

# CS 277: Control and Reinforcement Learning

## Winter 2026

# Lecture 19: Open Questions

**Roy Fox**

Department of Computer Science

School of Information and Computer Sciences

University of California, Irvine



# Logistics

---

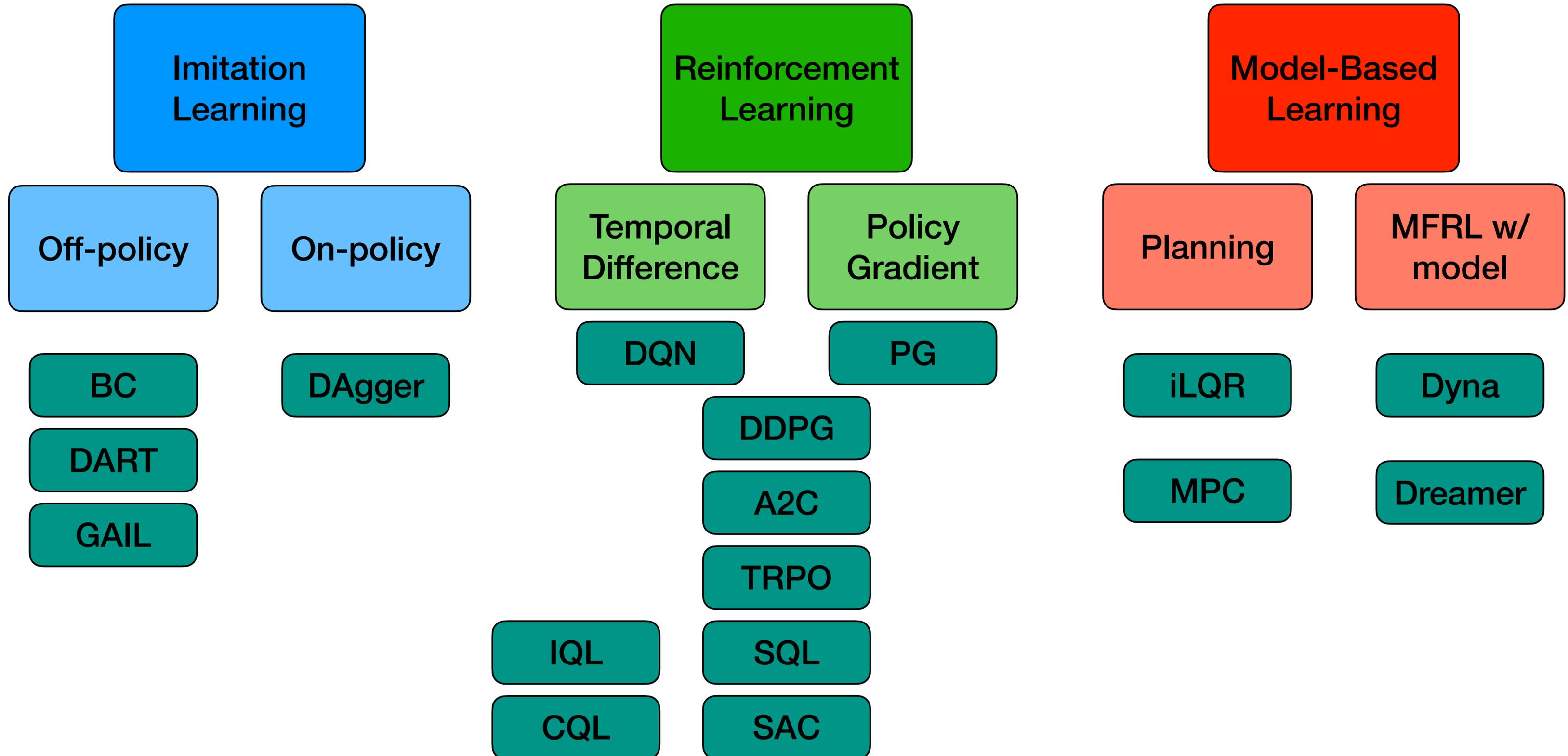
evaluations

- Course evaluations due **next Monday**

assignments

- Exercise 5 due **next Tuesday**

# Taxonomy

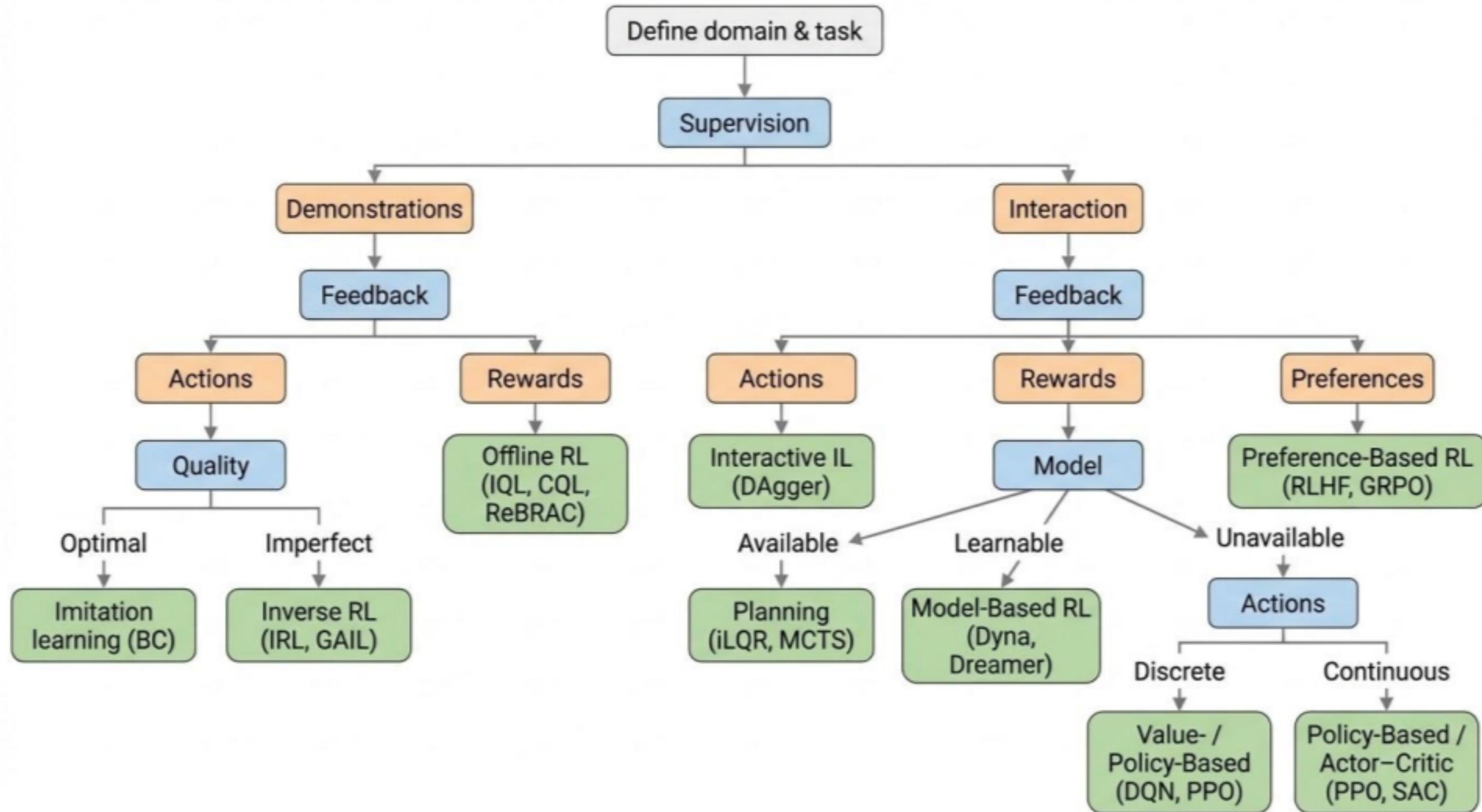


# Why so many algorithms?

---

- We may have different **modeling assumptions**
  - Is the environment **stochastic** or **deterministic**?
  - Is the state / action space **continuous** or **discrete**?
  - Is the horizon **episodic** or **infinite**?
- We may care about different **tradeoffs**
  - **Sample** efficiency? **Computational** efficiency while learning / executing? Succinct **representation**?
  - Algorithmic **stability**, **reproducibility**, ease of **use** (existing code), ease of **adaptation**
- Different difficulty to **represent or learn** in different domains
  - Represent / learn a **policy** or a **model**?
  - Discover **structure**? **Memory**? Transfer / share with **other tasks**?

# Flowchart: which approach to choose?



# On- or off-policy data?

---

- The faster our **simulator**  $\Rightarrow$  the faster we can **refresh** our data
  - And still keep sufficient **diversity** for training
- Fast enough  $\Rightarrow$  can use **on-policy** data
  - No need for **replay buffer**
  - No train $\rightarrow$ test distributional mismatch (= **covariate shift**)
  - Can still use off-policy **algorithms** with on-policy data
- Extremely slow simulator  $\Rightarrow$  not even off-policy, just **offline RL**

# Topics we covered

---

- Imitation learning
- Policy evaluation + improvement
  - Monte-Carlo vs. Temporal Difference
  - On- vs. off-policy
- Policy Gradient
  - Advantage estimation, Actor–Critic
- Exploration
- Partial observability
- Optimal control
- Planning and model-based RL
- Inverse RL
- Bounded RL
- Offline RL
- Multi-agent RL
- Structured control
- Foundation models

# Topics we didn't cover

---

- Hindsight Experience Replay (HER)
- Eligibility traces
- Generalized Value Functions (GVF)
  - Successor representation
- Value Iteration / Prediction Nets (VIN / VPN)
- Natural policy gradient
  - Mirror descent
- Distributional RL
- Bayesian RL
- Hyperparameter tuning
- Distributed RL
- Robot learning
- Safety
- Curiosity / diversity / empowerment
- Preference elicitation
- 3rd-person imitation
- Meta-learning
- Lifelong learning

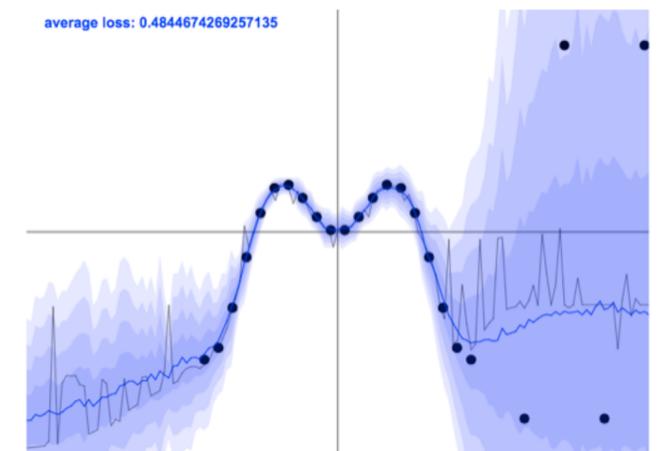
# Trends and open questions in ML

---

- Bayesian Deep Learning
- Optimization theory
- Neuro-symbolic AI
- Meta-learning / learning to learn
- Lifelong learning
- Causality
- Interpretability, explainability
- AI ethics: fairness, debiasing, alignment

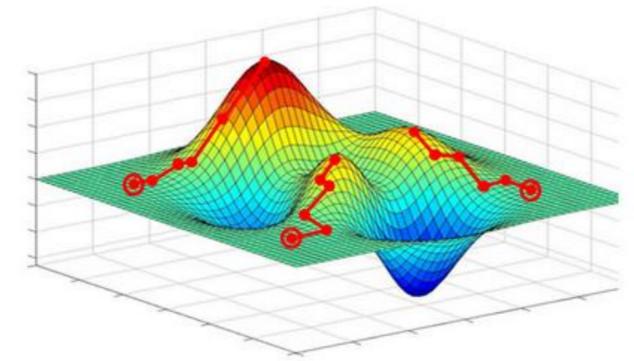
# Bayesian RL

- Two kinds of **uncertainty**
  - **Aleatoric** = things I haven't seen / haven't happened yet:  $p(s_t | m_t)$ ,  $p(r_{t+k} | m_t)$ , ...
  - **Epistemic** (= model uncertainty) = things I haven't modeled / learned yet:  $\hat{p}$ ,  $\pi_\theta$ ,  $Q_\phi$ , ...
- Standard RL already considers aleatoric uncertainty
  - “Overtake truck quickly, to reduce time with partial observability, probability of crash”
- Bayesian RL can estimate **epistemic uncertainty**:  $p(\theta | \mathcal{D})$ 
  - Can help improve **exploration** (cf. Thompson sampling)
  - Can improve learning in **bounded** agents and **offline** RL



# Optimization $\Leftrightarrow$ RL

- Special considerations of optimization  $\rightarrow$  RL:
  - ▶ Covariate shift
  - ▶ Temporal-Difference  $\Rightarrow$  non-stationary loss landscape
  - ▶ Saddle points in multi-agent RL
- RL  $\rightarrow$  optimization: iterative optimization is a dynamical process
  - ▶ Gradient descent = maximize “reward” of descending loss landscape
  - ▶ Optimal control concepts (e.g. Langevin dynamics) key in analysis



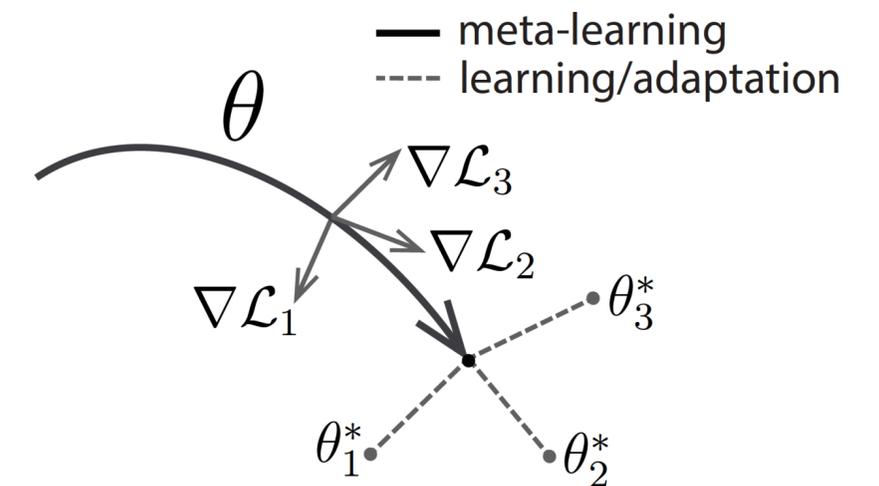
# Neuro-symbolic RL

- Is there any benefit to **discrete** components in gradient-based methods?
  - E.g. **modularity**
- **Structured control** = discrete memory components
  - Can help sample efficiency, generalization, transfer, interpretability, ...
- How to **learn** under given structure?
- How to **discover** useful structure?



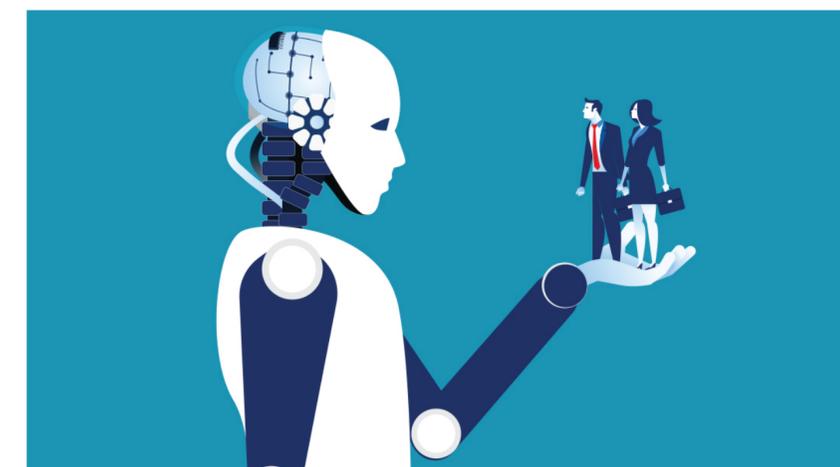
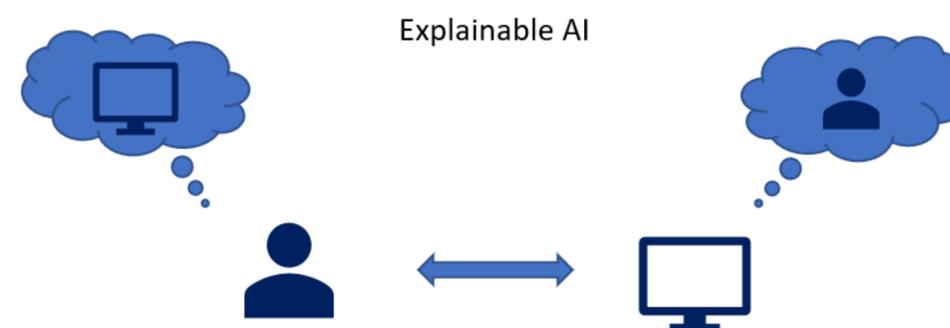
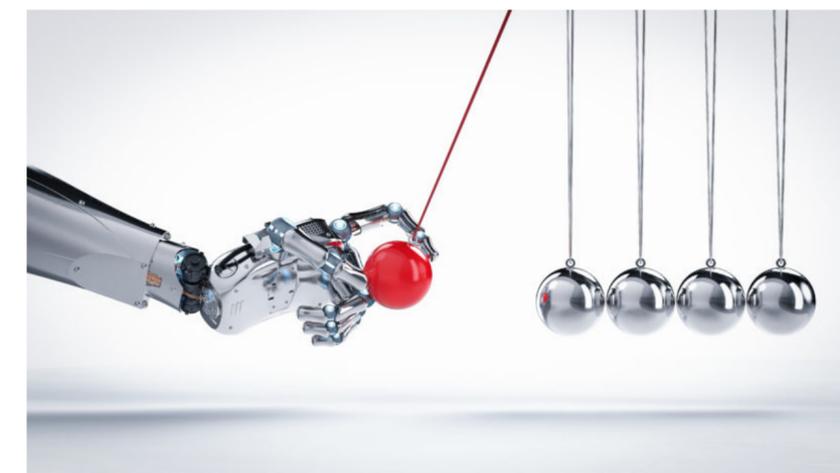
# Meta-learning $\Leftrightarrow$ RL

- **Multi-task learning** = transfer / share learning products between tasks
  - E.g. features, models, policies, skills
- **Meta-learning** = transfer / share learning of learner components
  - Network architecture = **Neural Architecture Search (NAS)**
  - **Optimizer** hyperparameters
  - Parameter **initializations** (MAML)
- Learning to perform **sequence of tasks** = sequential decision making
  - E.g. can use RNNs, Transformers



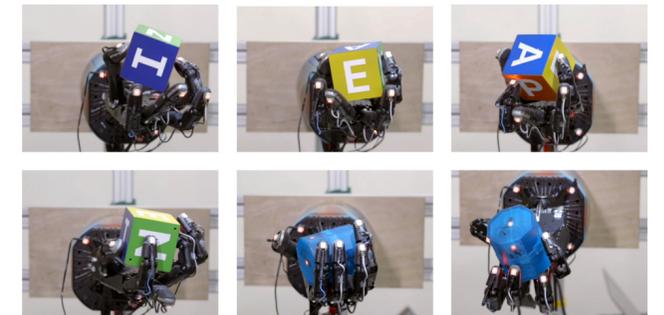
# Trends and open questions in ML

- Bayesian Deep Learning
- Optimization theory
- Neuro-symbolic AI
- Meta-learning / learning to learn
- Lifelong learning
- Causality
- Interpretability, explainability
- AI ethics: fairness, debiasing, alignment



# Reliability and reproducibility

- Reinforcement learning has seen immense **success**
  - But remains largely **irreproducible**, hard to **deploy**
- Many algorithms are very **sensitive to hyperparameters**
- Very sensitive to **parameter initialization**
  - Need to evaluate over **many runs**, prone to **cherry-picking**
- Small **implementation details** may have unexpected effects
- How to go beyond this **pre-paradigmatic** phase?
  - Better **RL theory**
  - Build practical RL (and ML) as **experimental** field (e.g. reliable)



# Other open questions

---

- Bounded RL
  - How “bounded” should the agent be?
- Structured control
  - How to discover structure for control / memory / reasoning?
  - Which structures are useful for multi-task control?
- World modeling
  - How to transfer large world models to new tasks?
- Multi-task learning
  - How to discover which tasks are related / unrelated?

# Questions?

