CS343
Artificial Intelligence

Prof: Peter Stone

Department of Computer Science
The University of Texas at Austin
Good Afternoon, Colleagues
Good Afternoon, Colleagues

Are there any questions?
Logistics

- Tournament qualification underway
Logistics

- Tournament qualification underway
- Tracking assignment due in two days
- Classification assignment to be assigned on Thursday
Some Context

- **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
Some Context

- **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
Some Context

- **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
Some Context

- **First weeks**: search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**
Some Context

• **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**

• **Next:** MDPs —
Some Context

- **First weeks**: search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**

- **Next**: MDPs — towards reinforcement learning
Some Context

• **First weeks:** search (BFS, A*, minimax, alpha-beta)
  – Find an optimal plan (or solution)
  – Best thing to do from the current state
  – Know transition and cost (reward) functions
  – Either execute complete solution (deterministic) or search again at every step
  – **Know current state**

• **Next:** MDPs — towards reinforcement learning
  – Still know transition and reward function
Some Context

- **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**
- **Next:** MDPs — towards reinforcement learning
  - Still know transition and reward function
  - Looking for a **policy** — optimal action from every state
Some Context

- **First weeks**: search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**

- **Next**: MDPs — towards reinforcement learning
  - Still know transition and reward function
  - Looking for a **policy** — optimal action from every state

- **Before Midterm**: Reinforcement learning
Some Context

- **First weeks:** search (BFS, A*, minimax, alpha-beta)
  - Find an optimal plan (or solution)
  - Best thing to do from the current state
  - Know transition and cost (reward) functions
  - Either execute complete solution (deterministic) or search again at every step
  - **Know current state**

- **Next:** MDPs — towards reinforcement learning
  - Still know transition and reward function
  - Looking for a **policy** — optimal action from every state

- **Before Midterm:** Reinforcement learning
  - Policy without knowing transition or reward functions
Some Context

• **First weeks:** search (BFS, A*, minimax, alpha-beta)
  – Find an optimal plan (or solution)
  – Best thing to do from the current state
  – Know transition and cost (reward) functions
  – Either execute complete solution (deterministic) or search again at every step
  – **Know current state**

• **Next:** MDPs — towards reinforcement learning
  – Still know transition and reward function
  – Looking for a **policy** — optimal action from every state

• **Before Midterm:** Reinforcement learning
  – Policy without knowing transition or reward functions
  – **Still know state**
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference
- Prior, net structure, and CPT’s known
Some Context (cont.)

- **Probabilistic Reasoning**: Now state is unknown
- Bayesian networks – state estimation/inference
- **Prior, net structure, and CPT’s known**
  - **Week 5**: Probability and utilities
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference
- **Prior, net structure, and CPT’s known**
  - **Week 5:** Probability and utilities
  - **Week 9:** Conditional independence and inference (exact and approximate)
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference
- **Prior, net structure, and CPT’s known**
  - **Week 5:** Probability and utilities
  - **Week 9:** Conditional independence and inference (exact and approximate)
  - **Week 10:** Exact state estimation over time
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference
- **Prior, net structure, and CPT’s known**
  - **Week 5:** Probability and utilities
  - **Week 9:** Conditional independence and inference (exact and approximate)
  - **Week 10:** Exact state estimation over time
  - **Week 11:** Approximate state estimation over time
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown

- Bayesian networks – state estimation/inference

- **Prior, net structure, and CPT’s known**
  - **Week 5:** Probability and utilities
  - **Week 9:** Conditional independence and inference (exact and approximate)
  - **Week 10:** Exact state estimation over time
  - **Week 11:** Approximate state estimation over time

- **This week:** What if they’re not known?
Some Context (cont.)

• Probabilistic Reasoning: Now state is unknown
• Bayesian networks – state estimation/inference

• Prior, net structure, and CPT’s known
  – Week 5: Probability and utilities
  – Week 9: Conditional independence and inference (exact and approximate)
  – Week 10: Exact state estimation over time
  – Week 11: Approximate state estimation over time

• This week: What if they’re not known?
  – Also Bayesian networks for classification
Some Context (cont.)

- **Probabilistic Reasoning:** Now state is unknown
- Bayesian networks – state estimation/inference

- **Prior, net structure, and CPT’s known**
  - **Week 5:** Probability and utilities
  - **Week 9:** Conditional independence and inference (exact and approximate)
  - **Week 10:** Exact state estimation over time
  - **Week 11:** Approximate state estimation over time

- **This week:** What if they’re not known?
  - Also Bayesian networks for **classification**
  - A type of **machine learning**
Some Context (cont.)

- **Next week**: Machine Learning
  
  - Just a taste – focus on concept learning = classification
Some Context (cont.)

• **Next week:** Machine Learning
  – Just a taste – focus on concept learning = classification

• **Week 14:** Classical planning
  – Reasoning with first order representations
Some Context (cont.)

- **Next week:** Machine Learning
  - Just a taste – focus on concept learning = classification

- **Week 14:** Classical planning
  - Reasoning with first order representations
  - So far we’ve dealt with propositions
Some Context (cont.)

- **Next week:** Machine Learning
  - Just a taste – focus on concept learning = classification

- **Week 14:** Classical planning
  - Reasoning with first order representations
  - So far we’ve dealt with propositions
  - Back to known transitions, known state, etc.
Some Context (cont.)

- **Next week:** Machine Learning
  - Just a taste – focus on concept learning = classification

- **Week 14:** Classical planning
  - Reasoning with first order representations
  - So far we’ve dealt with propositions
  - Back to known transitions, known state, etc.

- **Week 15:** Philosophical foundations and ethics
Some Context (cont.)

• **Next week:** Machine Learning
  – Just a taste – focus on concept learning = classification

• **Week 14:** Classical planning
  – Reasoning with first order representations
  – So far we’ve dealt with propositions
  – Back to known transitions, known state, etc.

• **Week 15:** Philosophical foundations and ethics

  It’s all about building agents

  Sense, decide, act
Some Context (cont.)

- **Next week:** Machine Learning
  - Just a taste – focus on concept learning = classification

- **Week 14:** Classical planning
  - Reasoning with first order representations
  - So far we’ve dealt with propositions
  - Back to known transitions, known state, etc.

- **Week 15:** Philosophical foundations and ethics

  It’s all about building agents

  Sense, decide, act  Maximize expected utility

Peter Stone
Topics not covered

- Constraint satisfaction (Chapter 6)
- Knowledge representation and reasoning (Chapters 7-9, 11, 12)
- Game theory and auctions (Sections 17.5, 17.6)
- Aspects of learning (Chapters 18, 19)
- Natural language (Chapters 22, 23)
- Vision (Chapter 24)
- Robotics (Chapter 25)