Algorithmic and Statistical Complexity in Exploration-Exploitation Trade-off

Brian Olsen [1] Sam Fatehmanesh [1] Jinhuang Zhou [1] Taylan Kargin [1]

[1]California Institute of Technology, Department of Electrical Engineering

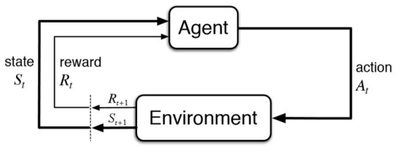
FSRI FIRST-YEAR SUCCESS RESEARCH INSTITUTE

Abstract

- **Objective:** Comparing the effectiveness of different reinforcement learning algorithms
- **Setting:** 10 Armed Gaussian Bandit [1]
- **Motivation:** Reinforcement learning can solve problems in robotics, finance, healthcare, and more [2]
- **Example:** 10 Armed Gaussian Bandit effectively models the best advertisement problem [3]

Background

- **Reinforcement Learning** "teaches" machines using rewards and punishments
  - Algorithm (agent) looks at the current environment (observing the state)
  - Agent chooses available actions in that state using a **policy**
  - Agent receives a reward or punishment from the environment
  - **Goal**: maximize reward



Agent and environment interaction in reinforcement learning [4]

- **Exploration-Exploitation Trade-off**: Agent chooses between exploring options or exploiting the current best action
  - **Rationale**: Agent does not know which action provide the maximum reward
  - Exploring risks lowering overall reward; Exploiting risks never choosing the true best action
- **Our four RL Algorithms**:
  - ϵ-greedy
    - Static ϵ: Explore and exploit randomly
    - Decaying-ϵ: As time increases, algorithm exploits more
  - **Upper Confidence Bound**
    - Each action has a mean reward, algorithm calculates the upper confidence bound (UCB) of that mean
    - Action with greatest UCB value is selected
  - **Thompson Sampling**
    - Maintain a belief about the underlying distribution of each arm's reward.
    - In each iteration, sample from the belief distribution for each arm.
    - Choose the arm with the highest sampled value.
    - Update the belief distributions based on the actual reward received.
  - **Optimistic Thompson Sampling**
    - Combination of the UCB and Thompson Sampling Algorithm
- **Regret**: A performance metric
  - Regret is the difference between the most rewarding action an algorithm can take and the action it took
  - We can measure how good an algorithm is at solving a problem by looking at the end behavior of its regret plot
  - For example, a good algorithm is one whose regret plot converges at a small number really fast.

Data and Graphs

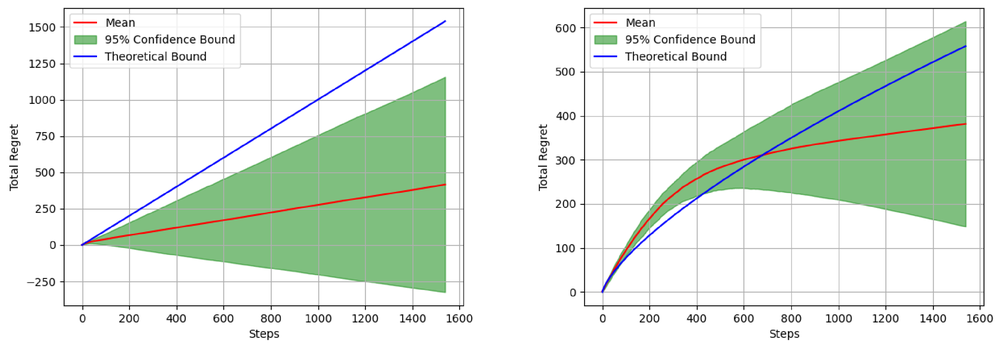


Figure 1. Cumulative regret for epsilon greedy without decay (left) and epsilon greedy with decay (right)

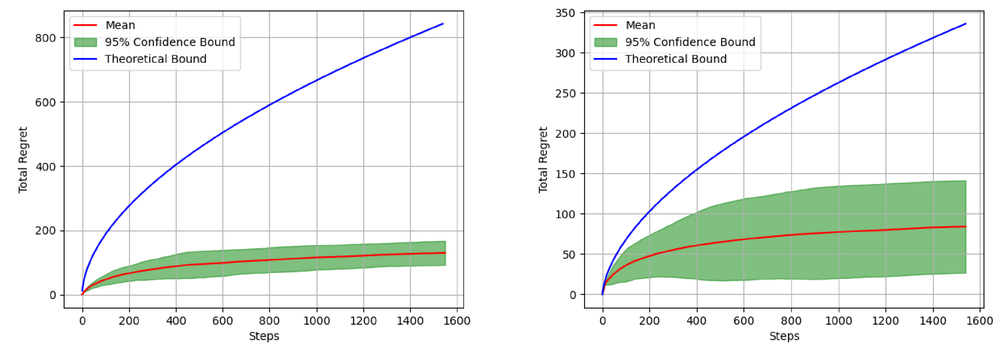


Figure 2. Cumulative regret for Upper Confidence Bound (left) and Optimistic Thompson Sampling (right)

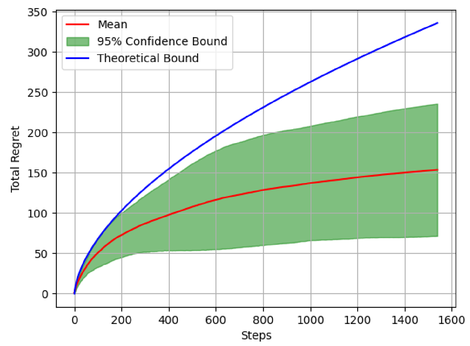


Figure 3. Cumulative regret for Thompson Sampling

Results

Figure 1 through 3 depict the cumulative regret curves for different reinforcement learning algorithms on a linear bandit environment. Epsilon greedy with decay has the expected behavior with a regret curve that is sub linear and logarithmic. Epsilon greedy without decay has the expected behavior with a regret curve that is linear, this is expected as this algorithm will choose a random not necessarily optimal action with a probability of epsilon. Upper confidence bound also has a sub linear logarithmic regret curve which is expected as UCB converges towards the optimal action. Optimistic Thompson Sampling total regret was also sub linear as expected and was the best performing algorithm as it had the lowest upper bound and lowest mean regret.

Mathematical Derivations:

We used a concentration inequality to show when the regret function will verifiably converge with the least amount of samples possible (to be more efficient with the training):

$$|r_t - r_*|_\infty = \max |r_t(a) - r_*(a)|$$

$$2\exp(-2N\varepsilon^2) \leq \delta \Rightarrow N \geq \frac{1}{2\varepsilon^2}\ln\frac{2}{\delta}$$

$$N \geq \frac{1}{2\varepsilon^2}\ln\frac{2|A|}{\delta}$$

We experimentally verified ε to be 0.080564 when the variance of the last trials using the UCB algorithm 100 trials was less than the maximum possible reward signal, meaning that the algorithm's average reward had converged. We used UCB because it is provably convergent. This ε gives us an approximate 1540 trials to converge for each algorithm. Note that this is also a analytical estimate.

Regret Analysis Compilation:

Note: The following represent the theoretic approximate end-behaviours for each of the algorithms.

Upper Confidence Bound Algorithm:

$$\mathbb{E}[R(T)] = O(2\sqrt{\log(\frac{KT}{\delta})}\sqrt{KT})$$

Decaying Epsilon Greedy Algorithm:

$$\mathbb{E}[R(T)] = O(T^{\frac{2}{3}}(K\log(T))^{\frac{1}{3}})$$

Non-Decaying Epsilon Greedy Algorithm:

$$\mathbb{E}[R(T)] = O(T)$$

Thompson Sampling Bound:

$$\mathbb{E}[R(T)] = O(\sqrt{KT\log(T)})$$

Where K is the amount of actions possible, and T is the number of pulls/actions that have occurred.

Future Directions

- Test our algorithms in a linear bandit setting
- Implement 1st order Optimistic Thompson Sampling and compare it to the other algorithms
- Test our algorithms in linear dynamical systems

Acknowledgements

Thank you to FSRI for making this research possible!

Thank you to Adam Blank for organizing this!

Thanks to our mentor Taylan Kargin for helping us throughout the research!

References

[1] ThomasLecat. "THOMASLECAT/Gym-Bandit-Environments: Multi-Armed Bandits Environments for Openai Gym." GitHub, 27 May 2018, github.com/ThomasLecat/gym-bandit-environments.

[2] Chan, Gary. "Applications of Reinforcement Learning in Real World." Medium, Towards Data Science, 5 Aug. 2018, towardsdatascience.com/applications-of-reinforcement-learning-in-real-world-1a94955bcd12.

[3] Russo, Daniel J., et al. "A tutorial on Thompson sampling." Foundations and Trends in Machine Learning, vol. 11, no. 1, 2018, pp. 1–96, https://doi.org/10.1561/9781680834710.

[4] Awan, Abid Ali. Reinforcement Learning Diagram. 16 May 2022. KDnuggets, www.kdnuggets.com/2022/05/reinforcement-learning-newbies.html.

[5] Abbasi-Yadkori, Y., Pál, D., Szepesvári, C. In Advances in Neural Information Processing Systems, pages 2312–2320, December, 2011.

[6] Abeille, M., Lazaric, A. (2017). Linear Thompson sampling revisited. In Electronic Journal of Statistics (Vol. 11, Issue 2). Institute of Mathematical Statistics. https://doi.org/10.1214/17-ejs1341si

[7] Granziol D, Ru B, Zohren S, Dong X, Osborne M, Roberts S. MEMe: An Accurate Maximum Entropy Method for Efficient Approximations in Large-Scale Machine Learning. Entropy. 2019; 21(6):551. https://doi.org/10.3390/e21060551

https://github.com/Sam-Fatehmanesh/FSRIresearch

https://www.overleaf.com/read/kfvdzcbpbpvg