Monte Carlo methods for solving Markov Decision Processes

A talk on course T-61.6020:
Reinforcement learning – theory and applications

Based on Ch. 5 in the book by Sutton & Barto

Ville.Viitaniemi@tkk.fi
2006-2-7

Contents

- What are Monte Carlo Methods?
- Background and motivation
- MC Policy evaluation
- MC Action value estimation
- MC Control
- Summary

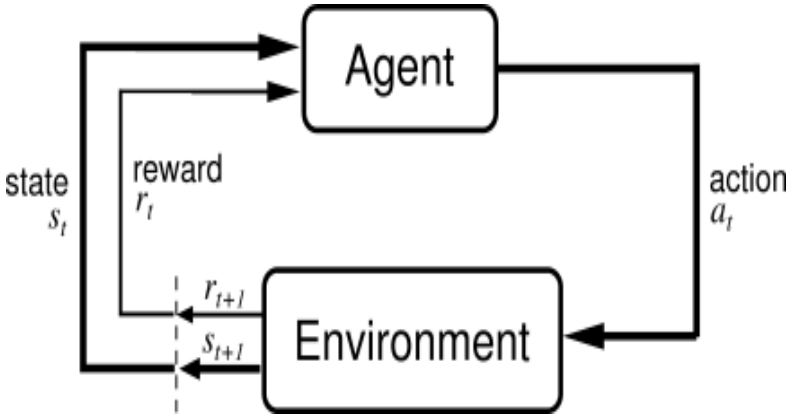
What is Monte Carlo

- use of random samples, hope that results average to right answer with large number of samples
- example: numerical integration by sampling
- example: estimating posteriori distribution in Bayesian inference by samplibg
- carrying one single sample value through calculations is much easier than considering whole distributions at once

Background

- reinforcement learning problems formalised as Markov Decision Processes
- environment dynamics
  - completely known -> “optimal control”
  - unknown: dynamics must be learned simultanously with control
    - exploitation vs. exploration dilemma
    - example: random exploration useful in simple simulations

Notation



- at time t environment in state s_t
- the agent performs action a_t
- agent controlled by policy $\pi = \pi(s, a)$: probabilities of choosing action a in state s
- the environment moves into state s_{t+1} sampled from density $P^{a_t}_{s_t s_{t+1}}$
- the agent receives a reward r_{t+1} according
- to density $R^{a_t}_{s_t s_{t+1}}$

Dynamic programming

- if dynamics of the environment completely known, optimal policy can be solved by dynamic programming
- DP based on
  - 1) dividing problem into subproblems
  - 2) storing the results of overlapping subproblems
- in MDP's storage via state and state-action value functions $V^{\pi}(s)$ and $Q^{\pi}(s,a)$

Dynamic programming

- Bellman's optimality equation turned into algorithms
  - policy evaluation
  - policy iteration: policy evaluation+improvement
  - generalised policy iteration
- mathematically sound
- requires complete knowledge of dynamics
- curse of dimensionality

Monte Carlo methods

- approximate expected returns by empirical averages as experience is observed
  - experience: a terminating sequence of states, actions and reward
- division of time into episodes enforced
  - averaging of returns over the episode
  - more fine-grained division of time not considered
- on-line (actual) experience
- simulated experience
  - complete model of the environment still not required

Monte Carlo policy evaluation

- repeat until convergence:
  - generate experience using policy π
  - for each visited state compute the following return
  - average states' observed return over episodes
- two flavours:
  - first visit
  - every visit
- both proven to converge to $V^{\pi}(s)$
  - rate of convergence $O(1/\sqrt{n})$

MC policy evaluation

- MC estimates independent for each state
  - not bound together via Bellman's equation
  - i.e. MC method doesn't bootstrap

Monte Carlo action value estimation

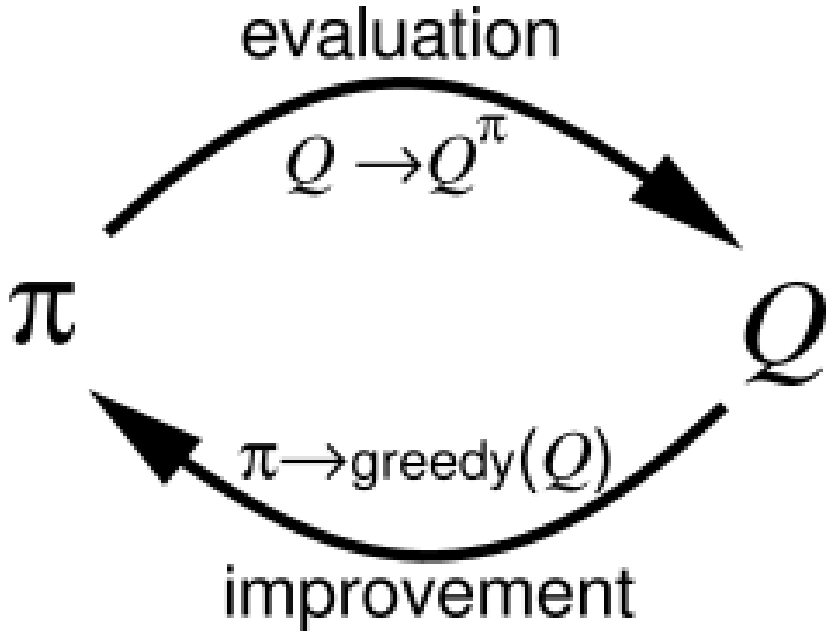
- without model of dynamics, state values not enough to determine policy
  - not known which states follow from an action
  - action values needed instead
  - with MC methods emphasis on estimating Q*
- basically the same MC method as for states
  - same rate of convergence

MC action value estimation

- main complication: all state-action pairs may not be in the experience
  - esp. deterministic policies
- this is problem of maintaining exploration
- remedy 1: exploring starts
  - start distribution forced to have prob. >0 for all pairs
  - what to do with real experience
- remedy 2: consider only stochastic policies with prob. > 0 for all pairs

Monte Carlo control

- same idea as discussed before in connection with DP: generalised policy iteration
  - alternating action value evaluation and greedy policy improvement steps



Monte Carlo control

- e.g. classical policy iteration with full iterative action value evaluation

$$\pi_0 \xrightarrow{\text{E}} Q^{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} Q^{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi^* \xrightarrow{\text{E}} Q^*,$$

- for convergence also less exact (less tedious) value iteration seems to be sufficient
  - moving Q towards actual value function enough
  - convergence of such MC method still to be proved, even though it seems inevitable

Maintaining exploration

- to get experience from all state-action pairs, experience needs to be generated with soft policy that assigns p > 0 for all actions
  - still, policy may be very close to optimal deterministic policy, if the probability mass distributed to the remaining actions is kept small
  - ϵ-soft policy: random action with prob = ϵ
  - luckily, policy iteration works also for ϵ -soft policies
    - greedy policy improvement step replaced with ϵ -greedy variant

Maintaining exploration: on-policy and off-policy control

- experience can be generated by
  - 1) same policy that is to be evaluated and optimised (on-policy control)
  - 2) different soft policy (off-policy control)
    - e.g. soft policy for control, greedy policy to be estimated
- when using off-policy control, the experienced returns must be weighted
  - the weight factor determined by the policies only, no knowledge of environment dynamics needed
  - potential problem: slow learning, only tails of experience lead to significant weights

Incremental implementation

- MC methods can be implemented incrementally
  - i.e. no need to store all the previous experience, just the accumulated returns
- policy improves over time
  - nonstationary return distributions
  - possibly desirable to e.g. weight recent returns more heavily

Summary of MC methods

- policy evaluation through “empirical averages”
- no knowledge about environment synamics needed
- can be used with simulated experience
- MC methods can be focused on a set of interesting states
- possibly less prone to violations to Markov property of states thaan DP (states not so intermingled)

Summary

- GPI techniques applicable
  - counterparts of policy evaluation and policy improvement exist
- maintaining sufficient exploration an issue
  - exploring starts
  - soft policies
    - on-policy and off-policy control
- identification on the methods only recently
  - little proofs of convergence
  - effectiveness tested only in some cases