CS394R
Reinforcement Learning: Theory and Practice

Peter Stone

Department of Computer Science
The University of Texas at Austin

Good Morning Colleagues

- Are there any questions?

Logistics

- I respond preferentially to responses that are on time

Logistics

- I respond preferentially to responses that are on time
- Use piazza more
- Do your programming assignments

Logistics

- I respond preferentially to responses that are on time
- Use piazza more
- Do your programming assignments
- Next week's readings: bandits (led by Sam)

Logistics

- I respond preferentially to responses that are on time
- Use piazza more
- Do your programming assignments
- Next week's readings: bandits (led by Sam)
  - I'll be out of town all week, but on email

Logistics

- I respond preferentially to responses that are on time
- Use piazza more
- Do your programming assignments
- Next week's readings: bandits (led by Sam)
  - I'll be out of town all week, but on email
- Email your project proposals before class

Logistics

- I respond preferentially to responses that are on time
- Use piazza more
- Do your programming assignments
- Next week's readings: bandits (led by Sam)
  - I'll be out of town all week, but on email
- Email your project proposals before class
  - Think about how to *model* the domain

Overview

- RMax: model-based learning in polynomial time

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space
  - Factored state space (pdf)
  - Bayes Nets, DBNs, CPTs (ppt)

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space
  - Factored state space (pdf)
  - Bayes Nets, DBNs, CPTs (ppt)
  - Structure learning

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space
  - Factored state space (pdf)
  - Bayes Nets, DBNs, CPTs (ppt)
  - Structure learning
- Rmax (and SLF-Rmax) not built to be practical

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space
  - Factored state space (pdf)
  - Bayes Nets, DBNs, CPTs (ppt)
  - Structure learning
- Rmax (and SLF-Rmax) not built to be practical
  - Built to be provably convergent

Overview

- RMax: model-based learning in polynomial time
  - High-level idea (pdf)
  - Q-learning vs. RMax (videos)
- Met-RMax: Exploiting a structured state space
  - Factored state space (pdf)
  - Bayes Nets, DBNs, CPTs (ppt)
  - Structure learning
- Rmax (and SLF-Rmax) not built to be practical
  - Built to be provably convergent
- Fitted R-Max: Extend to continuous state space (pdf)

Discussion

- What's more interesting? Theoretically grounded algorithms? Or algorithms that work in practice?