An Introduction to Stochastic Multi-armed Bandits

Shivaram Kalyanakrishnan
shivaram@csa.iisc.ernet.in

Department of Computer Science and Automation
Indian Institute of Science

August 2014

Today's Talk

- What we will cover

Today's Talk

- What we will cover
  - Stochastic bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits
  - Dueling bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits
  - Dueling bandits
  - Contextual bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits
  - Dueling bandits
  - Contextual bandits
  - Mortal bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits
  - Dueling bandits
  - Contextual bandits
  - Mortal bandits
  - Sleeping bandits

Today's Talk

- What we will cover
  - Stochastic bandits
- What we will not cover
  - Adversarial bandits
  - Dueling bandits
  - Contextual bandits
  - Mortal bandits
  - Sleeping bandits
  - Real bandits

A Game

Coin 1

$\mathbb{P}\{\text{heads}\} = p_1$

Coin 2

$\mathbb{P}\{\text{heads}\} = p_2$

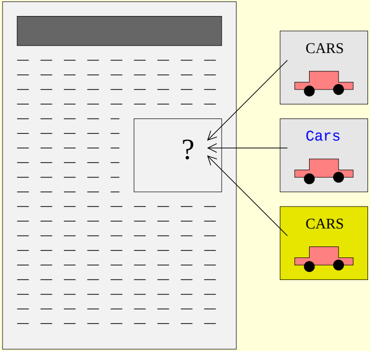
Coin 3

$\mathbb{P}\{\text{heads}\} = p_3$

- p_1, p_2, and p_3 are **unknown**.
- You are given a total of 20 tosses.
- Maximise the total number of heads!

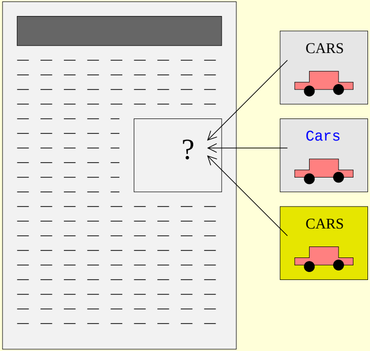
To Explore or to Exploit?

- On-line advertising: Template optimisation



To Explore or to Exploit?

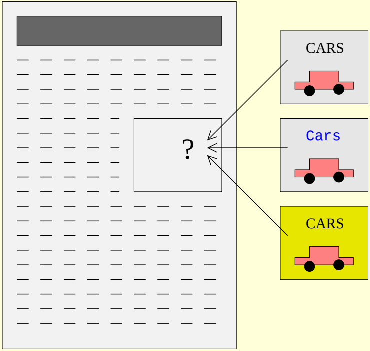
- On-line advertising: Template optimisation



- Clinical trials (Robbins, 1952)

To Explore or to Exploit?

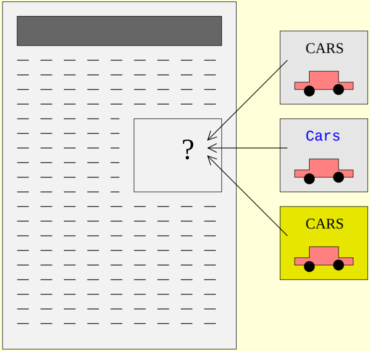
- On-line advertising: Template optimisation



- Clinical trials (Robbins, 1952)
- Packet routing in communication networks (Altman, 2002)

To Explore or to Exploit?

- On-line advertising: Template optimisation

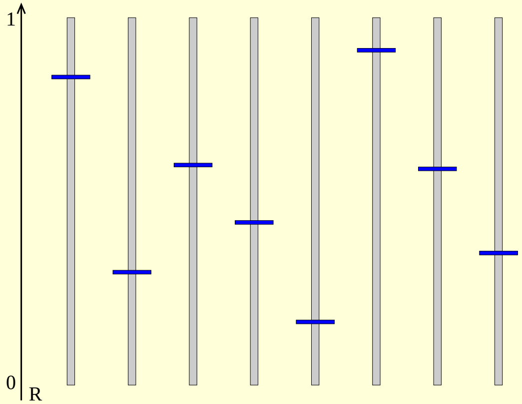


- Clinical trials (Robbins, 1952)
- Packet routing in communication networks (Altman, 2002)
- Game playing and reinforcement learning (Kocsis and Szepesvári, 2006)

Overview

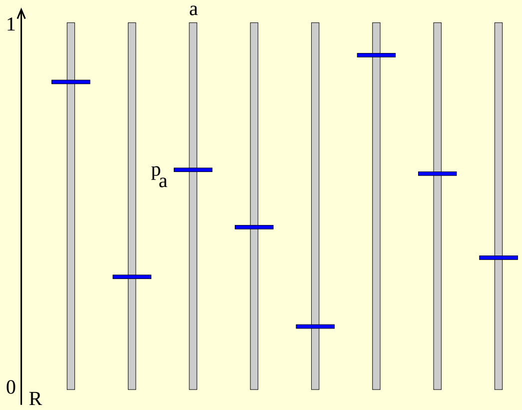
1. Problem definition
2. Two natural algorithms
3. Lower bound
4. Two improved algorithms
5. Conclusion

Stochastic Multi-armed Bandits



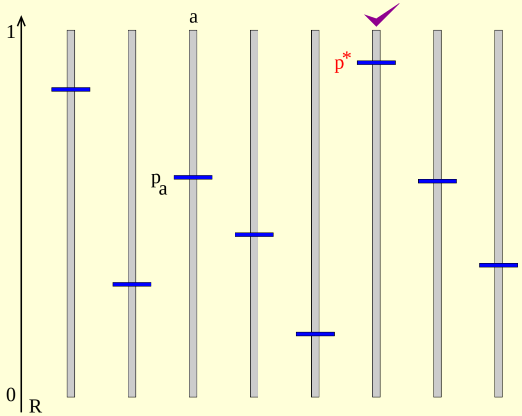
- *n* arms, each associated with a Bernoulli distribution.

Stochastic Multi-armed Bandits



- n arms, each associated with a Bernoulli distribution.
- Arm a has mean p_a.

Stochastic Multi-armed Bandits



- n arms, each associated with a Bernoulli distribution.
- Arm a has mean p_a.
- Highest mean is p^*.

One-armed Bandits

Regret Minimisation

- What does an **algorithm** do?

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.

- T is the total sampling budget, or the horizon.

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.

- T is the total sampling budget, or the horizon.

- The regret at time t is defined as $p^* - r^t$.

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.
- T is the total sampling budget, or the horizon.

- The regret at time t is defined as $p^* - r^t$.
- The cumulative regret over a run is $\sum_{t=1}^{T}(p^* - r^t)$.

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.
- T is the total sampling budget, or the horizon.

- The regret at time t is defined as $p^* - r^t$.
- The cumulative regret over a run is $\sum_{t=1}^{T}(p^* - r^t)$.
- The expected cumulative regret of the algorithm (or simply "regret") is

$$R_T = \mathbb{E}\left[\sum_{t=1}^{T}(p^* - r^t)\right] = Tp^* - \sum_{t=1}^{T}\mathbb{E}[r^t].$$

Regret Minimisation

- What does an **algorithm** do?

  For $t = 1, 2, 3, \ldots, T$:

  - Given the history $a^1, r^1, a^2, r^2, a^3, r^3, \ldots, a^{t-1}, r^{t-1}$,
  - Pick an arm a^t to sample, and
  - Obtain a reward r^t drawn from the distribution corresponding to arm a^t.
- T is the total sampling budget, or the horizon.

- The regret at time t is defined as $p^* - r^t$.
- The cumulative regret over a run is $\sum_{t=1}^{T}(p^* - r^t)$.
- The expected cumulative regret of the algorithm (or simply "regret") is

$$R_T = \mathbb{E}\left[\sum_{t=1}^{T}(p^* - r^t)\right] = Tp^* - \sum_{t=1}^{T}\mathbb{E}[r^t].$$

We desire an algorithm that minimises regret!

Overview

1. Problem definition
2. Two natural algorithms
3. Lower bound
4. Two improved algorithms
5. Conclusion

ϵ-Greedy Strategies

- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.

ϵ-Greedy Strategies

- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.

- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵ-Greedy Strategies

- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.

- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

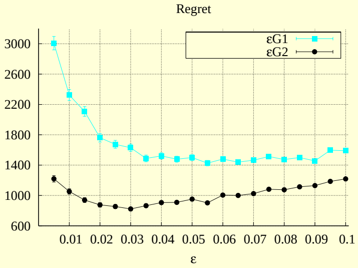
ϵ-Greedy Strategies

- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.
- ϵG2
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - If $t > \epsilon T$, sample an arm with the highest empirical mean.

- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵ-Greedy Strategies

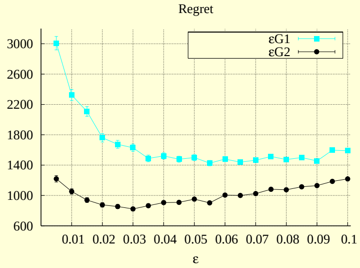
- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.
- ϵG2
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - If $t > \epsilon T$, sample an arm with the highest empirical mean.



- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵ-Greedy Strategies

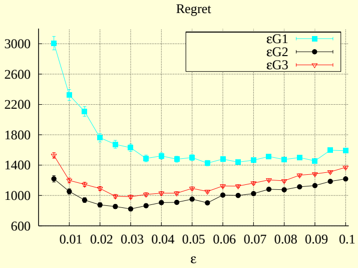
- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.
- ϵG2
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - If $t > \epsilon T$, sample an arm with the highest empirical mean.
- ϵG3 (Sutton and Barto, 1998; see Chapter 2.2)
  - With probability ϵ, sample an arm uniformly at random; with probability $1 - \epsilon$, sample an arm with the highest empirical mean.



- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵ-Greedy Strategies

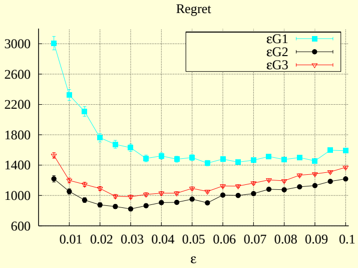
- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.
- ϵG2
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - If $t > \epsilon T$, sample an arm with the highest empirical mean.
- ϵG3 (Sutton and Barto, 1998; see Chapter 2.2)
  - With probability ϵ, sample an arm uniformly at random; with probability $1 - \epsilon$, sample an arm with the highest empirical mean.



- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵ-Greedy Strategies

- ϵG1 (parameter $\epsilon \in [0, 1]$ controls the amount of exploration)
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - At $t = \lfloor \epsilon T \rfloor$, identify a^{best}, an arm with the highest empirical mean.
  - If $t > \epsilon T$, sample a^{best}.
- ϵG2
  - If $t \leq \epsilon T$, sample an arm uniformly at random.
  - If $t > \epsilon T$, sample an arm with the highest empirical mean.
- ϵG3 (Sutton and Barto, 1998; see Chapter 2.2)
  - With probability ϵ, sample an arm uniformly at random; with probability $1 - \epsilon$, sample an arm with the highest empirical mean.



- Test instance I_1: $n = 20$; means = $0.01, 0.02, 0.03, \ldots, 0.2$; $T = 100,000$.

ϵG2 with $\epsilon = 0.03$ denoted ϵG^*. Regret of 822 ± 24 over a horizon of $100,000$.

Softmax Exploration

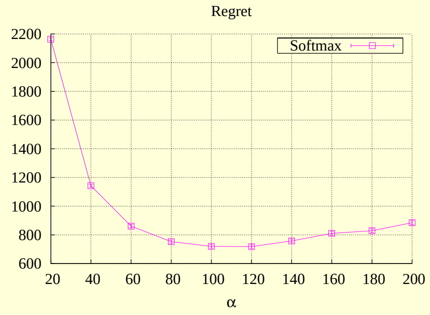
- Softmax (Sutton and Barto, 1998; see Chapter 2.3)
  - At time t, Sample arm a with probability proportional to $\exp\left(\frac{\alpha \hat{p}_a^t T}{t}\right)$.
- $\hat{p}_a^t$ the empirical mean of arm a.
- α a tunable parameter that controls exploration.
- One could "anneal" at rates different from $\frac{1}{t}$.

Softmax Exploration

- Softmax (Sutton and Barto, 1998; see Chapter 2.3)
  - At time t, Sample arm a with probability proportional to $\exp\left(\frac{\alpha \hat{p}_a^t T}{t}\right)$.
- $\hat{p}_a^t$ the empirical mean of arm a.
- α a tunable parameter that controls exploration.
- One could "anneal" at rates different from $\frac{1}{t}$.

Softmax Exploration

- Softmax (Sutton and Barto, 1998; see Chapter 2.3)
  - At time t, Sample arm a with probability proportional to $\exp\left(\frac{\alpha \hat{p}_a^t T}{t}\right)$.
- $\hat{p}_a^t$ the empirical mean of arm a.
- α a tunable parameter that controls exploration.
- One could "anneal" at rates different from $\frac{1}{t}$.



Softmax with $\alpha = 100$ denoted Softmax*. Regret of 720 ± 13 on I_1 over a horizon of $T = 100,000$.

Overview

1. Problem definition
2. Two natural algorithms
3. Lower bound
4. Two improved algorithms
5. Conclusion

A Lower Bound on Regret

Paraphrasing Lai and Robbins (1985; see Theorem 2).

> Let $\mathcal{A}$ be an algorithm such that for every bandit instance I and for every $a > 0$, as $T \to \infty$:
> $$R_T(\mathcal{A}, I) = o(T^a).$$

A Lower Bound on Regret

Paraphrasing Lai and Robbins (1985; see Theorem 2).

Let $\mathcal{A}$ be an algorithm such that for every bandit instance I and for every $a > 0$, as $T \to \infty$:

$$R_T(\mathcal{A}, I) = o(T^a).$$

Then, for every bandit instance I, as $T \to \infty$:

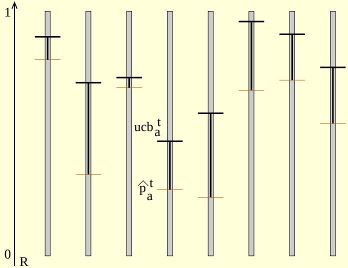
$$R_T(\mathcal{A}, I) \geq \left(\sum_{a : p_a(I) \neq p^*(I)} \frac{p^*(I) - p_a(I)}{KL(p_a(I), p^*(I))} \right) \log(T).$$

Overview

1. Problem definition
2. Two natural algorithms
3. Lower bound
4. Two improved algorithms
5. Conclusion

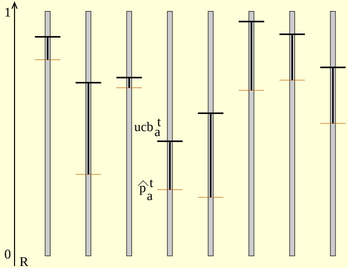
Upper Confidence Bounds

- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



Upper Confidence Bounds

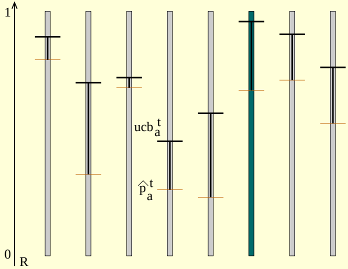
- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



  - Sample an arm a for which ucb_a^t is maximal.

Upper Confidence Bounds

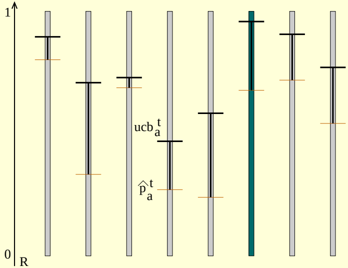
- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



  - Sample an arm a for which ucb_a^t is maximal.

Upper Confidence Bounds

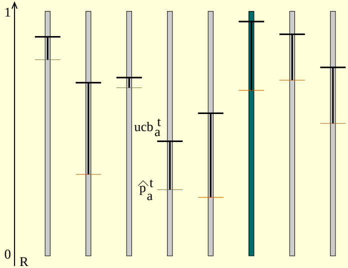
- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



  - Sample an arm a for which ucb_a^t is maximal.
- Achieves regret of $O\left(\sum_{a: p_a \neq p^*} \frac{1}{p^* - p_a} \log(T)\right)$: optimal dependence on T.

Upper Confidence Bounds

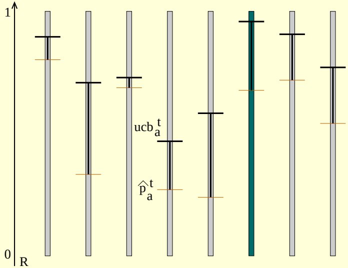
- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



  - Sample an arm a for which ucb_a^t is maximal.
- Achieves regret of $O\left(\sum_{a: p_a \neq p^*} \frac{1}{p^* - p_a} \log(T)\right)$: optimal dependence on T.
- KL-UCB (Garivier and Cappé, 2011) yields regret $O\left(\sum_{a: p_a \neq p^*} \frac{p^* - p_a}{KL(p_a, p^*)} \log(T)\right)$, matching the lower bound from Lai and Robbins (1985).

Upper Confidence Bounds

- UCB (Auer et al., 2002a)
  - At time t, for every arm a, define $\text{ucb}_a^t = \hat{p}_a^t + \sqrt{\frac{2\ln(t)}{u_a^t}}$.
  - u_a^t the number of times a has been sampled at time t.



  - Sample an arm a for which ucb_a^t is maximal.
- Achieves regret of $O\left(\sum_{a:p_a \neq p^*} \frac{1}{p^* - p_a} \log(T)\right)$: optimal dependence on T.
- KL-UCB (Garivier and Cappé, 2011) yields regret $O\left(\sum_{a:p_a \neq p^*} \frac{p^* - p_a}{KL(p_a, p^*)} \log(T)\right)$, matching the lower bound from Lai and Robbins (1985).

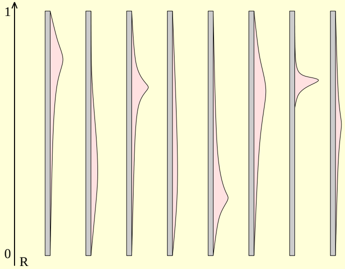
Regret on instance I_1 (with $T = 100,000$)–UCB: 1168 ± 16; KL-UCB: 738 ± 18.

Thompson Sampling

- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).

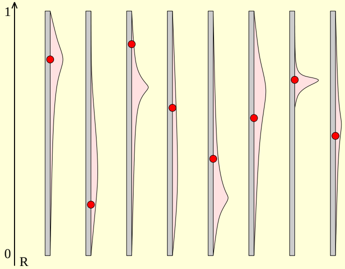
Thompson Sampling

- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).
  - $Beta(s_a^t + 1, f_a^t + 1)$ represents a “belief” about the true mean of arm a.
  - Mean = $\frac{s_a^t+1}{s_a^t+f_a^t+2}$; variance = $\frac{(s_a^t+1)(f_a^t+1)}{(s_a^t+f_a^t+2)^2(s_a^t+f_a^t+3)}$.



Thompson Sampling

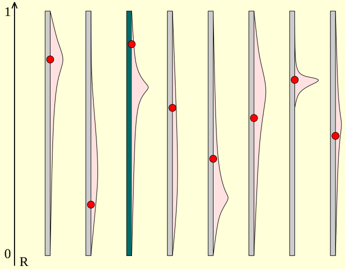
- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).
  - $Beta(s_a^t + 1, f_a^t + 1)$ represents a "belief" about the true mean of arm a.
  - Mean = $\frac{s_a^t+1}{s_a^t+f_a^t+2}$; variance = $\frac{(s_a^t+1)(f_a^t+1)}{(s_a^t+f_a^t+2)^2(s_a^t+f_a^t+3)}$.



  - Computational step: For every arm a, draw a sample $x_a^t \sim Beta(s_a^t + 1, f_a^t + 1)$.
  - Sampling step: Sample an arm a for which x_a^t is maximal.

Thompson Sampling

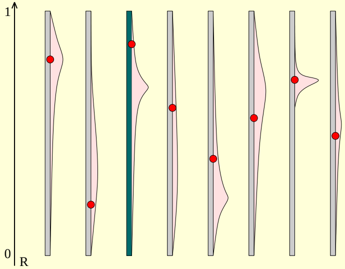
- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).
  - $Beta(s_a^t + 1, f_a^t + 1)$ represents a "belief" about the true mean of arm a.
  - Mean = $\frac{s_a^t+1}{s_a^t+f_a^t+2}$; variance = $\frac{(s_a^t+1)(f_a^t+1)}{(s_a^t+f_a^t+2)^2(s_a^t+f_a^t+3)}$.



  - Computational step: For every arm a, draw a sample $x_a^t \sim Beta(s_a^t + 1, f_a^t + 1)$.
  - Sampling step: Sample an arm a for which x_a^t is maximal.

Thompson Sampling

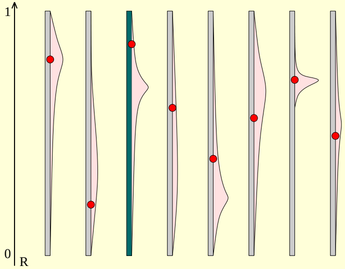
- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).
  - $Beta(s_a^t + 1, f_a^t + 1)$ represents a "belief" about the true mean of arm a.
  - Mean = $\frac{s_a^t+1}{s_a^t+f_a^t+2}$; variance = $\frac{(s_a^t+1)(f_a^t+1)}{(s_a^t+f_a^t+2)^2(s_a^t+f_a^t+3)}$.



  - Computational step: For every arm a, draw a sample $x_a^t \sim Beta(s_a^t + 1, f_a^t + 1)$.
  - Sampling step: Sample an arm a for which x_a^t is maximal.
- Achieves optimal regret (Kaufmann et al., 2012); is excellent in practice (Chapelle and Li, 2011).

Thompson Sampling

- Thompson (Thompson, 1933)
  - At time t, let arm a have s_a^t successes (ones) and f_a^t failures (zeroes).
  - $Beta(s_a^t + 1, f_a^t + 1)$ represents a "belief" about the true mean of arm a.
  - Mean = $\frac{s_a^t+1}{s_a^t+f_a^t+2}$; variance = $\frac{(s_a^t+1)(f_a^t+1)}{(s_a^t+f_a^t+2)^2(s_a^t+f_a^t+3)}$.



  - Computational step: For every arm a, draw a sample $x_a^t \sim Beta(s_a^t + 1, f_a^t + 1)$.
  - Sampling step: Sample an arm a for which x_a^t is maximal.
- Achieves optimal regret (Kaufmann et al., 2012); is excellent in practice (Chapelle and Li, 2011).

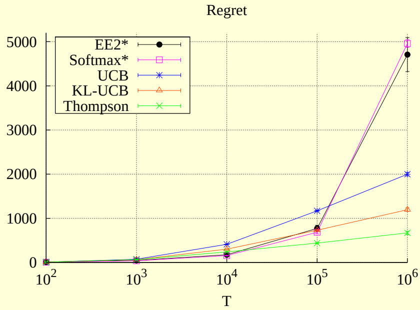
On instance I_1 (with $T = 100,000$), regret is 463 ± 18.

Consolidated Results on Instance I_1

Method	**Regret at $\mathbf{T = 100,000}$**
ϵG*	822 ± 24
Softmax*	720 ± 13
UCB	1168 ± 16
KL-UCB	738 ± 16
Thompson	463 ± 18

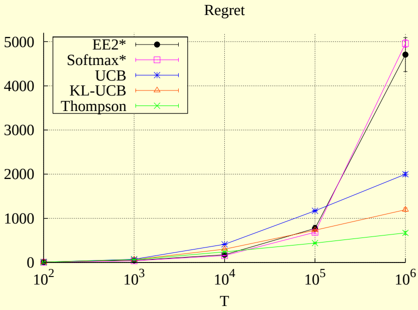
Consolidated Results on Instance I_1

Method	Regret at T $= 100,000$
ϵG*	822 ± 24
Softmax*	720 ± 13
UCB	1168 ± 16
KL-UCB	738 ± 16
Thompson	463 ± 18



Consolidated Results on Instance I_1

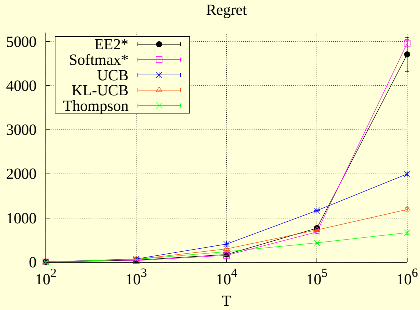
Method	**Regret at T = 100, 000**
ϵG*	822 ± 24
Softmax*	720 ± 13
UCB	1168 ± 16
KL-UCB	738 ± 16
Thompson	463 ± 18



Use Thompson Sampling!

Consolidated Results on Instance I_1

Method	Regret at T $= 100,000$
ϵG*	822 ± 24
Softmax*	720 ± 13
UCB	1168 ± 16
KL-UCB	738 ± 16
Thompson	463 ± 18



Use Thompson Sampling!

Principle: "Optimism in the face of uncertainty."

Overview

Discussion

- **Challenges and extensions**

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.
  - On-line updates not feasible; batch updating needed.

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.
  - On-line updates not feasible; batch updating needed.
  - Rewards are delayed.

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.
  - On-line updates not feasible; batch updating needed.
  - Rewards are delayed.
  - Arms might be *dependent*; “context” can be modeled (Li et al., 2010).

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.
  - On-line updates not feasible; batch updating needed.
  - Rewards are delayed.
  - Arms might be *dependent*; “context” can be modeled (Li et al., 2010).
  - Nonstationary rewards; adversarial modeling possible (Auer et al., 2002b).

Discussion

- **Challenges and extensions**
  - Set of arms can change over time.
  - On-line updates not feasible; batch updating needed.
  - Rewards are delayed.
  - Arms might be *dependent*; "context" can be modeled (Li et al., 2010).
  - Nonstationary rewards; adversarial modeling possible (Auer et al., 2002b).
- **Summary**
  - Adaptive sampling of options, based on stochastic feedback, to maximise total reward.
  - Well-studied problem with long history.
  - Thompson Sampling is an essentially optimal algorithm.
  - Modeling assumptions typically violated only slightly in practice.

References

W. R. Thompson, 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3-4):285–294, 1933.

Herbert Robbins, 1952. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5): 527–535, 1952.

T. L. Lai and Herbert Robbins, 1985. Asymptotically Efficient Adaptive Allocation Rules. *Advances in Applied Mathematics*, 6(1): 4–22, 1985.

Richard S. Sutton and Andrew G. Barto, 1998. Reinforcement Learning: An Introduction. MIT Press, 1998.

Eitan Altman, 2002. Applications of Markov Decision Processes in Communication Networks. *Handbook of Markov Decision Processes: International Series in Operations Research & Management Science*, 40: 489–536, Springer, 2002.

Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer, 2002a. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine Learning*, 47(2–3):235–256, 2002.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire, 2002b. The Nonstochastic Multiarmed Bandit Problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

Levente Kocsis and Csaba Szepesvári, 2006. Bandit Based Monte-Carlo Planning. In *Proceedings of the Seventeenth European Conference on Machine Learning (ECML 2006)*, pp. 282–293, Springer, 2006.

Lihong Li, Wei Chu, John Langford, and Robert E. Schapire, 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the Nineteenth International Conference on the World Wide Web (WWW 2010)*, pp. 661–670, ACM, 2010.

Olivier Chapelle and Lihong Li, 2011. An Empirical Evaluation of Thompson Sampling. In *Advances in Neural Information Processing Systems 24 (NIPS 2011)*, pp. 2249–2257, Curran Associates, 2011.

Aurélien Garivier and Olivier Cappé. The KL-UCB Algorithm for Bounded Stochastic Bandits and Beyond. *Journal of Machine Learning Research (Workshop and Conference Proceedings)*, 19: 359–376, 2011.

Emilie Kaufmann, Nathaniel Korda, and Rémi Munos, 2012. Thompson Sampling: An Asymptotically Optimal Finite-Time Analysis. In *Proceedings of the Twenty-third International Conference on Algorithmic Learning Theory (ALT 2012)*, pp. 199–213, Springer, 2012.

Upcoming Talks

- **PAC Subset Selection in Stochastic Multi-armed Bandits**
  - 4.00 p.m. – 5.30 p.m.; Thursday, August 14, 2014; CSA 254.
- **RoboCup: A Grand Challenge for AI**
  - 4.00 p.m. – 5.00 p.m.; Wednesday, August 20, 2014; CSA 254.
- **An Introduction to Reinforcement Learning**
  - 4.00 p.m. – 5.15 p.m.; Wednesday, August 27, 2014; CSA 254.

Upcoming Talks

- **PAC Subset Selection in Stochastic Multi-armed Bandits**
  - 4.00 p.m. – 5.30 p.m.; Thursday, August 14, 2014; CSA 254.
- **RoboCup: A Grand Challenge for AI**
  - 4.00 p.m. – 5.00 p.m.; Wednesday, August 20, 2014; CSA 254.
- **An Introduction to Reinforcement Learning**
  - 4.00 p.m. – 5.15 p.m.; Wednesday, August 27, 2014; CSA 254.

Thank you!