CS394R
Reinforcement Learning:
Theory and Practice

Peter Stone

Department of Computer Science
The University of Texas at Austin

Good Morning Colleagues

Good Morning Colleagues

- Are there any questions?

Logistics

- Reading responses

Logistics

- Reading responses
- Next week's readings

Logistics

- Reading responses
- Next week's readings
- The math is important

Logistics

- Reading responses
- Next week's readings
- The math is important
- Use piazza

Chapter 3

- Defines the problem

Chapter 3

- Defines the problem
- Introduces some important notation and concepts.

Chapter 3

- Defines the problem
- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations

Chapter 3

- Defines the problem
- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations
  - Get comfortable with them!

Chapter 3

- Defines the problem
- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations
  - Get comfortable with them!
    - $q_\pi(s,a) =$

Chapter 3

- Defines the problem
- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations
  - Get comfortable with them!
    - $q_\pi(s, a) =$
    - Backup diagrams (p. 62)

Chapter 3

- Defines the problem

- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations
  - Get comfortable with them!
    - $q_\pi(s,a) =$
    - Backup diagrams (p. 62)

- Solution methods start in Chapter 4

Chapter 3

- Defines the problem

- Introduces some important notation and concepts.
  - Returns
  - Markov property
  - State/action value functions
  - Bellman equations
  - Get comfortable with them!
    - $q_\pi(s, a) =$
    - Backup diagrams (p. 62)

- Solution methods start in Chapter 4
  - What does it mean to **solve** an RL problem?

Formulating the RL problem

- Art more than science
- States, actions, rewards
- Rewards: no hints on **how** to solve the problem

Formulating the RL problem

- Art more than science
- States, actions, rewards
- Rewards: no hints on **how** to solve the problem
- Discounted vs. non-discounted

Formulating the RL problem

- Art more than science
- States, actions, rewards
- Rewards: no hints on **how** to solve the problem
- Discounted vs. non-discounted
- Episodic vs. continuing

Value functions

- Consider the week 0 environment

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case
- Construct Q in undiscounted, episodic case

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case
- Construct Q in undiscounted, episodic case
- What if it's discounted?

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case
- Construct Q in undiscounted, episodic case
- What if it's discounted?
- What if it's continuing?

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case
- Construct Q in undiscounted, episodic case
- What if it's discounted?
- What if it's continuing?
- Continuing tasks without discounting?

Value functions

- Consider the week 0 environment
- For some s, what is $V(s)$?
- OK - consider the policy we ended with
- Now, for some s, what is $V(s)$?
- Construct V in undiscounted, episodic case
- Construct Q in undiscounted, episodic case
- What if it's discounted?
- What if it's continuing?
- Continuing tasks without discounting?
- Exercises 3.9, 3.10, 3.16

Markov property

- What is it?

Markov property

- What is it?
- Does it hold in the real world?

Markov property

- What is it?
- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?

Markov property

- What is it?

- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?
  - What if there's a time horizon?

Markov property

- What is it?
- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?
  - What if there's a time horizon?
- It's an ideal
  - Will allow us to prove properties of algorithms

Markov property

- What is it?

- Does it hold in the real world?

  – Are any systems "fundamentally" non-Markovian?
  – What if there's a time horizon?

- It's an ideal

  – Will allow us to prove properties of algorithms
  – Algorithms may still work when not provably correct

Markov property

- What is it?
- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?
  - What if there's a time horizon?
- It's an ideal
  - Will allow us to prove properties of algorithms
  - Algorithms may still work when not provably correct
  - Could you compensate? Do algorithms change?

Markov property

- What is it?
- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?
  - What if there's a time horizon?
- It's an ideal
  - Will allow us to prove properties of algorithms
  - Algorithms may still work when not provably correct
  - Could you compensate? Do algorithms change?
  - If not, you may want different algorithms (Monte Carlo)

Markov property

- What is it?
- Does it hold in the real world?
  - Are any systems "fundamentally" non-Markovian?
  - What if there's a time horizon?
- It's an ideal
  - Will allow us to prove properties of algorithms
  - Algorithms may still work when not provably correct
  - Could you compensate? Do algorithms change?
  - If not, you may want different algorithms (Monte Carlo)
- Exercise 3.6 (broken vision system)

Chapter 4

- Solution methods **given a model**

Chapter 4

- Solution methods **given a model**
  - So no exploration vs. exploitation

Chapter 4

- Solution methods **given a model**
  - So no exploration vs. exploitation

Policy Evaluation

- V^π exists and is unique if $\gamma < 1$ or termination guaranteed for all states under policy π.

Policy Evaluation

- V^π exists and is unique if $\gamma < 1$ or termination guaranteed for all states under policy π.
- Policy evaluation converges under the same conditions

Policy Evaluation

- V^π exists and is unique if $\gamma < 1$ or termination guaranteed for all states under policy π.
- Policy evaluation converges under the same conditions
- Policy evaluation on the week 0 problem
  - undiscounted, episodic

Policy Evaluation

- V^{π} exists and is unique if $\gamma < 1$ or termination guaranteed for all states under policy π.
- Policy evaluation converges under the same conditions
- Policy evaluation on the week 0 problem
  - undiscounted, episodic
  - Are the conditions met?

Policy Evaluation

- V^π exists and is unique if $\gamma < 1$ or termination guaranteed for all states under policy π.

- Policy evaluation converges under the same conditions

- Policy evaluation on the week 0 problem
  - undiscounted, episodic
  - Are the conditions met?

- Exercises 4.1, 4.2

Policy Improvement

- Policy improvement theorem:
  $\forall s, q_\pi(s, \pi'(s)) \geq v_\pi(s) \Rightarrow \forall s, v_{\pi'}(s) \geq v_\pi(s)$

Policy Improvement

- Policy improvement theorem:
  $\forall s, q_\pi(s, \pi'(s)) \geq v_\pi(s) \Rightarrow \forall s, v_{\pi'}(s) \geq v_\pi(s)$
- Polynomial time convergence (in number of states and actions) even though m^n policies.
  - Ignoring effect of γ and bits to represent rewards/transitions

Policy Improvement

- Policy improvement theorem:
  $\forall s, q_\pi(s, \pi'(s)) \geq v_\pi(s) \Rightarrow \forall s, v_{\pi'}(s) \geq v_\pi(s)$
- Polynomial time convergence (in number of states and actions) even though m^n policies.
  - Ignoring effect of γ and bits to represent rewards/transitions

Value Iteration on Week 0 problem

- Show the new policy at each step
  - Not actually to compute policy

Value Iteration on Week 0 problem

- Show the new policy at each step
  - Not actually to compute policy
  - Break policy ties with equiprobable actions

Value Iteration on Week 0 problem

- Show the new policy at each step
  - Not actually to compute policy
  - Break policy ties with equiprobable actions
  - No stochastic transitions

Value Iteration on Week 0 problem

- Show the new policy at each step
  - Not actually to compute policy
  - Break policy ties with equiprobable actions
  - No stochastic transitions
- How would policy iteration proceed in comparison?
  - More or fewer policy updates?

Value Iteration on Week 0 problem

- Show the new policy at each step
  - Not actually to compute policy
  - Break policy ties with equiprobable actions
  - No stochastic transitions
- How would policy iteration proceed in comparison?
  - More or fewer policy updates?
  - True in general?

Chapter 4 Summary

- Chapter 4 treats **bootstrapping** with a model

Chapter 4 Summary

- Chapter 4 treats **bootstrapping** with a model
  – Next: no model and no bootstrapping

Chapter 4 Summary

- Chapter 4 treats **bootstrapping** with a model
  - Next: no model and no bootstrapping
  - Then: no model, but bootstrapping