Course 2, Module 3
Temporal Difference Learning Methods for Control

CMPUT 397
Fall 2020

Comments

- Review C2M3 today
- Also finish off a few last things from C2M2
- Clarification on types of questions for slido: they must be about the RL content for that module
- Any questions?

Review of Course 2, Module 3
TD Control

Video 1: Sarsa: GPI with TD

- Building an algorithm to find near optimal policies: SARSA (**S**tate, **A**ction, **R**eward, Next **S**tate, **A**ction). Combining the ideas of *policy evaluation, policy improvement, TD,* and *epsilon-soft policies*

- Goals:

  - explain how **generalized policy iteration** can be used with TD to find improved policies

  - Describe the Sarsa Control algorithm

Video 2: Sarsa in the Windy Grid World

- We ran a fun **experiment with Sarsa** on a fancy gridworld
- Goals:
    - Understand how the Sarsa control algorithm operates in an example MDP.
        - the Windy Gridworld
    - Gain experience analyzing the performance of a learning algorithm.
        - understanding the plot of cumulative episodes completed vs steps

Video 3: What is Q-learning

- Just the most famous RL algorithm! Similar to SARSA, but learns the **optimal policy**
- Goals:
  - Describe the Q-learning algorithm
  - Explain the relationship between Q-learning and the Bellman optimality equations

Video 4: Q-learning in the Windy Gridworld

- How does Q-learning work in practice? We get some insight with an **experiment comparing with SARSA**

- Goals:

  - Gain insight into how Q-learning performs in an example MDP

  - Gain insight into the **differences between Q-learning and Sarsa**.

Video 5: How is Q-learning Off-policy?

- Q-learning **learns about** the greedy policy (which eventually becomes π*), while **following a different policy** ε-greedy. That is off-policy, but there are no importance sampling corrections!

- Goals:

  - Understand how Q-learning can be off-policy **without using importance sampling**

  - Describe how learning on-policy or off-policy might affect performance in control.

    - SARSA (on-policy learning), can be better!

Video 6: Expected SARSA

- **A new TD Control method**! Uses the probability of each action under the current policy in its update!
- Goals:
  - explain the Expected Sarsa algorithm.

Video 7: Expected SARSA in the Cliff World

- Why all the fuss about Expected Sarsa? We find out with an **experiment** in another gridworld: The cliff world. Spoiler: **Expected Sarsa learns faster** AND **is more robust to our choice of alpha**

- Goals:

    - Describe Expected Sarsa’s behaviour in an example MDP.

    - And Empirically compare Expected Sarsa and Sarsa

Video 8: The Generality of Expected SARSA

- Expected SARSA is pretty neat! It can perform better than either SARSA or Q-learning. In addition, the algorithm can be **used in different ways**

- Goals:

  - Understand how Expected Sarsa can do **off-policy** learning without using importance sampling

  - Explain how Expected Sarsa **generalizes Q-learning**

Terminology Review

- TD methods we have learned about are **tabular, one-step, model-free** learning algorithms

- **Tabular:** we store the value function in a table. One entry in the table per value, so each value is stored independently of the others. We are implicitly assuming the state-space ($\mathcal{S}$) is small

- **One-step**: we update a single state or state-action value on each time-step. Only the value of Q(S,A) from S -- A --->S',R. We never update more than one value per learning step

- **Model-free**: we don't assume access to or make use of a model of the world. All learning is driven by sample experience. Data generated by the agent interacting with the environment

Clarification Slido Qs

- “How is SARSA and Q-learning different than the previous TD methods we learned last week?”

- “What does "TD control" mean? Is a TD control algorithm supposed to be used some sort of TD prediction algorithm?”

- “Can you explain how learning from state-value is different from action-value and why we look at action-value learning in sarsa and q-learning instead of state-value? Is one better than the other?”

- “What is the difference between asymptotic vs interim performance. Moreover, Expected SARSA is identical to Q-learning when policy (pi) is greedy; will it converge faster to Q optimal (q*) than Q-learning, since the choice of greedy action (a) is updated every time?”

Worksheet Question 3

- Why is Q-learning an off-policy method?
- Why is Sarsa considered on-policy, but Expected Sarsa can be used off-policy?

Next steps

- Let’s do the worksheet questions before we talk about some of the slido questions
- First, let’s finish off slido Qs from last week

Slido Qs: Expected Sarsa

- “If computational power is not an issue, is Expected SARSA always better than SARSA and Q-Learning?”

- “It seems like Expected sarsa is better than sarsa which is better than q-learning (when measuring performance online). How do we know which method to pick when looking at different situations? Is E-Sarsa always the best?”

- “When is Expected Sarsa not a good option?”

- “Besides being computationally expensive, are there any other downfalls to using expected sarsa? If not, then is that always the best way to go?”

- “Isn't Q-learning similarly expensive compared to expected SARSA, given that it needs to iterate over the entire Q(S',A') space as well? Stemming from that, is there really a case where Q-learning is preferred over E-SARSA?”

Stochasticity and Variance

- “When comparing Sarsa and Q-learning in the cliff walking example, why does Sarsa learn the path that goes furthest from the cliff instead of a path that's only 1or 2 rows up from the cliff?”

- “How does sarsa fail to converge with higher values of alpha while expected sarsa stays constant for experiments such as the cliff walking experiment?”

- “If our rewards are stochastic how would q(s,a) change? I ask because while going through the videos and book our problems assume a constant reward per action but that's limiting so I'm curious how that might work.”

- “Since expected sarsa is more complex computationally than sarsa as it eliminates variance. Are there any way to improve this algorithm to make it more efficient?”

Choosing epsilon

- “Why are fixed epsilon values used for greedifying TD methods when it seems like, in general, they benefit from using epsilon values that vary over time such as epsilon=1/t

- “For the cliff walking example, the optimal policy learned by Q-learning is the optimal one but its online performance is worse than a safer policy learned by Sarsa. I wonder whether in practice we prefer having a better online performance and do not learn the exact optimal policy?”

- “For the cliff walk example, would expected sarsa find the same path as sarsa? That is will it avoid the cliff even if it is off policy?”

Convergence

- “How do we know if we have performed a sufficient number of episode iterations to obtain the optimal action-value function for Sarsa and Expected Sarsa? Is there a specific condition that will be met when they have converged?”

- “What is the mathematical intuition behind SARSA failing to converge at high alphas?”

- Related: “I am a little confused on why is larger step-size causing SARSA to fail to converge in the cliff gridworld. Is it just due to SARSA putting too much weight on exploratory steps in epsilon greedy? What would happen if you reduced epsilon, would it cause SARSA to converge with larger stepsize?”

Misc

- “For off-policy methods like Q-Learning and Expected Sarsa, does these algorithms use the behaviour policy b anywhere?”

- “Is my understanding true that Expected Sarsa can be either an off-policy algorithm and an on-policy algorithm, but more used an off-policy? If so, how to choose when to use on-policy and off-policy? How will the result be different?”

- “I wonder asymptotic or interim performance is more important in the real wold? I think asymptotic performance is more important, but if the asymptotic performance are close to each other, will the interim performance be a reason to choose a worse asymptotic performance?”

- “Why does Q-learning not learn about the outcomes of exploratory actions?”