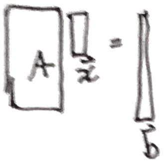
# M3 Lecture 5

①

Last time: **Least Squares**

use for OVERdetermined case:



more eqn's than unknowns
↳ inconsistent (no sol'n)

error $\vec{e} = A\vec{x} - \vec{b}$ ← how inconsistent are measurements?

to minimize me

find $\min_{\vec{x}} \|\vec{e}\|^2 = \min_{\vec{x}} \|Ax-b\|^2$

Find me! → $\vec{x}$

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

LS solution → "Pseudo inverse"

**Minimum Norm Solution**

use for UNDERdetermined case:

→ ∞ soln's, pick one → 'smallest'

$\min_{\vec{x}} \|\vec{x}\|^2$ such that $A\vec{x} = \vec{b}$

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

---

Today:
- Internet of Things example (IOT)
- Orthogonal Matching Pursuit (OMP)

IOT: - one central "homebox" (HB) analogous to GPS reciever) that recieves signals from various 'smart' devices (e.g. fridge, temp. sensor, roomba, sprinklers)

- at most 'k' devices 'on' at once.

beacons → devices/sensors
reciever → homebox
shifts → shifts (give distance?)

★ But sensors want to send their actual values... → e.g. Temp = 72° **How?**

- shifts? No, need to be long, can't do trilateration, delays
- strength of signal!

Goals: ① Identify devices 'on'
② Estimate the values (messages) sent
③ Locate device?

e.g. $\vec{s} = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} \longrightarrow \alpha\vec{s} = \begin{bmatrix} \alpha \\ 0 \\ 2\alpha \end{bmatrix}$

Example Situation:

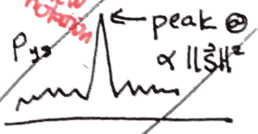
→ We have 2,000 sensors talking to HB

i.e. 2,000 songs: $\vec{S}_0, \vec{S}_1, \ldots \vec{S}_{1999}$

device ID

Recieved Signal: $\vec{y} = \alpha\vec{S}$

new notation



Cross-corr: $\vec{\rho}_{\vec{y}\vec{s}} = C_{\vec{s}}\vec{y} = \begin{bmatrix} \alpha\|\vec{s}\|^2 \\ \vdots \end{bmatrix}$

How long should song be? 2000 samples?

→ longer → more orthogonal, but slower, more computation
- doesn't need to be 2000.

Let's use 400 samples as length.

$\vec{S}_i \in \mathbb{R}^{400}$ ↖ dimension of song is large

Say only device 300 is transmitting from distance $d_{300}$, causing shift of $N_{300}$ sample

↳ with a value of a (e.g. temp.)

Recieved Signal $\vec{y} = a\,\vec{s}_{300}^{\,(N_{300})}$ ← shift gives distance

↑ message value we wish to transmit

↑ device ID (300 is 'on')

e.g. $\vec{s} = \begin{bmatrix} 1 \\ 1 \\ -1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$ ↑ use all ±1 $\quad a\vec{s} = \begin{bmatrix} a \\ a \\ -a \\ a \\ -a \\ -a \end{bmatrix}$

How to find device ID? do cross-corr.

$$\vec{\rho}_{\vec{r},\vec{s}_{300}}[N_{300}] = C_{\vec{s}_{300}\vec{y}}[N_{300}] = \left\langle a\vec{s}_{300}^{\,(N_{300})}, \vec{s}_{300}^{\,(N_{300})} \right\rangle$$

$$= a\left\langle \vec{s}_{300}^{\,(N_{300})}, \vec{s}_{300}^{\,(N_{300})} \right\rangle$$

$$= a\,\|\vec{s}_{300}\|^2$$

$$= a \times 400$$

↑ let's just look @ correct shift (others small → Boring)

← Length of song!

↑ so I can find a!

More generally:

Say there are only k devices on at once: $\boxed{\text{SPARSITY}}$ → many zeros makes life easier, only look for a few

devices: $i_1, i_2, i_3 \ldots i_k$ ← deviceID

shifts: $N_{i1}, N_{i2} \ldots N_{ik}$

messages: $a_{i1}, a_{i2} \ldots a_{ik}$

$$\vec{y} = a_{i1}\vec{s}_{i1}^{\,(N_{i1})} + a_{i2}\vec{s}_{i2}^{\,(N_{i2})} + \ldots + a_{ik}\vec{s}_{ik}^{\,(N_{ik})}$$

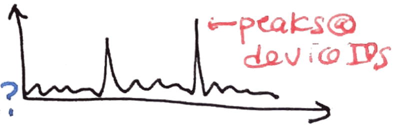
Will we be able to disambiguate these? I hope so!

Algorithm 1: ① cross-correlate with all songs (2000)

↳ if 'significant' peak, device is 'ON'

i.e. Find max$[C_{\vec{s}\vec{y}}]$ ← vector

what if a value is small? Trouble!



↳ for each song. Peaks ~~get~~ (should) give a values, but won't be exact... $\boxed{\text{Why not?}}$ Imperfect orthogonality

Normalized to 1 = same

$\boxed{\text{Can a values be found exactly?}}$ If songs orthogonal, yes! But they're not..

[illegible]

$\boxed{\text{What to do?}}$ Least squares? Not exaaactly...

Consider finding $a_{i_1}$ from recieved signal $\vec{y}$ when $k$ devices on:

Let's look at corr. w̄ correct shift $i_1$:

$$\langle \vec{s}_{i_1}, \vec{y} \rangle = \underset{\text{want}}{\boxed{a_{i_1}}} \, \underset{\text{known}}{\| \vec{s}_{i_1} \|^2} + \underbrace{a_{i_2} \langle \vec{s}_{i_1}^{\,(N_{i_1})}, \vec{s}_{i_2}^{\,(N_{i_2})} \rangle + \dots a_{i_k} \langle \vec{s}_{i_1}^{\,(N_{i_1})}, \vec{s}_{i_k}^{\,(N_{i_k})} \rangle}$$

Should be small, but only ZERO if orthogonal! Even many small values here will corrupt estimate of $a_{i_1}$

What to do to get rid of them? Anything?

Summary Algo ~~1 attempts~~ using cross-corr:

1) Find index of 'on' devices → device IDs $i_1, i_2, \dots i_k$

2) Find shifts $N_{i_1}, N_{i_2}, \dots N_{i_k}$

3) Let's define $\vec{z}_1 = \vec{s}_{i_1}^{\,(N_{i_1})}$, $\vec{z}_2 = \vec{s}_{i_2}^{\,(N_{i_2})}$, … $\vec{z}_k = \vec{s}_{i_k}^{\,(N_{i_k})}$

simpler notation

assume at correct shift

Recieved signal is

$$\vec{y} = \begin{bmatrix} | & | & & | \\ \vec{z}_1 & \vec{z}_2 & \cdots & \vec{z}_k \\ | & | & & | \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix}$$

message values from devices

assume we got these already from correlations

Note: This #cols is $\underline{k}$, not 2000

solve for these!

$$\vec{y} = Z \vec{a}$$

← System of linear eqns!

How to solve? GE? or $\underline{\text{LS}}$!

If noise, LS necessary:

$$\vec{y} = Z\vec{a} + \underset{\text{noise}}{\vec{n}} \rightarrow \boxed{\hat{a} = (Z^T Z)^{-1} Z^T \vec{y}}$$

Least-squares sol'n?

done?

**Algorithm #2**

1) cross-correlate to identify devices + shifts

2) Use those to estimate $\hat{a}$ with LS

$\underline{\text{Problem}}$: sometimes one $a$ >> others & drowns out rest.

What to do? Remove it's effect

↳ corr. noise from non-ortho is > peak of other device.