

CMSC 726

Lecture 24: Reinforcement Learning – Part II + wrapup

Lise Getoor
December 2, 2010

ACKNOWLEDGEMENTS: The material in this course is a synthesis of materials from many sources, including: Hal Daume III, Mark Drezde, Carlos Guestrin, Andrew Ng, Ben Taskar, Eric Xing, and others. I am very grateful for their generous sharing of insights and materials.

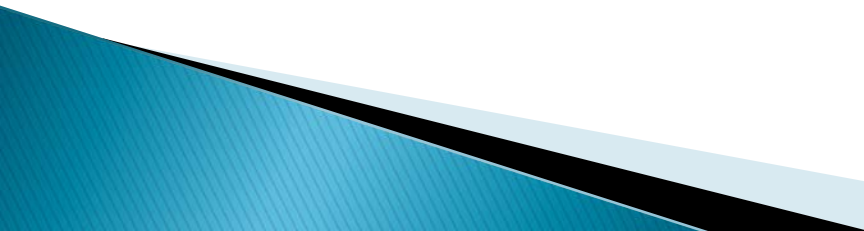
Outline

- ▶ Learning MDPs
 - Model-based vs. Model-free
 - Q-Learning
 - Exploration / Exploitation tradeoff
- ▶ Wrapup

Two main reinforcement learning approaches

- ▶ Model-based approaches:
 - explore environment, then learn model ($P(\mathbf{x}'|\mathbf{x},\mathbf{a})$ and $R(\mathbf{x},\mathbf{a})$) (almost) everywhere
 - use model to plan policy, MDP-style
 - approach leads to strongest theoretical results
 - works quite well in practice when state space is manageable
- ▶ Model-free approach:
 - don't learn a model, learn value function or policy directly
 - leads to weaker theoretical results
 - often works well when state space is large

Q-Learning

- ▶ An agent knows what state it is in and it has a number of actions it can perform in each state.
 - ▶ Initially it doesn't know the value of any of the states
 - ▶ augments value iteration by maintaining a utility value $Q(s,a)$ for every action at every state.
 - ▶ Value of a state $V(s)$ or $Q(s)$ is simply the maximum Q value over all the possible actions at that state.
 - ▶ The agent learns the values of states as it works its way through the state space.
- 

Exploration


- ▶ The agent may occasionally choose to explore suboptimal moves in the hopes of finding better outcomes. Only by visiting all the states frequently enough can we guarantee learning the true values of all the states.

Q-Learning

- ▶ foreach state s
 foreach action a $Q(s,a)=0$
 $s = \text{currentstate}$
 do forever
 $a = \text{select an action}$
 do action a
 $r = \text{reward from doing } a$
 $t = \text{resulting state from doing } a$
 $Q(s,a) += \alpha * (r + \gamma * (Q(t)-Q(s,a)))$
 $s = t$
- ▶ Notice that a learning coefficient, α , has been introduced into the update equation. Normally α is set to a small positive constant less than 1.

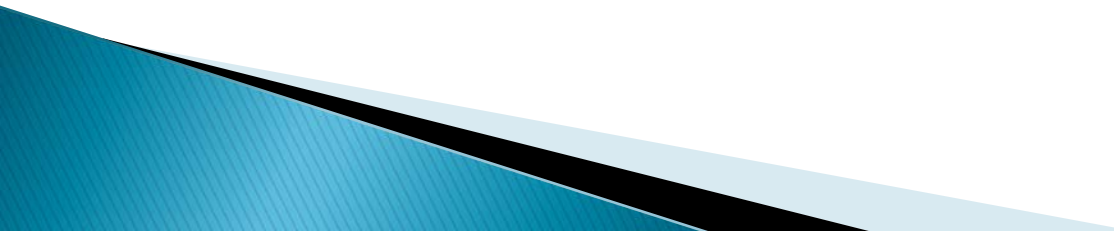
Selecting an Action

- ▶ simply choose action with highest expected utility?

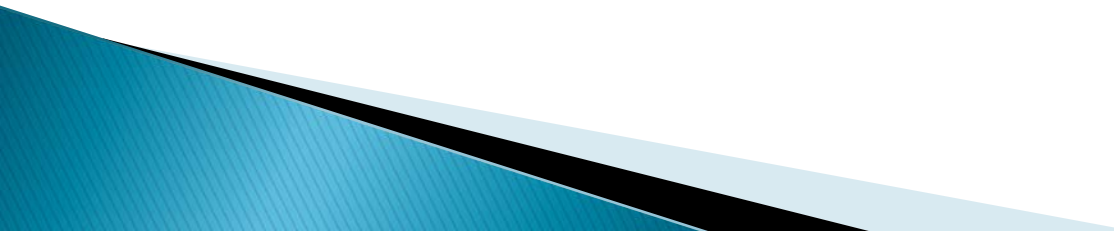
stuck in a rut
- ▶ problem: action  has two effects
 - gains reward on current sequence
 - information received and used in learning for future sequences
- ▶ trade-off immediate good for long-term well-being

jumping off a cliff just because you've never done it before...

Exploration policy

- ▶ wacky approach: act randomly in hopes of eventually exploring entire environment
 - ▶ greedy approach: act to maximize utility using current estimate
 - ▶ need to find some balance: act more wacky when agent has little idea of environment and more greedy when the model is close to correct
 - ▶ example: one-armed bandits...
- 

RL Summary

- ▶ active area of research
 - ▶ both in OR and AI
 - ▶ several more sophisticated algorithms that we have not discussed
 - ▶ applicable to game-playing, robot controllers, others
- 

Announcements

▶ University Course Assessments

- Please fill these out

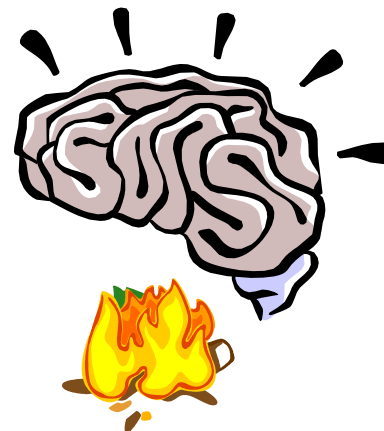
▶ Project:

- Poster session: Tuesday Dec 7 3:30–6:30pm, AV Williams 2120
 - please arrive a 15mins early to set up
- Paper: December 9th by midnight
 - maximum of 10 pages (can have appendix)
 - Submit electronically
- Final
 - 12/ 18 / 2010, 10:30am – 12:30pm, CSI 1121

Wrapup

What you have learned this semester

- ▶ Learning is function approximation
- ▶ MLE
- ▶ Regression
- ▶ Discriminative v. Generative learning
- ▶ Naïve Bayes
- ▶ Logistic regression
- ▶ Bias–Variance tradeoff
- ▶ Neural nets
- ▶ Decision trees
- ▶ Cross validation
- ▶ Boosting
- ▶ Instance–based learning
- ▶ SVMs
- ▶ Kernel trick
- ▶ PAC learning
- ▶ VC dimension
- ▶ K–means
- ▶ EM
- ▶ Bayes nets
 - representation, inference, parameter and structure learning
- ▶ MDPs
- ▶ Reinforcement learning
- ▶



BIG PICTURE

- ▶ Improving the performance at some task though experience!!! 😊
 - before you start any learning task, remember the fundamental questions:

What is the learning problem?

From what experience?

What model?

What loss function are you optimizing?

With what optimization algorithm?

Which learning algorithm?

With what guarantees?

How will you evaluate it?

What next?

- ▶ **Journal:**
 - JMLR – Journal of Machine Learning Research (free, on the web)
 - MLJ – Machine Learning Journal
- ▶ **Conferences:**
 - ICML: International Conference on Machine Learning
 - NIPS: Neural Information Processing Systems
 - COLT: Computational Learning Theory
 - UAI: Uncertainty in AI
 - AIStats: intersection of Statistics and AI
 - Datamining conferences: KDD, ICDM, SDM
 - Also AAAI, IJCAI and others
- ▶ **Many conferences have associated workshops**
 - Good places to present preliminary work
 - Good opportunity to meet people
 - NIPS, ICML, KDD, AAAI, etc.

You have done a lot!!!

- ▶ And (hopefully) learned a lot!!!
 - Implemented a number of ML algorithms
 - Answered hard questions and proved interesting results
 - Completed a fantabulous ML project
 - And did excellently on the final!

Thank you for the
work hard and
Good Luck!!!