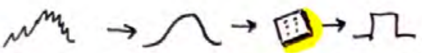
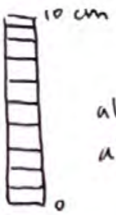

filtering to yes/no


adding probability

why? Reduce false positives + false negatives.



Model vs. simulation vs. theory + reality.

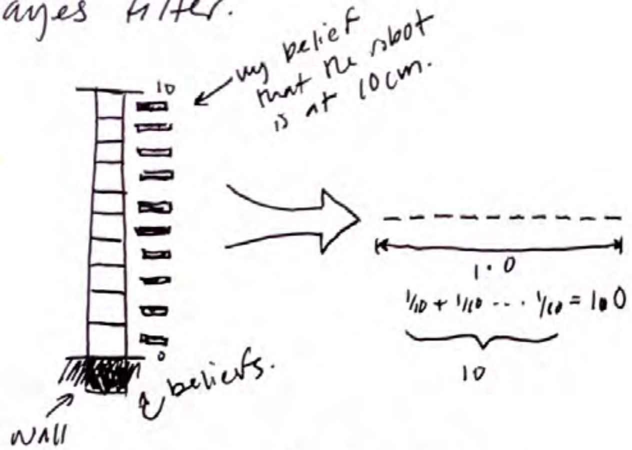
ahead of time: 1/10 each.
after a pulse: higher probability for some location.

Particle filters: motivation demo.

↳ Explain each error and how it would show up on your robot (up to 1° sim)

→ Experimental sensor simulation. dist.
→ particle filter.

Bayes filter.



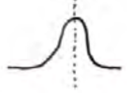
sensor() → 8 cm

no filter →
= threshold filter.


1.0 for 8cm
0 elsewhere

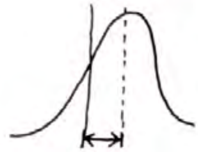
Assume: error is normally distributed:

sensor() = 8 cm



$$e^{\frac{-1}{2}\cdot\frac{(pos - sensor)^2}{\sigma^2}}$$

fancy way of measuring dist. from avg. in 0 - 1.0 units.



b[i] model (sensor, pos) × b[i]

$\downarrow$

b[10] = model (8, 10 9) → 0.9231 × 1/10

b[9] = model (8, ~~10~~) → 0.9802 × 1/10

b[8] = model (8, 8) → 1.0000 × 1/10

b[7] = model (8, 7) → 0.9802 × 1/10

⋮

b[1] = model (8, 1) → 0.3753 × 1/10

b[0] = model (8, 0) → 0.2780 × 1/10

Hey! That doesn't add up to 1.0

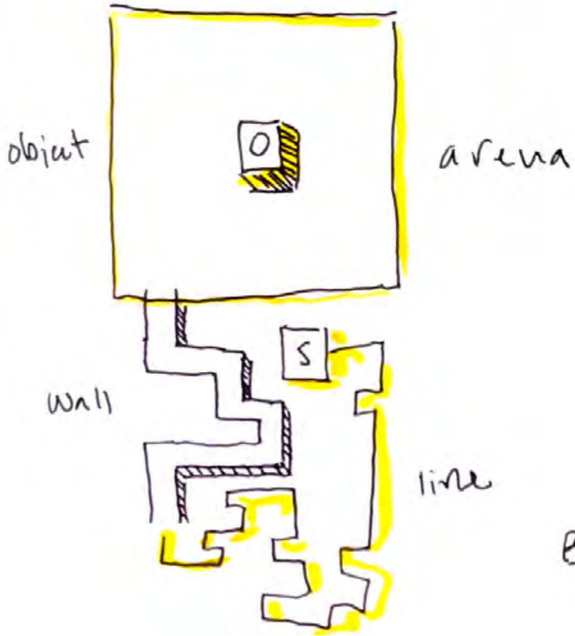
normalize: each $b[i] = \frac{b[i]}{sum(b)}$

That's it! Now when you run again:

sensor() = 7cm

reinforce previous beliefs!

Different methods add steps to deal with this problem.



There are a number of beliefs you will have to maintain.

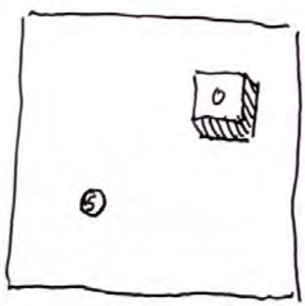
you can use Bayes to model them all.

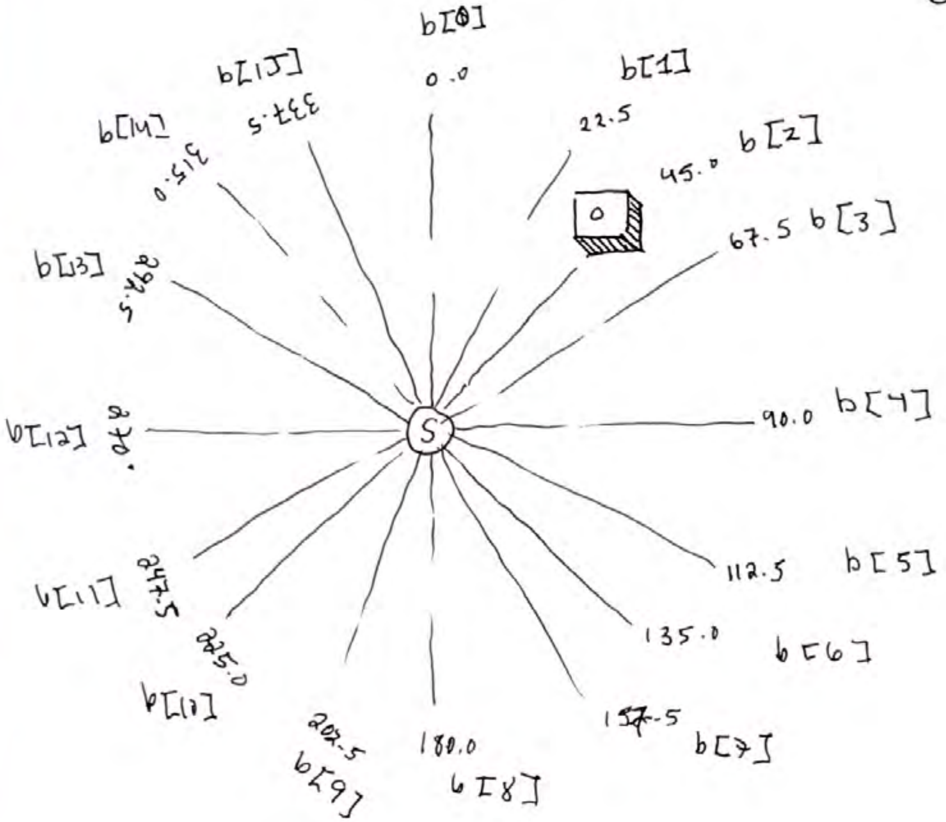
Brainstorm now.

1. On or off line.
2. In line maze, wall maze, or object arena.
3. position of the object
4. in or out of arena.

* think sonar DC.

How to set up Bayes if your robot was a fixed sonar tower?





$$\text{belief}[i] = \frac{\text{dist}}{\text{max}} \times \text{belief}[i]$$

normalized

then

$$\text{belief}[i] = \frac{\text{belief}[i]}{\text{sum}(\text{belief})}.$$

⇒ Any measurable state can be a belief.

Philosophical Question: We say "model", "belief", etc.

How much does the model represent reality?

coin toss:

$$P(H) = \frac{1}{2} \qquad P(T) = \frac{1}{2}$$

But outcomes are really: $\{H, T, ?\}$

$$P(H) = \frac{49}{100} \qquad P(T) = \frac{49}{100} \qquad P(?) = \frac{2}{100}$$

or $$P(H) = \frac{1}{2} - \varepsilon \qquad P(T) = \frac{1}{2} - \varepsilon \qquad P(?) = 2\varepsilon$$

$$\varepsilon = 0.0000\ldots1$$

No matter what, our model is framing something.

out of what we measured, X. experimental.

frame.

This becomes blindingly clear w/ the robot...
but applies to every model!

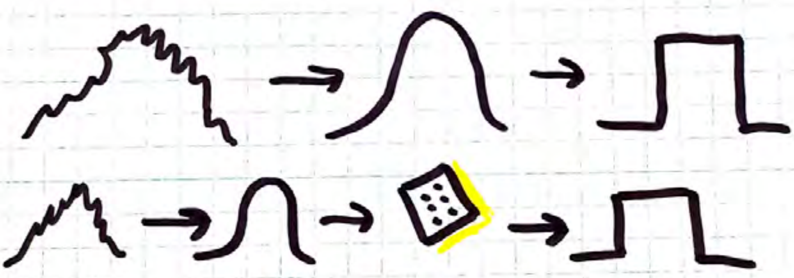
COGS 300 — **Detection 02** — Oct 9/25

Warm up: Draw non-rectangular grids. Eg. choose a polygon + project lines:

Notice: what "looks" even vs. what is measurably even?

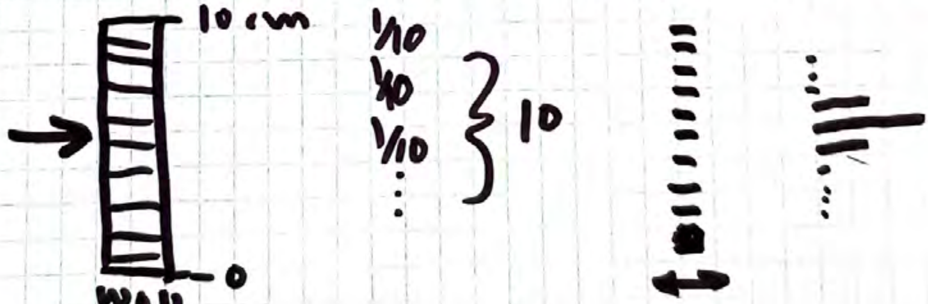
Try a few!

if-then

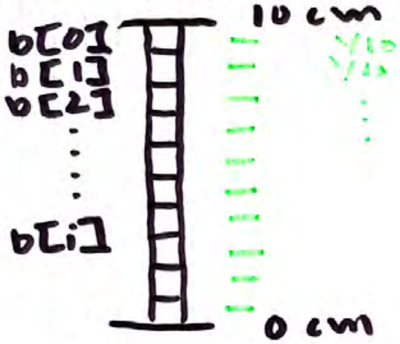


filtering to yes-no

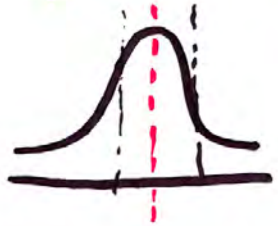
reduce false positives + false negatives.



beliefs[i] = b[i]



$\frac{1}{10} + \frac{1}{10} \cdots \frac{1}{10} = 1.0$



$\sigma = 5$

error

$$\text{model(sensor, pos)} = e^{-\frac{1}{2}\cdot\frac{(\text{pos}-\text{sensor})^2}{\sigma^2}}$$

0–1.0

b[i] = model(s, dist[i]) × b[i]

s = sensor() = 8cm

b[0] = model(8cm, 10cm) × b[0]

= 0.9231 = 1/10

b[1] = model(8cm, 9cm) × b[1]

= 0.9802 = 1/10

b[2] = model(8cm, 8cm) × b[2]

1.0 1/10

b[3] = model(8cm, 7cm) × b[3]

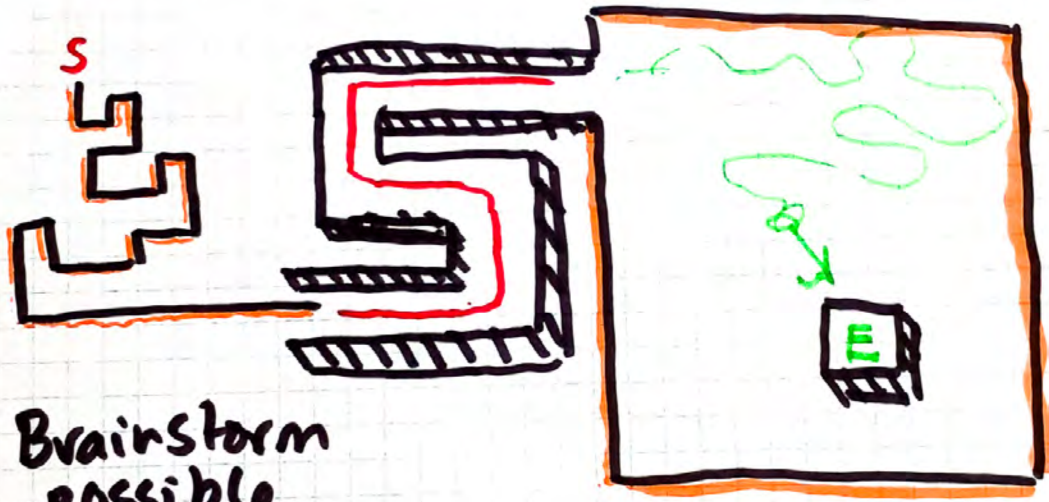
normalize

$$b[i] = \frac{b[i]}{sum(b)}$$

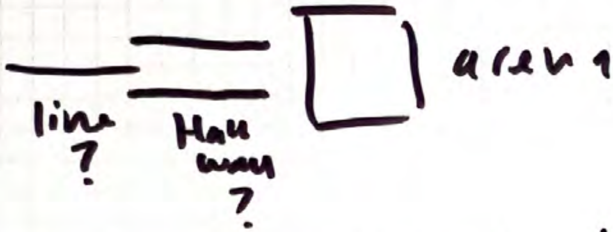
$$0.9231 \times \frac{1}{10}$$

$$= 0.09231$$

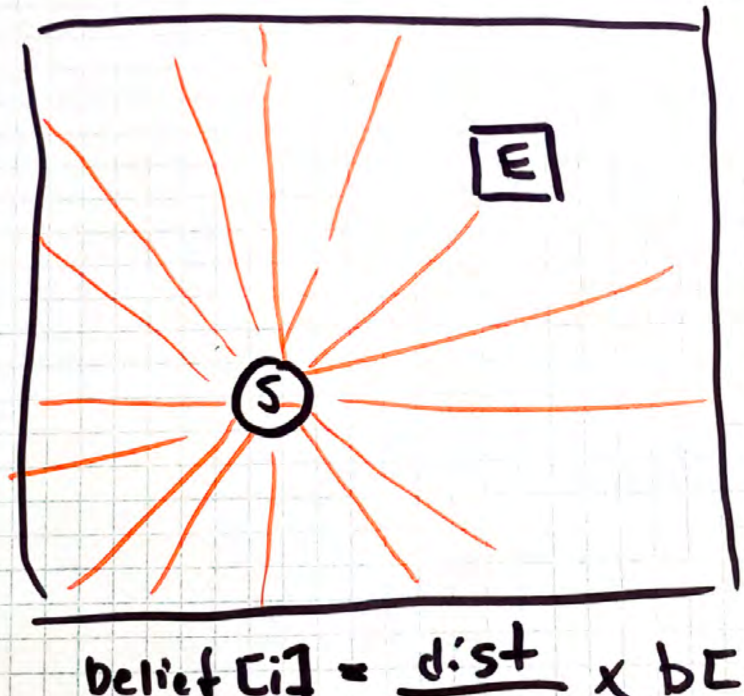
Tournament



Brainstorm possible beliefs.



b[0] = line belief 1/3
b[1] = hallway 1/3
b[2] = arena 1/3



$$\text{belief}[i] = \frac{\text{dist}}{\underset{0-1}{\text{max}}} \times \underline{\underline{b[i]}}$$

5

model — physical
\ representational
\ reductive

simulation

belief ↗

$$P(H) = 1/2 - \varepsilon$$
$$P(T) = 1/2 - \varepsilon$$
$$P(?) = 2\varepsilon$$