Reinforcement Learning and Decision Making Under Uncertainty

Christos Dimitrakakis

March 21, 2023

Outline

Schedule

Course name:
The course will give a thorough introduction to reinforcement learning. The first 8 weeks will be devoted to the core theory and algorithms of reinforcement learning. The final 6 weeks will be focused on project work, during which more advanced topics will be inroduced.
The first 6 weeks will require the students to complete 5 assignments. The remainder of the term, the students will have to prepare a project, for which they will need to submit a report.

Week	Topic
1	Beliefs and Decisions
2	Bayesian Decision Rules
3	Introduction to Bandit problems.
4	Finite Horizon MDPs Backwards Induction The Bandit MDP
5	Infite Horizon MDPs Value Iteration Policy Iteration
6	Sarsa / Q-Learning
7	Model-Based RL
8	Function Approximation, Gradient Methods
9	Bayesian RL: Dynamic Programming, Sampling
11	UCB/UCRL/UCT. UCT/AlphaZero.
12	Inverse Reinforcement Learning
13	Multiagent extensions: Bayesian Games
13	Project presnetations
14	Q&A, Mock exam

Utility theory (90')

1. Rewards and preferences (15')
2. Transitivity of preferences (15')
3. Random rewards (5')
4. Decision Diagrams (10')
5. Utility functions and the expected utility hypothesis (15')
6. Utility exercise: Gambling (10' pen and paper)
7. The St. Petersburg Paradox (15')

Probability primer

1. Objective vs Subjective Probability: Example (5')
2. Relative likelihood: Completeness, Consistency, Transitivity, Complement, Subset (5')
3. Measure theory (5')
4. Axioms of Probability (5')
5. Random variables (5')
6. Expectations (5')
7. Expectations exercise (10')
8.
9. Quantum Physics
10. Coin toss
11. Relative Likelihood

Completeness A>B, B>A or A=B for any A,B Transitivity A>B, B>C, A>C Complement: A>B => ~A<~B Subset: $A \subset B \Rightarrow A < B$

1. Measure theory

We can use probability to quantify this, so that $A > B$ iff $P(A) > P(B)$. But what do we mean by this?

Lab: Probability, Expectation, Utility

1. Exercise Set 1. Probability introduction.
2. Exercise Set 2. Sec 2.4, Exercises 4, 5.

Assignment.

Exercise 7, 8, 9.

Seminar:

Utility. What is the concept of utility? Why do we want to always maximise utility?

Example:

U	w1	w2
a1	4	1
a2	3	3

Regret. Alternative notion.

L	w1	w2
a1	0	2
a2	1	0

Minimising regret is the same as maximising utility when w does not depend on a. Hint: So that if $E[L|a^*] \leq E[L|a]$ for all a', $E[U|a^*] \geq E[L|a]$ for all a',

The utility analysis of choices involving risk: `https://www.journals.uchicago.edu/doi/abs/10.1086/256692`

The expected-utility hypothesis and the measurability of utility `https:`

Problems with Observations (45')

1. Discrete set of models example: the meteorologists problem (25')
2. Marginal probabilities (5').
3. Conditional probability (5').
4. Bayes theorem (10').

Statistical decisions (45')

1. ML Estimation (10')
2. MAP Estimation (10')
3. Bayes Estimation (10')
4. MSE Estimation (10') [not done]
5. Linearity of Expectations (10') [not done]
6. Convexity of Bayes Decisions (10') [not done]

Lab: Decision problems and estimation (45')

1. Problems with no observations. Book Exercise: 13,14,15.
2. Problems with observations. Book Exercise: 17, 18.

Assignment: James Randi

n meteorologists as prediction with expert advice

- Predictions $p_t = p_{t,1}, \ldots, p_{t,n}$ of all models for outcomes y_t
- Make decision a_t.
- Observe true outcome y_t
- Obtain instant reward $r_t = \rho(a_t, y_t)$
- Utility $U = \sum_{t=1}^T r_t$.
- T is the problem horizon

At each step t:

1. Observe p_t.
2. Calculate $\hat{p}_t = \sum_\mu \xi_t(\mu) p_{t,\mu}$
3. Make decision $a_t = \arg\max_a \sum_y \hat{p}_t(y) \rho(a, y)$.
4. Observe y_t and obtain reward $r_t = \rho(a_t, y_t)$.
5. Update: $\xi_{t+1}(\mu) \propto \xi_t(\mu) p_{t,\mu}(y_t)$.

The update does not depend on a_t

Prediction with expert advice

- Advice $p_t = p_{t,1}, \ldots, p_{t,n} \in D$
- Make prediction $\hat{p}_t \in D$
- Observe true outcome $y_t \in Y$
- Obtain instant reward $r_t = u(\hat{p}_t, y_t)$
- Utility $U = \sum_{t=1}^{T} r_t$.

Relation to n meteorologists

- D is the set of distributions on Y.
- However, there are only predictions, no actions. To add actions:

$$u(\hat{p}_t, y_t) = \rho(a^*(\hat{p}_t), y_t), \qquad a^*(\hat{p}_t) = \arg\max_a \rho(a, y_t)$$

The update does not depend on a_t

The Exponentially Weighted Average

MWA Algorithm

- Predict by averaging all of the predictions:
$$\hat{p}_t(y) = \sum_{\mu} \xi_t(\mu) p_{t,\mu}(y)$$

- Update by weighting the quality of each prediction
$$\xi_{t+1}(\mu) = \frac{\xi_t(\mu)\exp[\eta u(p_{t,\mu}, y_t)]}{\sum_{\mu'} \xi_t(\mu')\exp[\eta u(p_{t,\mu}, y_t)]}$$

Choices for u

- $u(p_{t,\mu}, y_t) = \ln p_{t,\mu}(y_t)$, $\eta = 1$, Bayes's theorem.
- $u(p_{t,\mu}, y_t) = \rho(a^*(p_{t,\mu}), y_t)$: quality of expert prediction.

The n armed stochastic bandit problem

- Take action a_t
- Obtain reward $r_t \sim P_{a_t}(r)$ with expected value μ_{a_t}.
- The utility is $U = \sum_t r_t$, while P is unknown.

The Regret

-Total regret with respect to the best arm:

$$L \triangleq \sum_{t=1}^{T} [\mu^* - r_t], \qquad \mu^* = \max_a \mu_a$$

- Expected regret of an algorithm π:

$$\mathbb{E}^\pi[L] = \sum_{t=1}^{T} \mathbb{E}^\pi[\mu^* - r_t], = \sum_{a=1}^{n} \mathbb{E}^\pi[n_{T,a}](\mu^* - \mu_a)$$

- $n_{T,a}$ is the number of times a has been pulled after n steps.

Bernoulli bandits

A classical example of this is when the rewards are Bernoulli, i.e.

$$r_t | a_t = i \sim \mathrm{Bernoulli}(\mu_i)$$

Greedy algorithm

- Take action $a_t = \arg\max_a \hat{\mu}_{t,a}$
- Obtain reward $r_t \sim P_{a_t}(r)$ with expected value μ_{a_t}.
- Update arm: $s_{t,a_t} = s_{t-1,a_t} + r_t$, $n_{t,a_t} = n_{t-1,a_t} + 1$.
- Others stay the same: $s_{t,a} = s_{t-1,a}$, $n_{t,a} = n_{t-1,a}$ for $a \neq a_t$.
- Update means: $\hat{\mu}_{t,i} = s_{t,i}/n_{t,i}$.

Policies and exploration

- $n_{t,i}, s_{t,i}$ are sufficient statistics for Bernoulli bandits.
- The more often we pull an arm, the more certain we are the mean is correct.

Upper confidence bound: exploration bonuses

- Take action $a_t = \arg\max_a \hat{\mu}_{t,a} + O(1/\sqrt{n_{t,a}})$.

Posterior sampling: randomisation

- Given some prior parameters $\alpha, \beta > 0$ (e.g. 1).
- $\xi_t(\mu_a) = \mathrm{Beta}(\alpha + s_{t,a}, \beta + n_{t,a} - s_{t,a})$.
- Sample $\hat{\mu} \sim \xi_t(\mu)$.
- Take action $a_t = \arg\max_a \hat{\mu}_a$.

The upper confidence bound

Let

$$\hat{\mu}_n = \sum_{i=1}^{t} r_i/n,$$

be the sample mean estimate of an iiid RV in [0,1] with $\mathbb{E}[r_i] = \mu$. Then we have

$$\mathbb{P}(\hat{\mu}_n \geq \mu + \epsilon) \leq \exp(-2n\epsilon^2)$$

or equivalently

$$\mathbb{P}(\hat{\mu}_n \geq \mu_n + \sqrt{\ln(1/\delta)/2n} \leq \delta.)$$

Beta distributions as beliefs

- [Go through Chapter 4, Beta distribution]
- [Visualise Beta distribution]
- [Do the James Random Exercise 3]
- Note that the problem here is that this is only a point estimate: it ignores uncertainty. In fact, we can represent our uncertainty about the arms in a probabilistic way with the Beta distribution:
  If our prior over an arm's mean is $\mathrm{Beta}(\alpha, \beta)$ then the -posterior at time t is $\mathrm{Beta}(\alpha + s_{t,i}, \beta + n_{t,i} - s_{t,i})$.
- [Visualise how the posterior changes for a biased coin as we obtain more data].

Assignment and exercise

1. Implement epsilon-greedy bandits (lab, 30')
2. Implement Thompson sampling bandits (lab, 30')

3, Implement UCB bandits (home)

1. Compare them in a benchmark (home)

1. The bandit MDP (30')
2. MDP definitions (15')
3. MDP examples (15')
4. Monte Carlo Policy Evaluation (15')
5. DP: Finite Horizon Policy Evaluation (15')
6. DP: Finite Horizon Backward Induction (15')
7. DP: Proof of Backwards Induction (15')
8. DP: Implementation of Backwards Induction (30')

The Markov decision process

Interaction at time t

- Observe state $s_t \in S$
- Take action $a_t \in A$.
- Obtain reward $r_t \in \mathbb{R}$.

The MDP model μ

- Transition kernel $P_\mu(s_{t+1} | s_t, a_t)$.
- Reward with mean $\rho_\mu(s_t, a_t)$

Policies

- Markov policies $\pi(a_t | s_t)$

Utility

Total reward up to a finite (but not necessarily fixed) horizon T

$$U_1 = \sum^{T} r_t$$

MDP examples

Shortest path problems

- Goal state $s^* \in S$.
- Reward $r_t = -1$ for all $s \neq s^*$
- Game ends time T where $s_T = s^*$.

Blackjack against a croupier

- Croupier shows one card.
- Current state is croupier's card and your cards.
- Reward is $r_T = 1$ if you win, $r_T = -1$ if you lose at the end, otherwise 0.

Monte Carlo Policy Evaluation

$$
\begin{aligned}
V_t^{\pi}(s) &= \mathbb{E}^{\pi}[U_t | s_t = s] \\
&\approx \frac{1}{N} \sum_{n=1}^{N} U_t^{(n)}
\end{aligned}
$$

Policy Evaluation

$$
\begin{aligned}
V_t^\pi(s) &= \mathbb{E}^\pi[U_t | s_t = s] \\
&= \mathbb{E}^\pi[\sum_{k=t}^{T} r_k | s_t = s] \\
&= \mathbb{E}^\pi[r_t | s_t = s] + \mathbb{E}^\pi[\sum_{k=t+1}^{T} r_k | s_t = s] \\
&= \mathbb{E}^\pi[r_t | s_t = s] + \mathbb{E}^\pi[U_{t+1} | s_t = s] \\
&= \mathbb{E}^\pi[r_t | s_t = s] + \sum_{s'} \mathbb{E}^\pi[U_{t+1} | s_{t+1} = s'] \, \mathbb{P}^\pi(s_{t+1} = s' | s_t = s) \\
&= \mathbb{E}^\pi[r_t | s_t = s] + \sum_{s'} V_{t+1}^\pi(s') \, \mathbb{P}^\pi(s_{t+1} = s' | s_t = s) \\
&= \mathbb{E}^\pi[r_t | s_t = s] + \sum_{s'} V_{t+1}^\pi(s') \sum_{a} \mathbb{P}(s_{t+1} = s' | s_t = s, a_t = a) \pi_t(a|
\end{aligned}
$$

Backwards induction

Let v_t be the estimates of the backwards induction algorithm. We want to prove that $v_t = V_t^*$. This is true for $t = T$. Let us assume by induction that $v_{t+1} > V_{t+1}^*$. Then it must hold for t as well:

$$\begin{aligned}
v_t(s) &= \max_a r(s) + \sum_j p(j|s,a) v_{t+1}(j) \\
&\geq \max_a r(s) + \sum_j p(j|s,a) V_{t+1}^*(j) \\
&\geq \max_a r(s) + \sum_j p(j|s,a) V_{t+1}^\pi(j) & \forall \pi \\
&\geq V_t^\pi(s)
\end{aligned}$$

If π^* is the policy returned by backwards induction, then $v_t = V^{\pi^*}$. Consequently

$$V^* \geq V^* \pi^* = v \geq V^* \Rightarrow v = V^*.$$

Plan

1. DP: Value Iteration (45')
2. DP: Policy Iteration (45')

Infinite horizon setting

Utility

$$U = \sum_{t=0}^{\infty} \gamma^t r_t$$

Discount factor $\gamma \in (0, 1)$

Tells us how much we care about the future. Note that

$$\sum_{t=0}^{\infty} \gamma^t = \frac{1}{1-\gamma}$$

Value iteration

Idea: Run backwards induction, discounting by γ until convergence.

Algorithm

- Input: MDP μ, discount factor γ, threshold ϵ
- $v_0(s) = \rho_\mu(s)$ for all s
- For $n = 1, \ldots$

$$v_{n+1}(s) = \rho_\mu(s) + \gamma \sum_j P_\mu(j|s,a) v_n(j).$$

- Until $\|v_{n+1} - v_n\|_\infty \leq \epsilon$.

Norms

- $\|x\|_1 = \sum_t |x_t|$
- $\|x\|_\infty = \max_t |x_t|$
- $\|x\|_p = (\sum_t |x_t|^p)^{1/p}$

Matrix notation for finite MDPs

- r: reward vector.
- P_π: transition matrix.
- v: value function vector.

Stationary policies

$$\pi(a_t|s_t) = \pi(a_k|s_k)$$

Matrix formula for value function

$$v^\pi = \sum_{t=0}^{\infty} \gamma^t P_\pi^t r.$$

Note that $(P_\pi r)(s) = \sum_j P_\pi(s,j) r(j)$.

Convergence of value iteration

Proof idea

1. Define the VI operator L so that $v_{n+1} = Lv_n$.
2. Show that if $v = V^*$ then $v = Lv$.
3. Show that $\lim_{n\to\infty} v = V^*$.

Further questions

- How fast does it converge?
- When is the policy actually optimal?

Policy evaluation

Policy evaluation theorem

For any stationary policy π, the unique solution of

$$v = r + \gamma P_\pi v \qquad \text{is} \qquad v^\pi = (I - \gamma P_\pi)^{-1} r$$

Proof

If $\|A\| < 1$, then $(I - A)^{-1}$ exists and

$$(I - A)^{-1} = \lim_{T \to \infty} \sum_{t=0}^{T} A^t.$$

Interpretation: $X = (I - P)^{-1}$

Is the total discounted number of times reaching a state

$$X(i, j) = \mathbb{E} \sum_{t=0}^{\infty} \gamma^t \, \mathbb{I} \{s_t = j | s_0 = i\}$$

Optimality equations

Policy operator

$$L_\pi v = r + \gamma P_\pi v.$$

Bellman operator

$$Lv = \max_\pi \{r + \gamma P_\pi v\}.$$

Bellman optimality equation

$$v = Lv$$

Value iteration convergence proof

Contraction mappings

M is a contraction mapping if there is $\gamma < 1$ so that

$$\|Mx - My\| \leq \gamma \|x - y\| \qquad \forall x, y.$$

Banach fixed point theorem

If M is a contraction mapping

1. There is a unique x^* so that $Mx^* = x^*$.
2. If $x_{n+1} = Mx_n$ then $x_n \to x^*$.

Value iteration

- Since L is a contraction mapping, it converges to $v^* = Lv^*$ (Theorem 6.5.7)
- If $v = Lv$ then $v = \max_\pi v^\pi$ (Theorem 6.5.3)
- Hence, value iteration converges to v^*.

Speed of convergence of value iteration

Theorem

If $r_t \in [0, 1]$, $v_0 = 0$, then

$$\|v_n - v^*\| \leq \gamma^n/(1-\gamma).$$

Proof

Note that $\|v_0 - v^*\| = \gamma^0/(1-\gamma)$, and

$$\|v_{n+1} - v^*\| = \|Lv_n - Lv^*\| \leq \gamma\|v_n - v^*\|.$$

Induction: $\|v_n - v^*\| \leq \gamma^n/(1-\gamma)$

$$\|v_{n+1} - v^*\| \leq \gamma\|v_n - v^*\| \leq \gamma^{n+1}/(1-\gamma).$$

1. DP: Temporal Differences (45')
2. DP: Modified Policy Iteration (45')

1. Sarsa (45')
2. Q-learning (45')

1. Actor-Critic Algorithms (45')
2. Model-based RL (45')

1. Fitted Value Iteration (45')
2. Approximate Policy Iteration (45')

1. Direct Policy Gradient, i.e. REINFORCE (45')
2. Actor-Critic Methods, e.g. Soft Actor Critic (45')

1. Thompson sampling (25')
2. Bayesian Policy Gradient (20')
3. BAMDPs (25')
4. POMDPs (20')

1. UCB (45')
2. UCRL (45')

1. UCT (45')
2. Alphazero (45')

1. Linear Models (20')
2. Gaussian Processes (25')
3. GPTD (45')

1. Apprenticeship learning (45')
2. Probabilistic IRL (45')

   Bayesian games (90')