Reducing Regret in Q-Learning with Ensemble Mechanics

Bowen Jing, Kao Kitichotkul, George Wang

Department of Computer Science; Department of Electrical Engineering; Department of Physics, Stanford University

Introduction

Q-learning:

$$Q(s,a) \leftarrow Q(s,a) - \alpha\big(Q(s,a) - (r + \gamma \max_{a' \in \mathcal{A}(s')} Q(s',a'))\big)$$

ε-greedy:

$$\pi_{\text{act}}(s) = \begin{cases} \arg\max_{a \in \mathcal{A}(s)} Q(s,a) & \text{with probability } 1-\epsilon \\ \text{uniformly from } \mathcal{A}(s) & \text{with probability } \epsilon \end{cases}$$

Softmax:

$$\pi_{\text{act}}(a \mid s) = \begin{cases} \frac{\exp(Q(s,a)/\tau)}{\sum_b \exp(Q(s,b)/\tau)} & \text{probability } \epsilon \\ \mathbf{1}[a = \arg\max_{a' \in \mathcal{A}(s)} Q(s,a')] & \text{probability } 1-\epsilon \end{cases}$$

VDBE: adaptive π_{act} automatically decrease ε in response to environment

$$\delta = r + \gamma \max_{a' \in \mathcal{A}(s')} Q(s',a') - Q(s,a)$$

$$\epsilon \leftarrow \lambda \frac{1-\exp(-|\alpha\delta|/\sigma)}{1+\exp(-|\alpha\delta|/\sigma)} + (1-\lambda)\epsilon$$

Goal: improve VDBE to reduce cumulative regret:

$$R(N) = N \times \mathbb{E}_{\tau \sim \pi_{\text{opt}}}\Big[\sum_{t \in \tau} r(\hat{s}_t, \hat{a}_t)\Big] - \sum_{i=1}^{N} \sum_{t \in \tau_i} r(s_t, a_t)$$

Methodology

Let

$$Q(s,a) = f(\phi(s,a); \mathbf{w})$$

$$\delta_{\mathbf{w}} = \nabla_{\mathbf{w}} \tfrac{1}{2}\big(Q(s,a) - (r + \gamma \max_{a' \in \mathcal{A}(s')} Q(s',a'))\big)^2$$

Fluctuation energy, where $\hat{a}$ is action from previous state

$$H(s,\hat{a}) = |Q(s,\hat{a}) - Q'(s,\hat{a})|$$

Then

$$\epsilon \leftarrow \lambda \frac{1-\exp(-\sum_{\hat{a}} H(s,\hat{a})/|\mathcal{A}(s)|\sigma)}{1+\exp(-\sum_{\hat{a}} H(s,\hat{a})/|\mathcal{A}(s)|\sigma)} + (1-\lambda)\epsilon$$

- Baselines: ε-greedy, decaying ε, VDBE, and **ensemble-mechanics ε-greedy** (aka "**Stat Mech**")
  - Python 3.6.8
- Tests: **Blackjack** (stochastic) from homework; **Minefield** (nonstochastic) and **FrozenLake** (stochastic) from Ref. 9

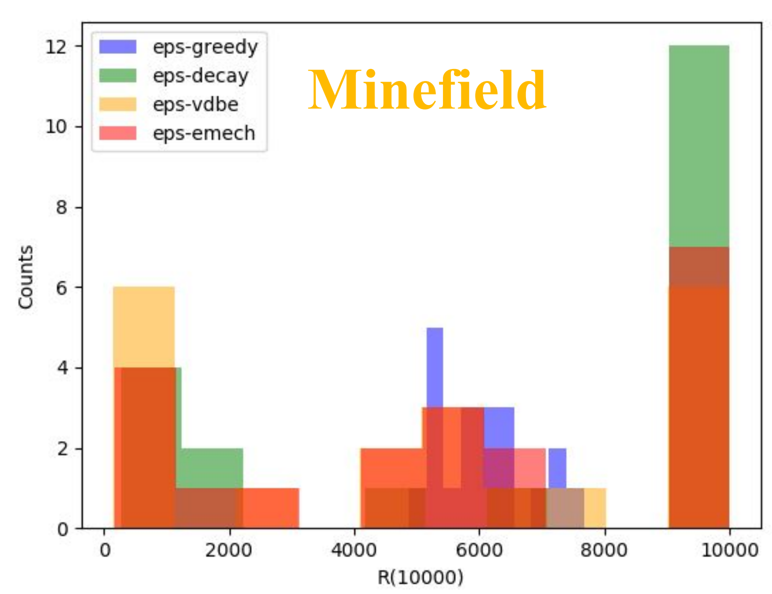
```
SLLL    S = start
LHLH    L = land, safe
LLLH    H = hole/mine
HLLE    E = end
```

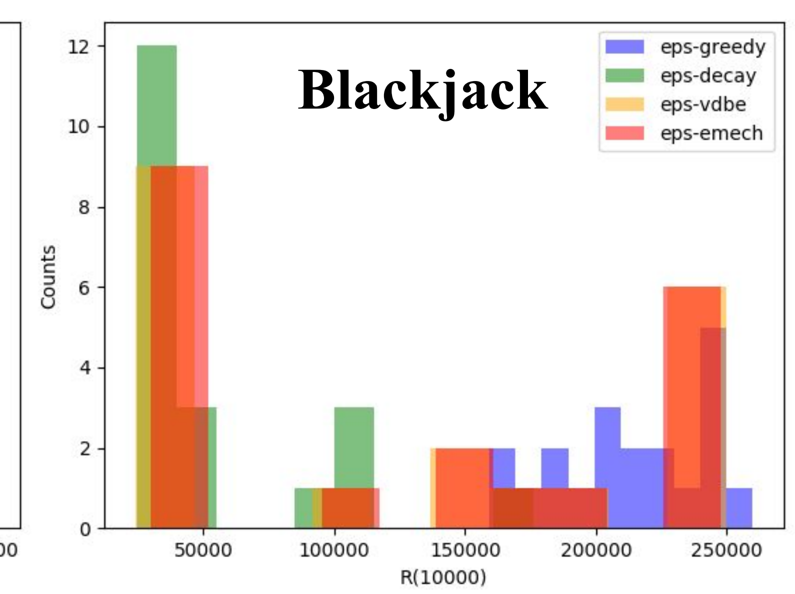
Abstract

- Reduce speed of convergence (regret) in RL algorithms
- Our algorithm: ε-greedy, but update ε as agent moves through environment according to statistical mechanics
- Compare with three existing RL algorithms
- No improvement but a better *a priori* hyperparameter distribution could help

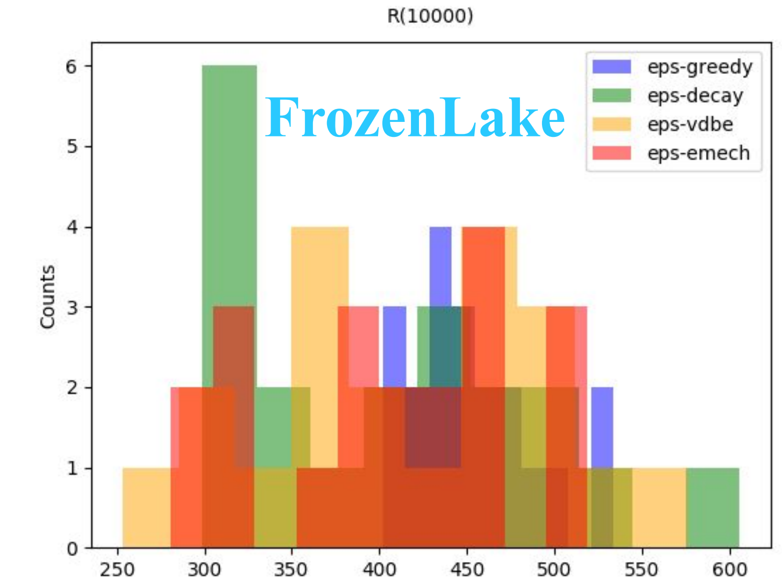
Results and Analysis

		ε-greedy	ε-decay	ε-VDBE	Stat Mech
Blackjack	Mean	213744	55601	127089	128443
$R(10^4)$	Std Error	6572	8898	20313	20115
Minefield	Mean	6089	6816	5350	5840
$R(10^4)$	Std Error	161	920	840	809
Frozen Lake	Mean	453	407	414	403
$W(10^4)$	Std Error	8.1	18.6	18.5	15.7

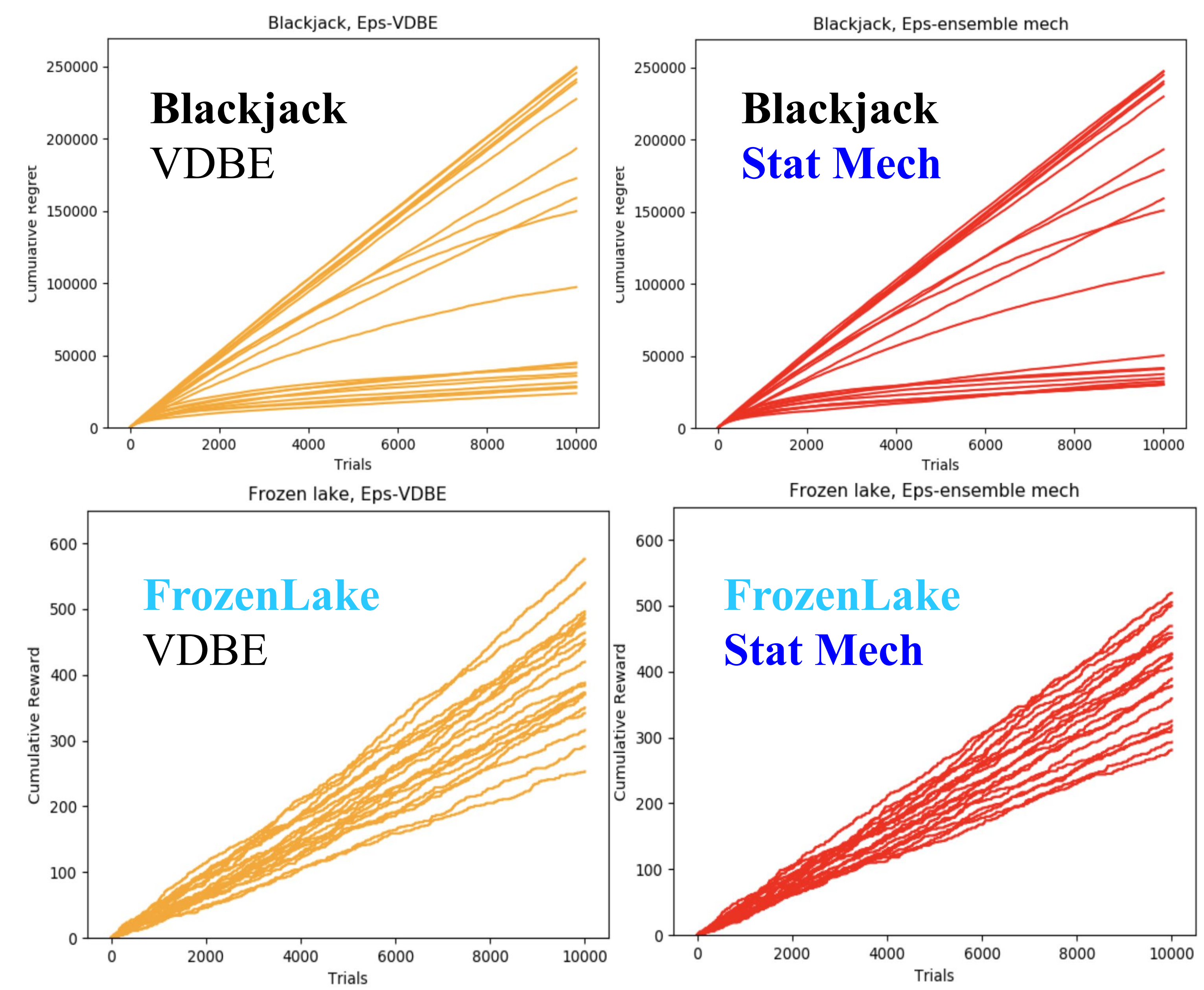




- **Blackjack**: high variance, average performance
- **Minefield**: high variance, decent performance
  - Only nonstochastic environment
- **FrozenLake**: low variance, bad performance
  - Use cumulative reward W (higher is better): $W(N) = \sum_{i=1}^{N} \sum_{t \in \tau_i} r(s_t, a_t)$



Results and Analysis (continued)



Conclusion and Future Work

- Our algorithm ("**Stat Mech**") performs comparable to VDBE in two environments
- VDBE performs slightly better in one environment
- Used same hyperparameter distribution (a convenient assumption that can be improved) for all tests
  - Future: improve hyperparameter distributions
- Adaptive algorithms underperform nonadaptive in 2/3 tests, possibly due to stochasticity, as noted in Ref. 8
  - Future: formalize stochasticity

References and Acknowledgements

1. Michel Tokic, etc. Ann. Conf. Artificial Intelligence, 2011, p. 335–346. Springer.
2. Michel Tokic. Ann. Conf. Artificial Intelligence, 2010, p. 203–210. Springer.
3. Peter Auer, etc. SIAM journal on computing, 32(1):48–77, 2002.
4. Jad Rahme, etc. arXiv preprint arXiv:1906.10228, 2019.
5. Zihan Zhang, etc. arXiv preprint arXiv:1906.05110, 2019.
6. Adithya Devraj, etc. arXiv preprint arXiv:1707.03770, 2017.
7. Chi Jin, etc. Adv. Neural Information Processing Systems, 2018, p. 4863–4873.
8. James Gupta, etc. Stanford University. CS234 lecture note.
9. Greg Brockman, etc. Openai gym. arXiv preprint arXiv:1606.01540, 2016.

We appreciate Zach Barnes for his advice and help.

Stanford University