Interaction Models

Ahmed Ahmed and Andrew Conkey

Using paired comparison data

Basic definitions

Given a set of objects $\{1, \ldots, n\}$, denote by Y_{ij} the binary random variable associated with the result of a paired comparison between i and j, taking value 1 if i is preferred to j and 0 otherwise.

Denote by π_{ij} the corresponding probability that i is preferred to j by a random subject.

Notional worths and choice probability

Many traditional paired preference models are formulated with the assumption that π_{ij} depends only on the difference between the "notional worths" (or utility values) of objects i and j.

That is, denoting these "notional worths" by a vector μ, we have

$$\pi_{ij} = F(\mu_i - \mu_j)$$

for some cumulative distribution function F of a zero-symmetric random variable.

Why might this make sense?

Suppose that, when prompted to make a comparison between objects, a subject's utility from each is given by their notional worths, up to a random error term. So

$$U_{si} = \mu_i + \delta_{si}$$
$$U_{sj} = \mu_j.$$

Assume that the δ_{sij} are i.i.d. Then

$$P\left(U_{si} \geq U_{sj}\right) = F\left(\mu_i - \mu_j\right)$$

where F is the c.d.f. of the distribution from which the δ_{sij} are drawn.

Bradley-Terry example

If we make our assumption that

$$\pi_{ij} = F(\mu_i - \mu_j),$$

and take F to be the c.d.f. of the logistic distribution centered at 0, we recover the Bradley-Terry model.

$$F(x) = \frac{1}{1 + e^{-x}}$$

$$\pi_{ij} = F(\mu_i - \mu_j) = \frac{1}{1 + e^{\mu_j - \mu_i}} = \frac{e^{\mu_i}}{e^{\mu_i} + e^{\mu_j}}.$$

Practical application

		1	X	2
London	Paris	186	26	91
London	Milan	221	26	56
Paris	Milan	121	32	59
London	St. Gallen	208	22	73
Paris	St. Gallen	165	19	119
Milan	St. Gallen	135	28	140
London	Barcelona	217	19	67
Paris	Barcelona	157	37	109
Milan	Barcelona	104	67	132
St. Gallen	Barcelona	144	25	134
London	Stockholm	250	19	34
Paris	Stockholm	203	30	70
Milan	Stockholm	157	46	100
St. Gallen	Stockholm	155	50	98
Barcelona	Stockholm	172	41	90

Estimation

One way to estimate μ is by MLE. Denote by n the number of subjects and by x_{ij} the number of responses where object i was preferred to object j.

Optimization problem:

$$\max \prod_{i<j} \left(\frac{e^{\mu_i}}{e^{\mu_i} + e^{\mu_j}} \right)^{x_{ij}} \left(\frac{e^{\mu_j}}{e^{\mu_i} + e^{\mu_j}} \right)^{n-x_{ij}}$$

$$\max \sum_{i<j} x_{ij}(\mu_i - \mu_j) + n(\mu_j - \log(e^{\mu_i}+e^{\mu_j}))$$

Results

	Thurstone			cumulative Thurstone		
	Est.	**S.E.**	**Q.S.E.**	**Est.**	**S.E.**	**Q.S.E.**
Barcelona	0.333	0.043	0.030	0.332	0.041	0.028
London	0.982	0.045	0.033	0.998	0.043	0.031
Milan	0.240	0.044	0.031	0.241	0.041	0.029
Paris	0.561	0.044	0.031	0.566	0.042	0.030
St. Gallen	0.325	0.043	0.030	0.324	0.040	0.028
Stockholm	0	–	0.031	0	–	0.029
τ_2	–	–	–	0.153	0.007	–

Alternative approach

- Instead of estimating separate worths for each object, assume some structural relationship between object attributes and worth
  - E.g., take $\mu_i = \beta z_i$, estimate β instead of μ.

	Est.	S.E.
Economics	0.757	0.066
Management	0.789	0.080
Latin country	−0.835	0.071
Discipline:Management	0.238	0.054
English:London	0.141	0.075
French:Paris	0.652	0.049
Italian:Milan	1.004	0.094
Spanish:Barcelona	0.831	0.095
τ_2	0.160	0.007

Quick Primer on RL

Notations

- $s \in \mathcal{S}$ = state/observation of the world (e.g. object and robot positions/pose)
- $a \in \mathcal{A}$ = actions taken by the agent (e.g. motor torques at low level, turn steering left/right, take route A vs route B to airport etc.)
- $P(s'|s,a)$ = dynamics of the world
- $r(s,a)$ = immediate reward for choosing action a in state s
- $\pi(a|s)$ = policy or decision making rule – tells us what to do in every state. The optimization problem of interest is find ($r_t \equiv r(s_t, a_t)$):

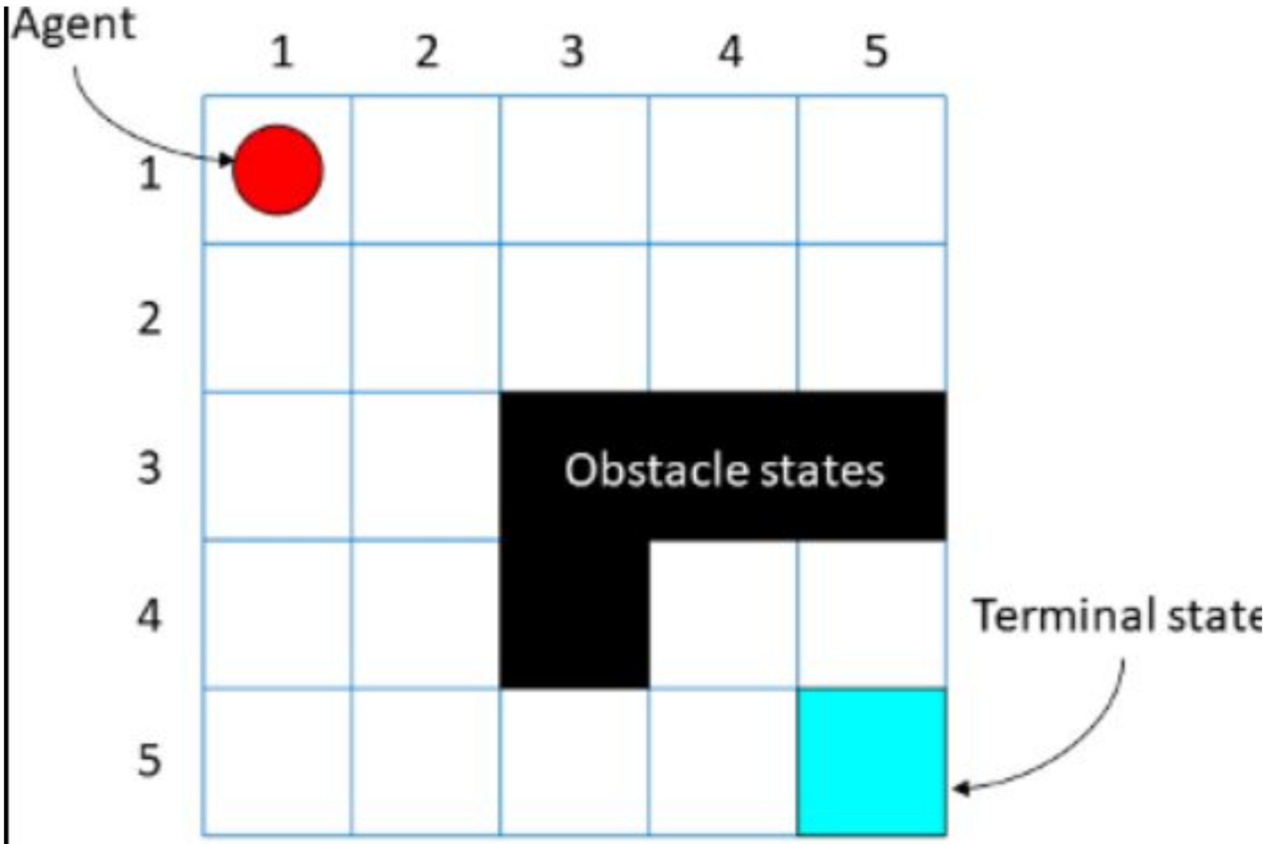
$$\pi^*(a|s) = \text{argmax}_\pi \mathbb{E}\left[r_0 + \gamma r_1 + \gamma^2 r_2 + \ldots\right]$$

Goal: find "near-optimal" policy $\pi^*(a|s)$ which maximizes the long term reward.

- $Q^\pi(s, a)$: a function that summarizes long term reward for choosing a in s. Future actions will be taken according to policy π.

$$Q^\pi(s, a) = \mathbb{E}_{a_t \sim \pi(.|s_t)} \left[r_0 + \gamma r_1 + \gamma^2 r_2 + \dots \mid s_0 = s, a_0 = a \right]$$

- $V^\pi(s) = \mathbb{E}_{a \sim \pi(.|s)} Q(s, a)$ summarizes how good a state is under current policy



Grid World

The IRL debate: how can we handle suboptimal demos?

[Amodei et al, 2017], [Krakovna, 2018]

Reward functions often have unintended consequences

[Russell, 1998], [Ng et al, 2000], [Abbeel and Ng, 2004]

We can use inverse reinforcement learning (IRL)!

<Too many papers to cite>

But humans are not optimal planners…

[Ziebart et al, 2008]

Let's model the human as **noisily** rational

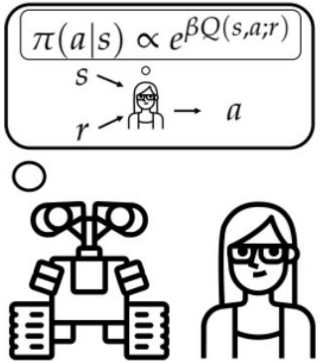
The IRL debate: how can we handle suboptimal demos?

[Christiano, 2015]

Then you are limited to human performance, since you don't know **how** the human made a mistake

[Ziebart et al, 2008]

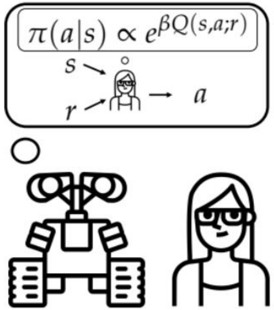
Let's model the human as **noisily** rational



[Evans et al, 2016], [Zheng et al, 2014], [Majumdar et al, 2017]

We can model human biases:

- Myopia
- Hyperbolic time discounting
- Sparse noise
- Risk sensitivity



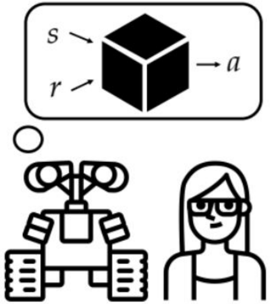
[Evans et al, 2016], [Zheng et al, 2014], [Majumdar et al, 2017]

We can model human biases:

- Myopia
- Hyperbolic time discounting
- Sparse noise
- Risk sensitivity

[Steinhardt and Evans, 2017]

Your human model will inevitably be misspecified



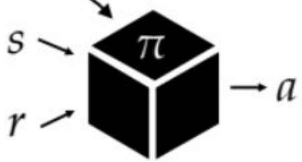
Hmm, maybe we can learn the **systematic** biases from data? Then we could correct for these biases during IRL

[Armstrong and Mindermann, 2017]

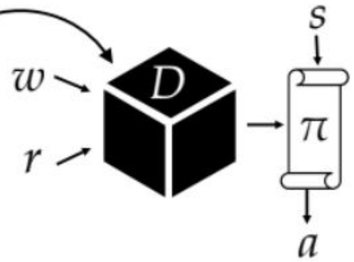
That's **impossible** without additional assumptions

Are minimal assumptions enough?

Learning a policy **isn't enough** to learn systematic biases



We need to learn the **planner** that produces the policy



Why learn the model?

If we knew $f(\mathbf{s}_t, \mathbf{a}_t) = \mathbf{s}_{t+1}$, we could use the tools from last week.

(or $p(\mathbf{s}_{t+1}|\mathbf{s}_t, \mathbf{a}_t)$ in the stochastic case)

So let's learn $f(\mathbf{s}_t, \mathbf{a}_t)$ from data, and *then* plan through it!

model-based reinforcement learning version 0.5:

1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. plan through $f(\mathbf{s}, \mathbf{a})$ to choose actions

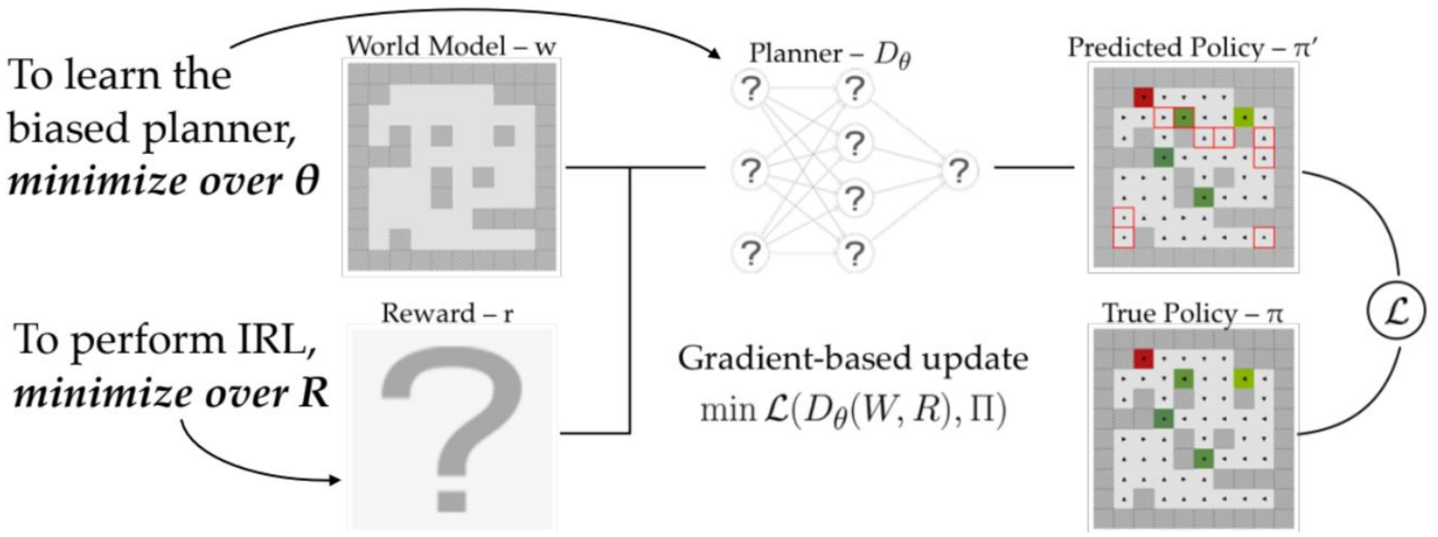
Can we do better?

model-based reinforcement learning version 1.5:

1. run base policy $\pi_0(\mathbf{a}_t|\mathbf{s}_t)$ (e.g., random policy) to collect $\mathcal{D} = \{(\mathbf{s}, \mathbf{a}, \mathbf{s}')_i\}$
2. learn dynamics model $f(\mathbf{s}, \mathbf{a})$ to minimize $\sum_i \|f(\mathbf{s}_i, \mathbf{a}_i) - \mathbf{s}'_i\|^2$
3. plan through $f(\mathbf{s}, \mathbf{a})$ to choose actions
4. execute the first planned action, observe resulting state $\mathbf{s}'$ (MPC)
5. append $(\mathbf{s}, \mathbf{a}, \mathbf{s}')$ to dataset $\mathcal{D}$

every N steps

This will be on HW



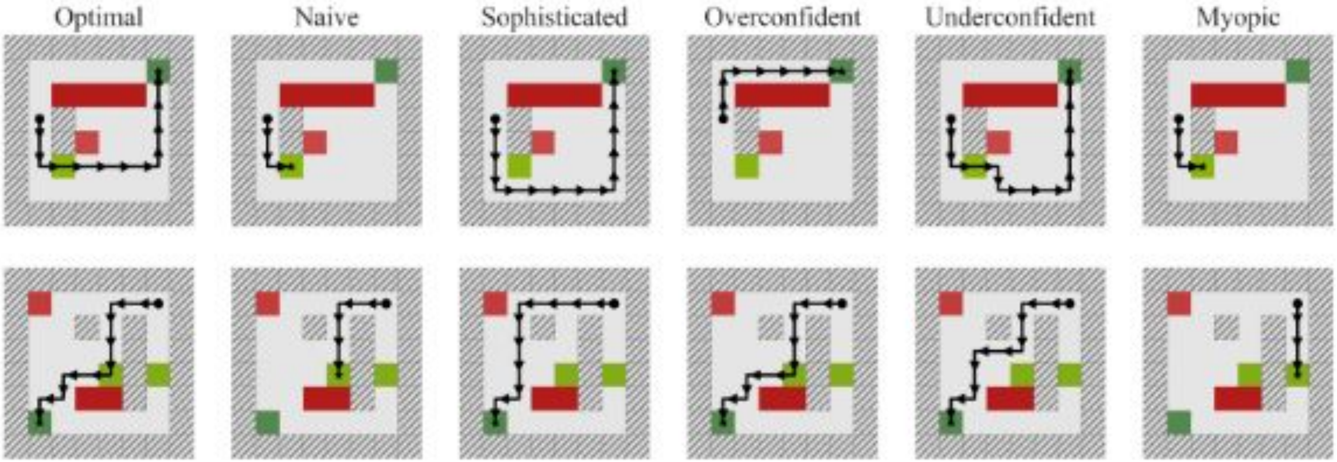
Algorithm 1: Some **known rewards**

1. On tasks with known rewards, learn the planner
2. Freeze the planner and learn the reward on remaining tasks

Algorithm 2: **"Near" optimal**

1. Use Algorithm 1 to mimic a simulated optimal agent
2. Finetune planner and reward jointly on human demonstrations

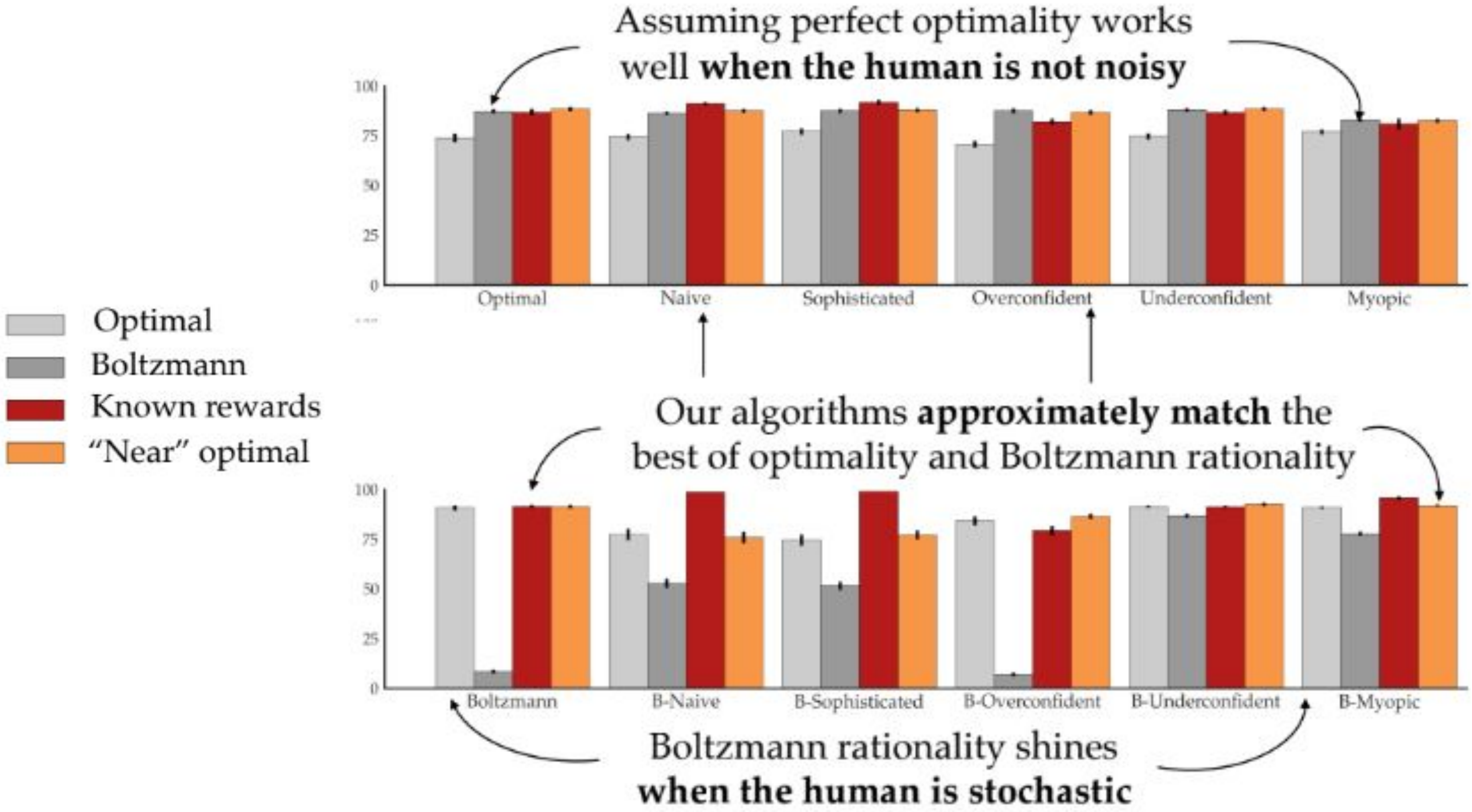
We created five **simulated human biases**, along with noisy variants:



Baselines: IRL using a **learned** optimal or Boltzmann human model.

For each algorithm (Optimal/Boltzmann/Alg 1/Alg 2) and bias, we:

1. Generate many environments and policies and run the algorithm
2. Optimize the inferred reward using value iteration to get a policy
3. Measure the policy's value, as a fraction of the optimal policy's value


Assuming perfect optimality works well **when the human is not noisy**
100
75
50
25
0
Optimal
Naive
Sophisticated
Overconfident
Underconfident
Myopic
Optimal
Boltzmann
Known rewards
"Near" optimal
Our algorithms **approximately match** the best of optimality and Boltzmann rationality
Boltzmann
B-Naive
B-Sophisticated
B-Overconfident
B-Underconfident
B-Myopic
Boltzmann rationality shines **when the human is stochastic**

What types of human feedback can we leverage?

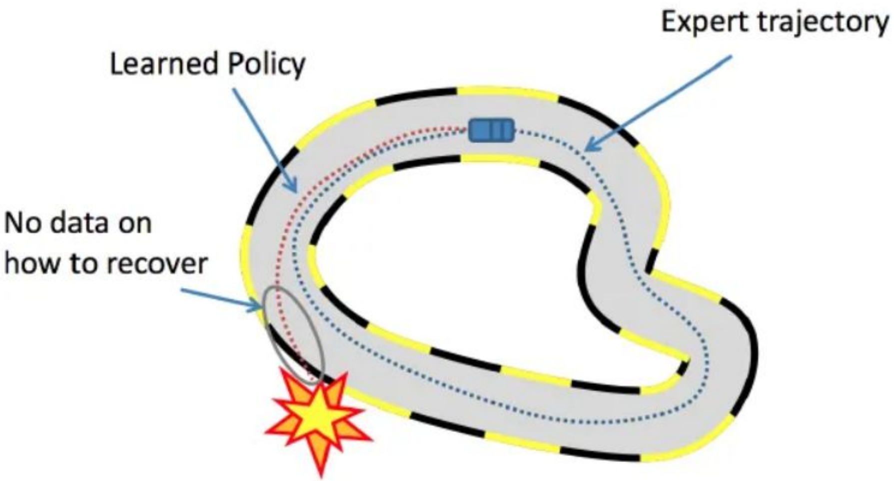
Preferences!

$$P(\xi_A \mid r, \beta) = \frac{\exp\left(\beta \cdot r(\xi_A)\right)}{\exp\left(\beta \cdot r(\xi_A)\right) + \exp\left(\beta \cdot r(\xi_B)\right)}$$

What types of human feedback can we leverage?

E-stops (counterfactual reasoning)

$$P(t \mid \xi, r, \beta) = \frac{\exp(\beta \cdot r(\xi_{0:t}))}{\sum_{k=0}^{T} \exp(\beta \cdot r(\xi_{0:k}))}.$$



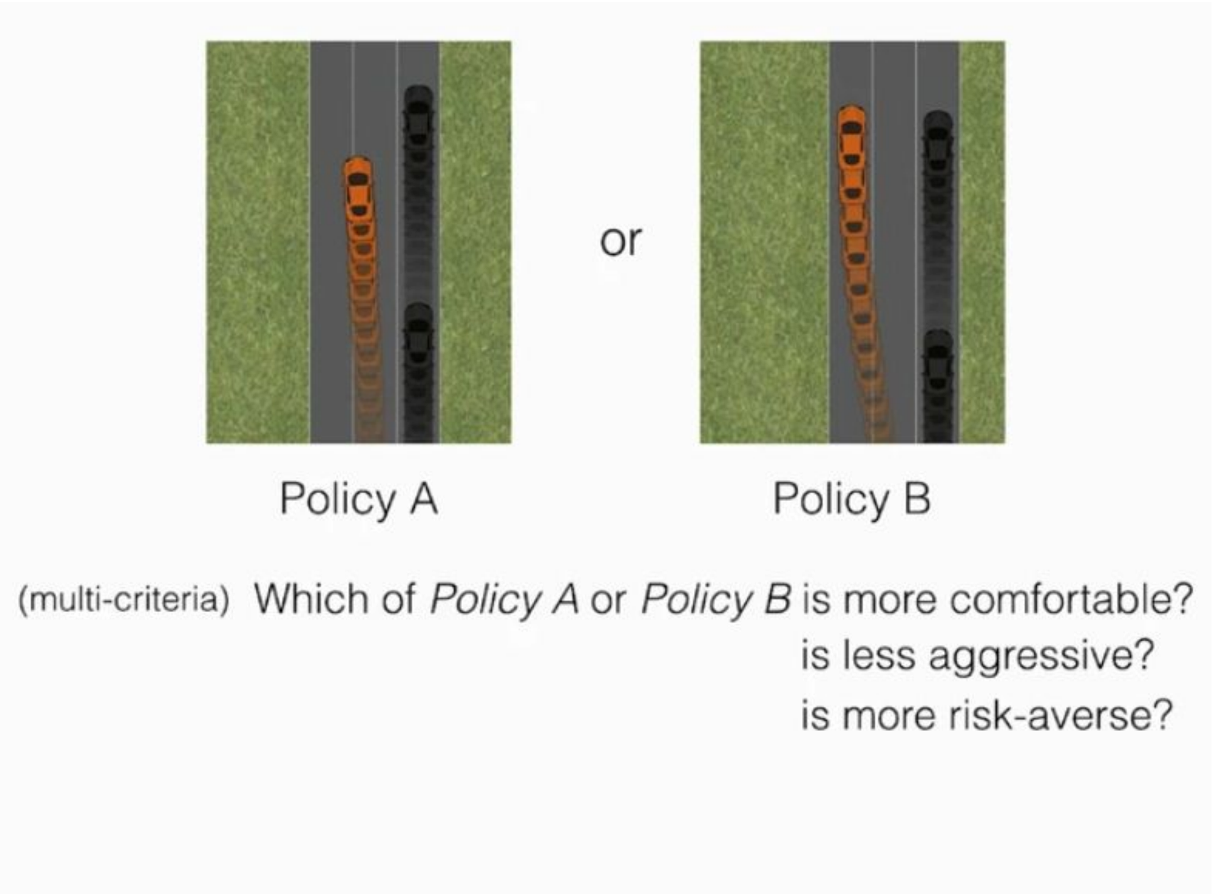
What types of human feedback can we leverage?

Demonstrations

$$
\begin{aligned}
P(\xi \mid r, \beta) &= \prod_{(s_t, a_t) \in \xi} \pi_\beta(a_t \mid s_t) \\
&= \prod_{(s_t, a_t) \in \xi} \frac{\exp(\beta Q_t^{\text{soft}}(s_t, a_t \mid r))}{\sum_{b \in \mathcal{A}} \exp(\beta Q_t^{\text{soft}}(s_t, b \mid r))}
\end{aligned} \tag{1}
$$

where $Q_t^{\text{soft}}(s, a \mid r) = r(s, a) + \gamma \mathbb{E}_{s'}[V_{t+1}^{\text{soft}}(s')]$, and $V_t^{\text{soft}}(s) = \mathbb{E}_{a \sim \pi_\beta}[Q_t^{\text{soft}}(s, a) - \log \pi_\beta(a \mid s)]$ are the soft Q-function, and Value function, respectively (Kitani et al. 2012; Haarnoja et al. 2017), and π_β is the corresponding (time-dependent) policy.

What about when we have multiple preference criteria?

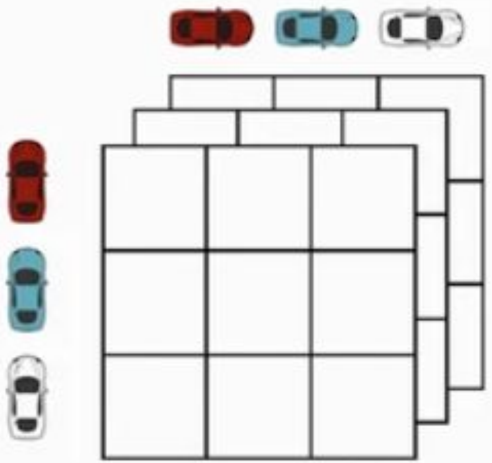


What about when we have multiple preference criteria?

Complex real-world problems are multi-criteria.

Uni-criterion framework are insufficient to model these complexities

Multi-criteria Preference Learning



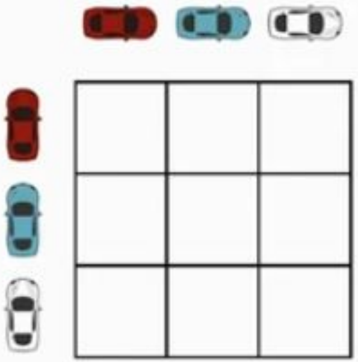
$\mathbf{P} \in [0,1]^{d \times d \times k}$

Preference Tensor

$$\mathbf{P}(i_1, i_2; j) = \text{Prob}(\text{Pol } i_1 \succeq \text{Pol } i_2 \text{ along criteria } j)$$

Objective: Given such pairwise comparisons, which is the **best policy**?

Multi-criteria Preference Learning



$\mathbf{P} \in [0,1]^{d \times d}$

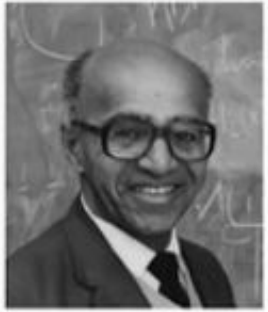
Preference Matrix

$$\mathbf{P}(i_1, i_2; j) = \text{Prob}(\text{Pol } i_1 \succeq \text{Pol } i_2 \text{ along criteria } j)$$

Objective: Given such pairwise comparisons, which is the **best policy**?

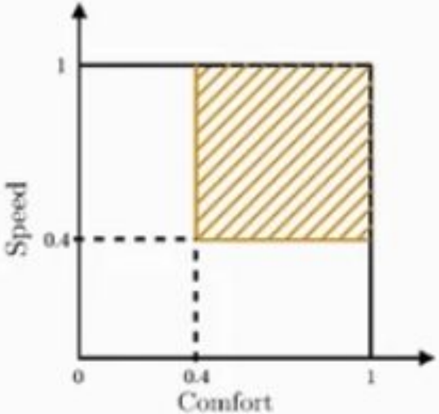
von Neumann winner (uni-criterion setup)

A randomized policy which is preferred over every other policy by more than 50% of population



David Blackwell

What is a natural generalization of von Neumann's minimax theorem for vector-valued zero-sum games?



Proposed notion of *Target Set*

Blackwell Winner: Randomized policy which “best” trades-off the criteria according to user-specified *target sets*.

References

- CS 285 Deep Reinforcement Learning - Sergey Levine: https://rail.eecs.berkeley.edu/deeprlcourse/
- Cattelan, Manuela. "Models for paired comparison data: A review with emphasis on dependent data." (2012): 412-433. https://arxiv.org/abs/1210.1016
- Bhatia, Kush, Ashwin Pananjady, Peter Bartlett, Anca Dragan, and Martin J. Wainwright. "Preference learning along multiple criteria: A game-theoretic perspective." Advances in neural information processing systems 33 (2020): 7413-7424. https://proceedings.neurips.cc/paper/2020/hash/52f4691a4de70b3c441bca6c546979d9-Abstract.html
- Shah, Rohin, Noah Gundotra, Pieter Abbeel, and Anca Dragan. "On the feasibility of learning, rather than assuming, human biases for reward inference." In International Conference on Machine Learning, pp. 5670-5679. PMLR, 2019. https://arxiv.org/abs/1906.09624
- Ghosal, Gaurav R., Matthew Zurek, Daniel S. Brown, and Anca D. Dragan. "The effect of modeling human rationality level on learning rewards from multiple feedback types." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, no. 5, pp. 5983-5992. 2023. https://ojs.aaai.org/index.php/AAAI/article/view/25740
-