CS394R
Reinforcement Learning: Theory and Practice

Scott Niekum and Peter Stone

Department of Computer Science
The University of Texas at Austin

Good Morning Colleagues

- Are there any questions?

Logistics

- Feedback on final project proposals given

Logistics

- Feedback on final project proposals given
- Midterm results

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online
- Next step: literature surveys

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online
- Next step: literature surveys
  - Build on proposal

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online
- Next step: literature surveys
  - Build on proposal
- Next week's readings

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online
- Next step: literature surveys
  - Build on proposal
- Next week's readings
  - Exploration and intrinsic motivation

Logistics

- Feedback on final project proposals given
- Midterm results
- Forum for AI talk on GT Sophy - online
- Next step: literature surveys
  - Build on proposal
- Next week's readings
  - Exploration and intrinsic motivation
  - No longer a textbook

Abstraction

- What are the two types?

Abstraction

- What are the two types?
  - State abstraction
  - Temporal abstraction
- Week 0 task

Options

- Extension of RL to temporal abstraction

Options

- Extension of RL to temporal abstraction
- They don't address **what** temporal abstraction to use — they just show how it can fit into the RL formalism

Options

- Extension of RL to temporal abstraction
- They don't address **what** temporal abstraction to use — they just show how it can fit into the RL formalism
  - Why couldn't it before?

Options

- Extension of RL to temporal abstraction
- They don't address **what** temporal abstraction to use — they just show how it can fit into the RL formalism
  - Why couldn't it before?
- Markov vs. Semi-markov:
  - states, actions
  - mapping from (s, a) to expected discounted reward
  - well-defined distribution of next state, transit time

Options

- Extension of RL to temporal abstraction
- They don't address **what** temporal abstraction to use — they just show how it can fit into the RL formalism
  - Why couldn't it before?
- Markov vs. Semi-markov:
  - states, actions
  - mapping from (s, a) to expected discounted reward
  - well-defined distribution of next state, transit time
- Options can be detrimental without good state abstractions (slides)

Common Questions

- Please discuss intra-option learning more

Common Questions

- Please discuss intra-option learning more
- What are the current challenges in abstraction? (From chapter 16 it doesnt look like people have widely adapted it.)
- What techniques exist to automate the abstraction selection process (discovery)?
  - bottleneck states
  - novelty
  - changed useful state abstractions (slides)

Other Common Questions

- How do you determine the proper degree of abstraction?
  - What is maximum allowable abstraction before it hurts agent? Can we quantify it?

Other Common Questions

- How do you determine the proper degree of abstraction?
  - What is maximum allowable abstraction before it hurts agent? Can we quantify it?
  - Global vs. hierarchical vs. recursive optimality (slides)

Other Common Questions

- How do you determine the proper degree of abstraction?
  - What is maximum allowable abstraction before it hurts agent? Can we quantify it?
  - Global vs. hierarchical vs. recursive optimality (slides)
- How is transfer learning typically performed in RL? (slides)
- What do positive and negative transfer mean?

Other Interesting Questions

- Daniel Almeraz: With infinite resources and time, would abstraction hinder an agent?

Other Interesting Questions

- Daniel Almeraz: With infinite resources and time, would abstraction hinder an agent?
- Oguzhan Akcin: Are function approximation methods(i.e., neural networks) a form of abstraction?
- Shwetha Ramachandran: What's the difference between state abstraction and state aggregation?

Other Interesting Questions

- Rishikumar Salem: Are most abstractions understandable to people, or are they often selected in a black-box manner by a machine learning model?

Other Interesting Questions

- Rishikumar Salem: Are most abstractions understandable to people, or are they often selected in a black-box manner by a machine learning model?
- Haroon Mushtaq: Can options be used to make RL systems safe?

Discussion Points

- What happens when initial value functions are optimistic? (slides)

Discussion Points

- What happens when initial value functions are optimistic? (slides)

MAXQ

- Defines how to learn given a task hierarchically

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies
  - Weaker or stronger than hierarchical optimality?

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies
  - Weaker or stronger than hierarchical optimality?
- Enables reuse of subtasks

MAXQ

- Defines how to learn given a task hierarchically
- Does not address how to construct the hierarchy
- Strives for **recursive optimality**— local optimality given subtask policies
  - Weaker or stronger than hierarchical optimality?
- Enables reuse of subtasks
- Enables useful state abstraction (how?)

Some details

- a means both primitive actions and subtasks (options)

Some details

- a means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent

Some details

- a means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly

Some details

- a means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
- What does $C_i^\pi(s, a)$ mean?

Some details

- a means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
- What does $C_i^\pi(s, a)$ mean? (Nick slides)

Some details

- a means both primitive actions and subtasks (options)
- Context-dependent vs. context-independent
- Higher-level subtasks are essentially policies over options
  - But subtasks are learned too
  - And the values propagate correctly
- What does $C_i^\pi(s, a)$ mean? (Nick slides)

Discussion Points

- What does MAXQ-Q buy you over flat?

Discussion Points

- What does MAXQ-Q buy you over flat?
- What does polling buy you over flat?