Last Time:

- ☐ Mathematical human models
- ☐ Boltzmann model

Lecture 5
HRI, Fall '24
Andrea Bajcsy

Today:

- ☐ Trajectory forecasting
- ☐ planning with forecasts

REMINDERS

- project proposal (1 page, 0% grade) - Tues, Sept 24
  - ↳ Discuss w/ me + Pranay!
- guest lecture Prof. Dylan Losey - Tues, Sept 24
- HW 1 - Tues, Oct 8

We have learned about how to mathematically model humans. BUT WHY?

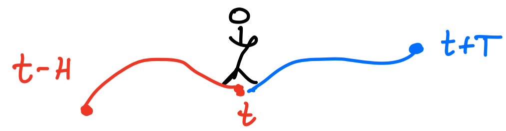
A large reason why we want to model human behavior in robotics, is to <u>predict</u> human behavior so robots can <u>plan</u> or <u>make decisions</u> around them!

A large subfield interested in building human prediction models is called trajectory forecasting.

<u>Trajectory forecasting</u> seeks to design or learn from data a model $\mathcal{P}$ which given (a history of) observations about interaction, yields future behaviors of the person.

$$\mathcal{P} : \mathcal{O} \longmapsto \mathcal{X}_H$$

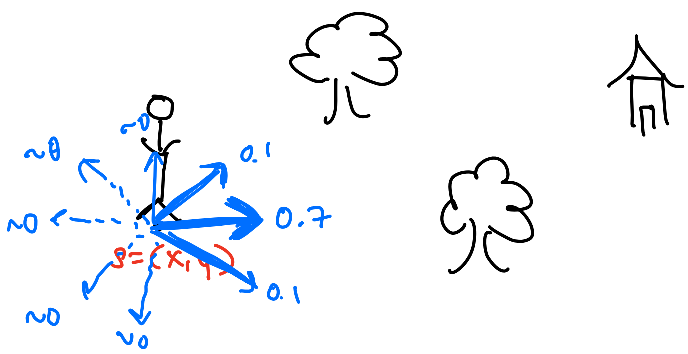
observation space (e.g. XY positions) human states or actions



Ⓠ Have we already seen a prediction model? If yes, what is $\mathcal{O}$ and $\mathcal{X}_H$?

Ⓐ Yes! The <u>Boltzmann model</u>.

$$\underset{\substack{\uparrow \\ a \in \mathcal{A} := \mathcal{X}_H}}{\pi(a \mid} \overset{\substack{s \in S := \mathcal{O} \\ \downarrow}}{s}) = \frac{e^{Q(s,a)}}{\sum_{\bar{a} \in \mathcal{A}} e^{Q(s,\bar{a})}}$$



Ⓠ Wait, this model only gives me <u>actions</u>. But what if I want to know their future <u>state</u>?

Ⓐ Combine $\pi(a \mid s)$ with transition function $P(s' \mid s, a)$ to get predicted future states!

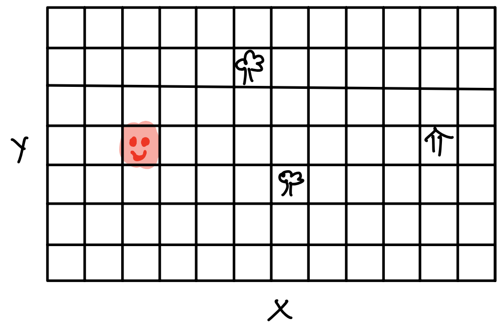
EXERCISE: Assume I give you 3 things:

(1) $P(s^{t+1} \mid s^t, a^t)$ transition function

(2) $\pi(a \mid s)$ the Boltzmann distribution over actions.

(3) $P(s^0)$ prior over initial states

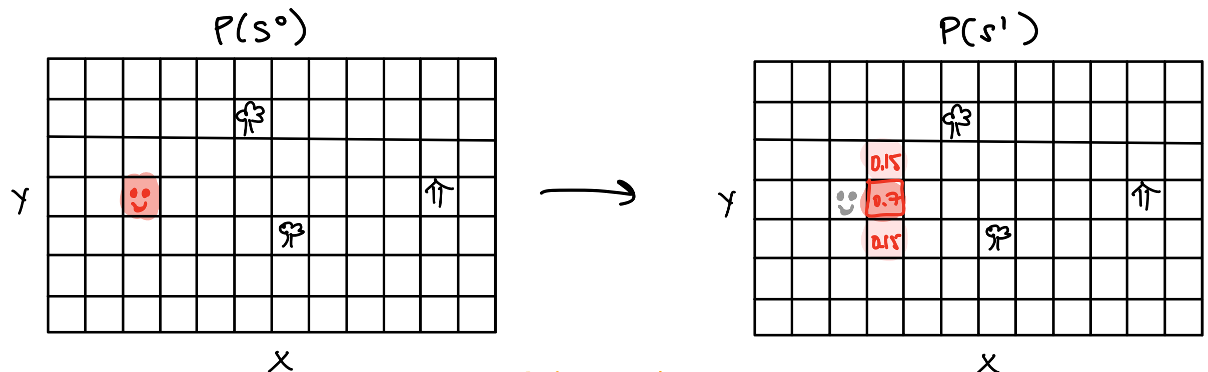
$$P(s^0) = \begin{cases} 1 & \text{if } s^0 = \text{☺} \\ 0 & \text{else} \end{cases}$$



Can you compute $P(s^1)$ with just (1-3) & our basic prob. rules?

$$P(s^1) = \sum_{s^0 \in S} \sum_{a^0 \in A} P(s^1, a^0, s^0)$$ // marginalize over s^0, a^0

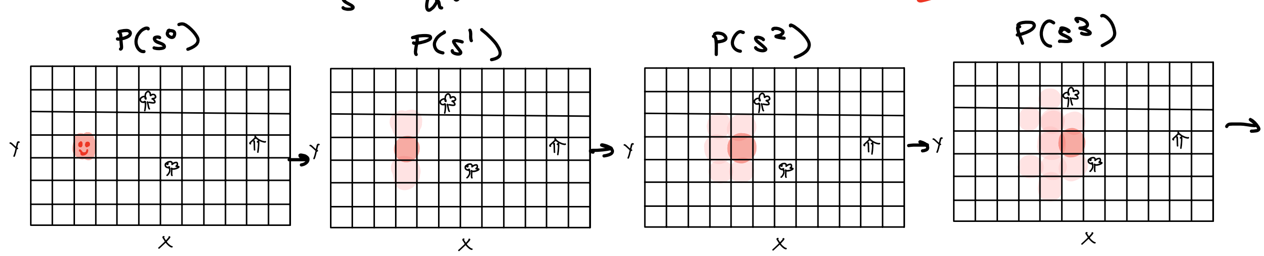
$$= \sum_{s^0 \in S} \sum_{a^0 \in A} \underbrace{P(s^1 \mid a^0, s^0)}_{\text{transition function}} \underbrace{P(a^0 \mid s^0)}_{:= \pi(a \mid s)} \underbrace{P(s^0)}_{✓ ☺}$$ // apply product rule



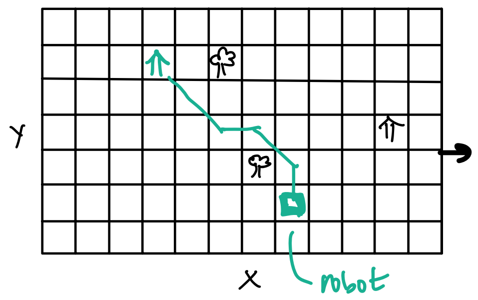
Assume $P(s^1 \mid s^0, a^0)$ is deterministic

$\pi(a \mid s)$

In general, $P(s^{t+1}) = \sum_{s^t} \sum_{a^t} P(s^{t+1} \mid s^t, a^t) P(a^t \mid s^t) \boxed{P(s^t)}$ computed recursively!

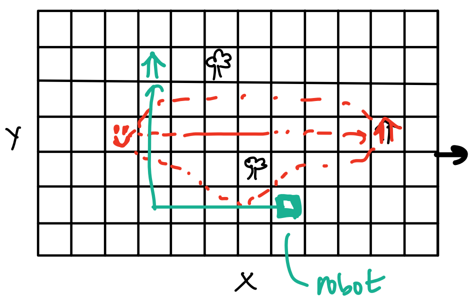


Now, we want to plan with this!



BEFORE:

$$R(s_R, a_R) = \mathbb{1}\{s_R = \Uparrow\} - 10 \cdot \mathbb{1}\{s_R = \text{🌳}\}$$



NOW:

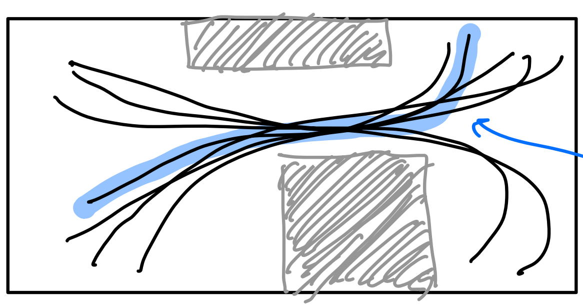
$$R(s_R, a_R) = \mathbb{1}\{s_R = \Uparrow\} - 10 \cdot \mathbb{1}\{s_R = \text{🌳}\} - \alpha \cdot \underbrace{P(s_H = s_R)}_{\text{prediction model!}}$$

DATA-DRIVEN PREDICTION

So far, we have modelling / making assumptions about the structure of an agents' decision-making (e.g. they are optimizing a reward!)

But, as data of human behavior explodes (e.g. AV datasets, Youtube/TikTok, motion capture), its tempting to take a step back & minimize the behavioral assumptions we make when constructing a trajectory prediction model, P.

Assume I get a dataset of <u>TONS</u> of human behavior!



$$\mathcal{D} := \{\xi_1, \xi_2, \xi_3, \dots \xi_N\}$$

sequence of states $\xi = (s_0, s_1, s_2, \dots s_T)$

Lets <u>directly fit</u> (e.g. regression) the prediction model P to predict the future states $\underbrace{s^{t+1}, s^{t+2}, \dots s^{T}}_{x_H}$ from the past $\underbrace{s^{0}, s^{1}, \dots s^{t}}_{\theta}$.

To do this, we parameterize our predictor P_θ w/ params θ.

<u>NOTE</u>: before we often modelled the parameter θ as goals! But now, its much more general (e.g. θ could be weights of NN).

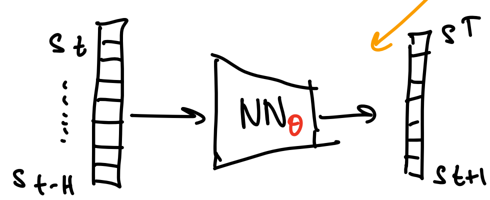
$\boxed{\text{GOAL}}$ Find values of θ such that the prediction model P outputs:

$$P_\theta(s^{0:t}) \longmapsto \hat{s}^{t+1:T}$$

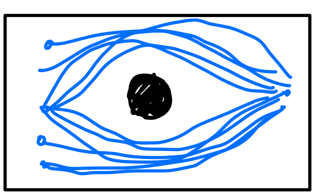
to align with the actual observed dataset of behavior, $\mathcal{D}$.

We can divide each trajectory in $\mathcal{D}$ into $(\underbrace{s^{0:t}}_{\text{INPUT}}, \underbrace{s^{t+1:T}}_{\text{OUTPUT}})$ pairs

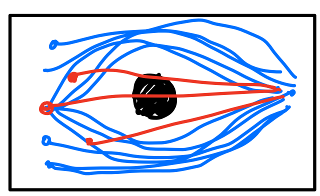
$$\min_{\theta} \sum_{\xi \in \mathcal{D}} \text{DIST}(\underbrace{P_\theta(s^{0:t})}_{\hat{s}^{t+1:T}}, s^{t+1:T})$$



We now need to be careful about NN_θ architecture choice!



$\Rightarrow$ P_θ will degenerate!



If NN_θ naively chosen, can't handle multimodality in human behavior

ENTER: generative models

(e.g. GANS, CVAEs, GMMs, Transformers)

often used in enc-decoder but
features from TF decoder are used to fit GMM which represents the state predictions

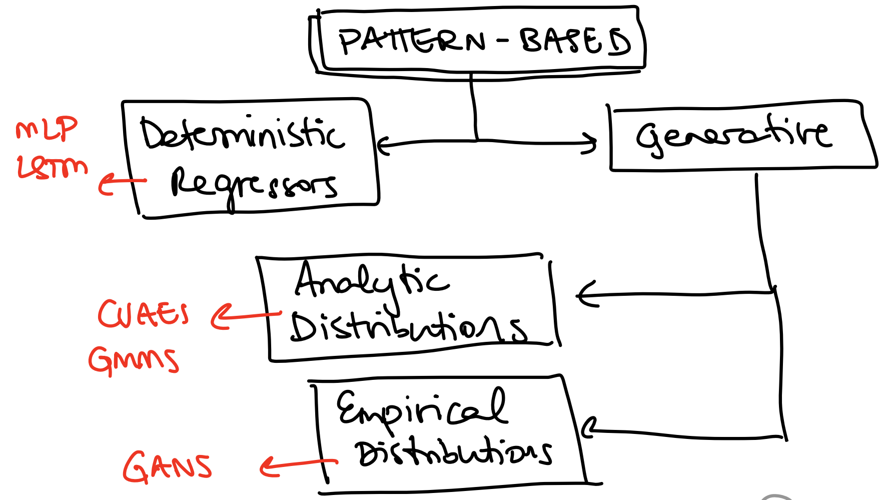
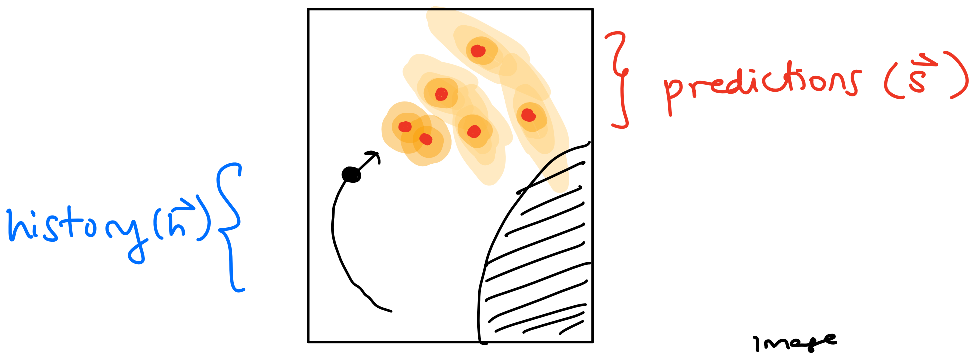
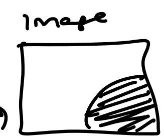


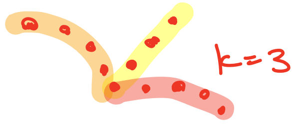
figure from ⊛ Boris Ivanovic!

One structure for a data-driven human predictor looks like this, inspired by MultiPath which was designed by Waymo.





- Let $\vec{h} = [s_{t-H}, \dots s_{t-1}, s_t]$ be history of states & scene context
- Let $\vec{s} = [s_{t+1} \dots s_T]$ be prediction.



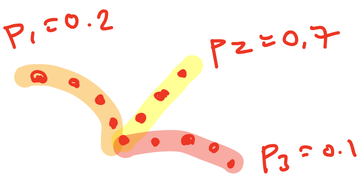
$\vec{a}^k = [a_1^k, a_2^k \dots]$

MultiPATH, WAYMO

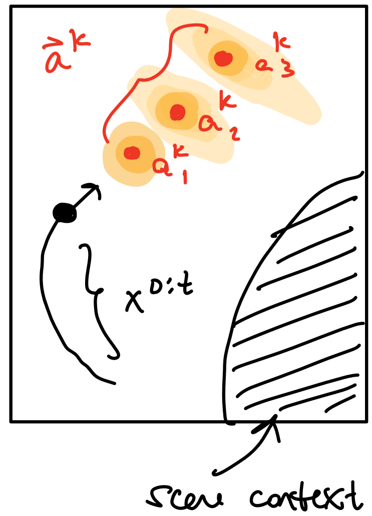
1) Discrete set of INTENTS: $\mathcal{A} = \{\vec{a}^k\}_{k=1}^{K}$ K = # intent trajs

2) Uncertainty over INTENT:

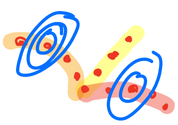
$$P(\vec{a}^k | \vec{h}) = \frac{e^{\phi_k(\vec{h})}}{\sum_i e^{\phi_i(\vec{h})}}$$



$\phi_k(\vec{h}) : \mathbb{R}^{d(\vec{h})} \mapsto \mathbb{R}$ is the output of a deep NN!



3) Uncertainty over state given intent:



$$P(s_t^k | \vec{a}^k, \vec{h}) = \mathcal{N}\left(s_t^k \,\middle|\, a_t^k + \underline{\mu_t^k(\vec{h})}, \underline{\Sigma_t^k(\vec{h})}\right)$$

predicted by model

4) Prediction:

$$p(\vec{s} | \vec{h}) = \sum_{k=1}^{K} \underbrace{P(\vec{a}^k | \vec{h})}_{\text{mixture weights fixed}} \prod_{t=1}^{T} P(s_t^k | \vec{a}^k, \vec{h})$$

GMM

For training, the MultiPath paper is trained via Imitation learning (i.e. maximize log-likelihood of recorded driving trajs).