} \cdot \cancel{\|\vec{a}\|}$$
$$= \frac{|\langle \vec{a}, \vec{b} \rangle|}{\|\vec{a}\|}$$

1D sol'n to least squares

Need to generalize to multi-dimensional $A\vec{x} \approx \vec{b}$ and solve for $\hat{x}$

↑ matrix, not vector

Ex. In GPS, we have multiple satellites, so A has multiple cols

Theorem: Consider matrix A, vector $\vec{y} \in$ colspace(A)

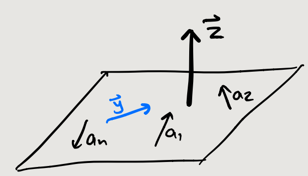
Then, consider vector $\vec{z}$

$$\left.\begin{array}{l} \langle \vec{z}, \vec{a}_1 \rangle = 0 \\ \langle \vec{z}, \vec{a}_2 \rangle = 0 \\ \vdots \\ \langle \vec{z}, \vec{a}_n \rangle = 0 \end{array}\right\}$$

$\vec{z}$ is orthogonal to all vectors in colspace(A)

↓ then

$$\langle \vec{z}, \vec{y} \rangle = 0$$



Proof: we $\boxed{\text{know}}$ $\vec{y} \in$ colspace(A), so it's a lin. combo. of cols:

$$\vec{y} = c_1 \cdot \vec{a}_1 + c_2 \cdot \vec{a}_2 + c_3 \cdot \vec{a}_3 \ldots c_n \cdot \vec{a}_n$$

scalar coeffs

we $\boxed{\text{want}}$

$$\langle \vec{z}, \vec{y} \rangle = 0 \longrightarrow \langle \vec{z}, \vec{y} \rangle = \langle \vec{z}, c_1\vec{a}_1 + c_2\vec{a}_2 + \ldots + c_n\vec{a}_n \rangle$$

scalar

$$= \langle \vec{z}, c_1\vec{a}_1 \rangle + \langle \vec{z}, c_2\vec{a}_2 \rangle + \ldots + \langle \vec{z}, c_n\vec{a}_n \rangle$$

$$= c_1 \langle \vec{z}, \vec{a}_1 \rangle + c_2 \langle \vec{z}, \vec{a}_2 \rangle + \ldots c_n \langle \vec{z}, \vec{a}_n \rangle$$

given

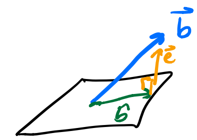
$$= c_1(0) + c_2(0) + \ldots c_n(0) = 0$$

✓ yay! ☺

Ok, but we need to find $\hat{x}$ from $\hat{b} = A\hat{x}$

Least Squares: minimize $\|A\vec{x} - \vec{b}\| = \|\vec{e}\|$

First, write A in terms of column view $A = \begin{bmatrix} | & | & & | \\ \vec{a}_1 & \vec{a}_2 & \cdots & \vec{a}_n \\ | & | & & | \end{bmatrix}$



$A\vec{x}$ is in col(A)

↳ search for $\hat{b} = A\vec{x}$ ← should be in col(A), even tho $\vec{b}$ is not

$\hat{b} + \vec{e} = \vec{b}$, $\vec{e} = \vec{b} - \hat{b}$

since $\vec{e} \perp$ col(A): $\left.\begin{array}{l} \langle \vec{a}_1, \vec{e} \rangle = 0 \\ \langle \vec{a}_2, \vec{e} \rangle = 0 \\ \vdots \\ \langle \vec{a}_n, \vec{e} \rangle = 0 \end{array}\right\}$ $\begin{array}{l} \vec{a}_1^T (\vec{b} - \hat{b}) = 0 \\ \vec{a}_2^T (\vec{b} - \hat{b}) = 0 \\ \vdots \\ \vec{a}_n^T (\vec{b} - \hat{b}) = 0 \end{array}$ $\xrightarrow{\text{write in mtx form}}$ $\begin{bmatrix} - & \vec{a}_1^T & - \\ - & \vec{a}_2^T & - \\ & \vdots & \\ - & \vec{a}_n^T & - \end{bmatrix} \begin{bmatrix} \vec{b} - \hat{b} \end{bmatrix} = \vec{0}$

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

$\underset{n \times m}{A^T} (\vec{b} - A\hat{x}) = \vec{0}$

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

$\underbrace{A^T A}_{n\times m \; m \times n} \hat{x} = A^T \vec{b}$

$n \times n$

SQUARE! and will be invertible when A has lin. ind. cols (Note 23)

Sometimes we want $\vec{b} = A\vec{x}$

$\boxed{\hat{b} = A(A^T A)^{-1} A^T \vec{b}}$

Least-Squares Solution!

$\boxed{\hat{x} = (A^T A)^{-1} A^T \vec{b}}$ ☺ ✓yay

$n \times 1 = \underbrace{\;}_{n \times n} \; n \times m \; m \times 1$

Example: $\underset{2\times1}{A} \underset{1\times1}{\vec{x}} = \underset{2\times1}{\vec{b}}$ $\begin{bmatrix} 2 \\ 1 \end{bmatrix} [x] = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$A = \underset{2\times1}{\begin{bmatrix} 2 \\ 1 \end{bmatrix}}$ $\vec{b} = \underset{2\times1}{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}$

If we did Gauss. Elim.:

$\left[\begin{array}{c|c} 2 & 1 \\ 1 & 1 \end{array}\right] \rightarrow \left[\begin{array}{c|c} 1 & 1/2 \\ 0 & 1/2 \end{array}\right]$ Inconsistent (no sol'n)

Or, note that $\vec{b} \neq$ col(A)

So let's find best estimate $\hat{x}$ by least squares:

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

$= \left(\frac{1}{5}\right) [2 \; 1] \begin{bmatrix} 1 \\ 1 \end{bmatrix}$

$= \left(\frac{1}{5}\right) 3$

$= 3/5$

$A^T = [2 \; 1]$

$A^T A = [2 \; 1] \begin{bmatrix} 2 \\ 1 \end{bmatrix} = 5$

$(A^T A)^{-1} = 1/5$

Example: $A\vec{x}=\vec{b}$

$$\begin{bmatrix}1&0\\0&1\\0&1\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix}=\begin{bmatrix}1\\2\\3\end{bmatrix}\quad \begin{matrix}x_1=1\\x_2=2\\x_2=3\end{matrix}$$

If we try Gauss. Elim.:

$$\left[\begin{array}{cc|c}1&0&1\\0&1&2\\0&1&3\end{array}\right]\rightarrow\left[\begin{array}{cc|c}1&0&1\\0&1&2\\0&0&1\end{array}\right]$$ ← Inconsistent (no sol'n!)

No $\vec{x}$ exists that solves $A\vec{x}=\vec{b}$!

Let's find the best estimate

Least-squares Algorithm:

estimate

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

$$(A^TA)^{-1}=\left(\begin{bmatrix}1&0&0\\0&1&1\end{bmatrix}\begin{bmatrix}1&0\\0&1\\0&1\end{bmatrix}\right)^{-1}=\left(\begin{bmatrix}1&0\\0&2\end{bmatrix}\right)^{-1}=\begin{bmatrix}1&0\\0&1/2\end{bmatrix}$$

$$=\begin{bmatrix}1&0\\0&1/2\end{bmatrix}\underbrace{\begin{bmatrix}1&0&0\\0&1&1\end{bmatrix}\begin{bmatrix}1\\2\\3\end{bmatrix}}$$

$$=\begin{bmatrix}1&0\\0&1/2\end{bmatrix}\begin{bmatrix}1\\5\end{bmatrix}$$

$$\hat{\vec{x}}=\begin{bmatrix}1\\2.5\end{bmatrix}$$

$\hat{\vec{x}}$ is the sol'n that minimizes $\|\vec{e}\|^2$

Average of measurements of x_2!

Least squares is first attributed to Gauss (1800s)

↳ scientist Piazzi tracked a bright spot b/w orbits of Mars & Jupiter, thinking it might be a new planet. (it was Ceres, not a full planet, in asteroid belt)

↳ he missed a few days when he got sick, lost some days due to sun obscured

↳ so he published data, and others tried to calculate future position from existing data

↳ Gauss won the competition by inventing least squares

How did Gauss find Ceres? fit to Kepler's Laws (elliptical orbits)

$$ax^2+by^2+cxy+dx+ey=1$$ Ellipse eq'n

since squared terms are known → finding coeffs is linear problem

Position (knowns!)

coefficients (unknowns!)

How to set up least squares problem?

Let's put unknowns into a vector: $\begin{bmatrix}a\\b\\c\\d\\e\end{bmatrix}$

Write some equations for measured position (x_1,y_1): $ax_1^2+by_1^2+cx_1y_1+dx_1+ey_1=1$

↳ There are 22 measurements in dataset, so let's put in a matrix:

$$\begin{bmatrix} x_1^2 & y_1^2 & x_1y_1 & x_1 & y_1 \\ x_2^2 & y_2^2 & x_2y_2 & x_2 & y_2 \\ & & \vdots & & \\ x_{22}^2 & y_{22}^2 & x_{22}y_{22} & x_{22} & y_{22} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$

22×5 5×1 22×1

solve for me!

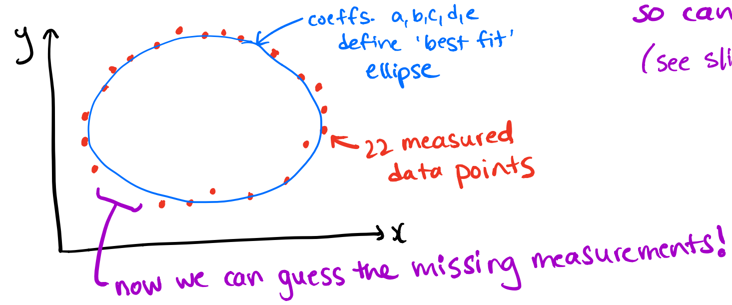
in machine learning, cols are called 'features'

22 equations, 5 unknowns

'Overdetermined'

*Gauss did this by hand! We have Jupyter notebooks so can be lazy ☺ yay!

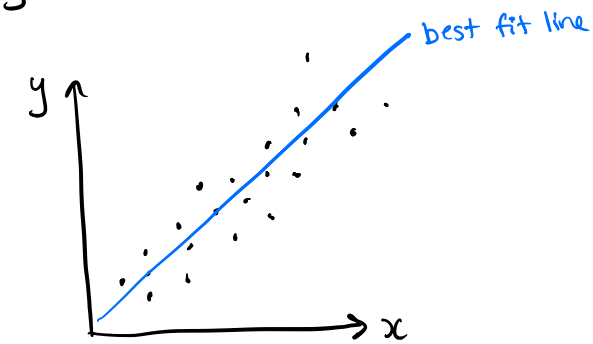
(see slides or 'Ceres-orbit' notebook)



Least-squares is building block for all of signal processing / machine learning / pattern matching

Linear regression

↳ fit to a line



$$y = mx + c$$

Known: (x_1, y_1), (x_2, y_2), $\vdots$, (x_n, y_n) unknown: $\begin{bmatrix} m \\ c \end{bmatrix}$

$$\underset{A}{\begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ x_3 & 1 \\ \vdots & \\ x_n & 1 \end{bmatrix}} \underset{\vec{w}}{\begin{bmatrix} m \\ c \end{bmatrix}} = \underset{\vec{b}}{\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}}$$

solve for me!

Best estimate for $\hat{\vec{w}} = (A^T A)^{-1} A^T \vec{b}$ least squares