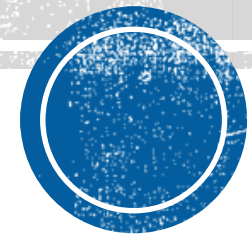


A tutorial on **Bandits and Reinforcement Learning**



Shipra Agrawal
Columbia University

Based on some joint works with Randy Jia, Steven Yin, Assaf Zeevi

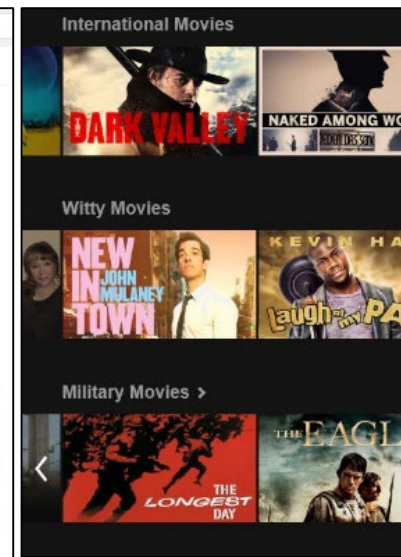
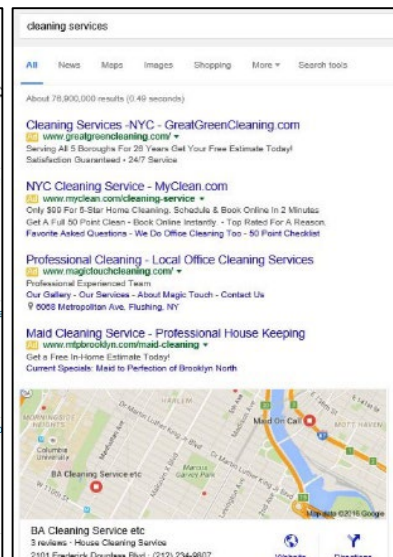
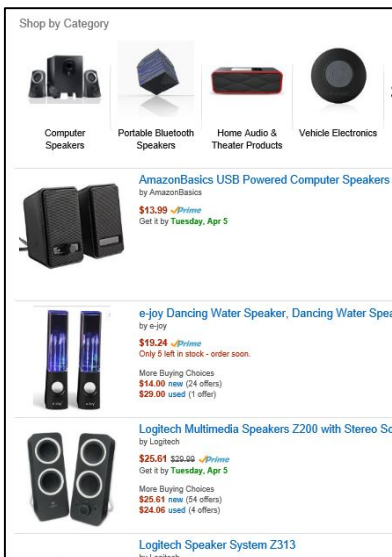
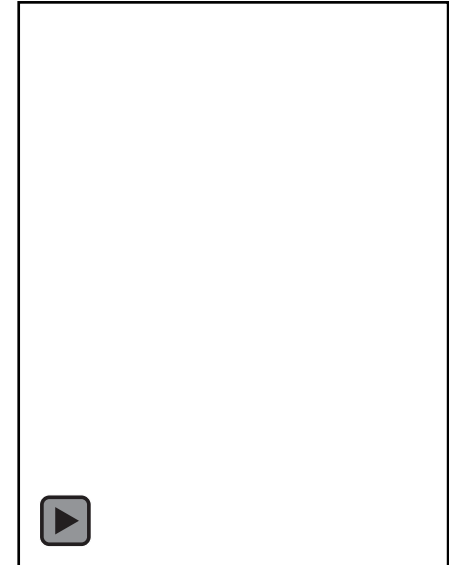
Overview of the tutorial

- The Multi-Armed Bandit (MAB) problem
 - Some algorithmic techniques (UCB, TS)
 - Handling large number arms with linear and convex models
- The Reinforcement Learning problem (RL)
 - Limits on scalability without structure
- Using bandit techniques for **structured** RL problems
 - Problem #1: Inventory management
 - Problem #2: Dynamic Pricing

Learning from sequential interactions

Tradeoff between

- information and rewards
- learning and optimization
- Exploration and exploitation



Robotic Manipulation and Mobility

Stochastic multi-armed bandit problem

- Online decisions
 - At every time step $t = 1, \dots, T$, pull one arm out of N arms
- Stochastic feedback
 - For each arm i , reward is generated **i.i.d.** from a **fixed but unknown distribution** support $[0,1]$, mean μ_i
- Bandit feedback
 - Only the reward of the pulled arm can be observed
- Minimize **regret** compared to the best arm

$$E[\sum_{t=1}^T (\mu^* - \mu_{i_t})] \quad \text{where } \mu^* = \max_j \mu_j$$



The need for exploration

- Two arms **black** and **red**
 - Random rewards with unknown mean $\mu_1 = 1.1$, $\mu_2 = 1$
 - Optimal expected reward in T time steps is $1.1 \times T$
- Exploit only strategy: use the current best estimate (MLE/empirical mean) of the unknown means to pick arms
- Initial few trials can mislead into playing red action forever

1.1, 1, 0.2,

1, 1, 1, 1, 1, 1, 1,

- Expected regret in T steps is close to $0.1 \times T$

Exploration-Exploitation tradeoff

- Exploitation: play the empirical mean reward maximizer
- Exploration: play less explored actions to ensure empirical estimates converge

UCB algorithm [Auer 2002]

- Empirical mean at time t for arm i

$$\hat{\mu}_{i,t} = \frac{\sum_{s=1: I_s=i}^t r_s}{n_{i,t}}$$

- Upper confidence bound (UCB)

$$UCB_{i,t} = \hat{\mu}_{i,t} + \sqrt{\frac{4 \ln t}{n_{i,t}}}$$

- Optimism:

$$UCB_{i,t} \geq \mu_i \text{ w.h.p.}$$

- UCB Algorithm

- At each time step t , play the with best optimistic estimates

$$i_t = \arg \max_i UCB_{t,i}$$

Intuition for regret analysis

- Key property : Optimism $UCB_{i,t} > \mu_i$ for all arms i w.h.p.

$$\text{Regret} = \sum_{t=1}^T (\mu_{i^*} - \mu_{i_t}) \leq \sum_t (UCB_{i^*,t} - \mu_{i_t})$$

- Since the arm with highest UCB was played at time t

$$\text{Regret} \leq \sum_t (UCB_{i_t,t} - \mu_{i_t})$$

- Regret is bounded by error in estimates of the played arms

$$\text{Regret} \approx \sum_t \frac{1}{\sqrt{n_{i,t}}} = \sum_i \left(1 + \frac{1}{\sqrt{2}} + \frac{1}{\sqrt{3}} + \dots + \frac{1}{\sqrt{n_{i,T}}} \right) \leq 2\sqrt{NT}$$

- Error improves every time an arm is played, gives sublinear bound on regret

Thompson Sampling [Thompson, 1933]

- Natural and Efficient heuristic
- Maintain belief about parameters (e.g., mean reward) of each arm
- Observe feedback, update belief of pulled arm i in Bayesian manner
- Pull each arm with its posterior probability of being the best arm
 - NOT same as choosing the arm that is most likely to be best

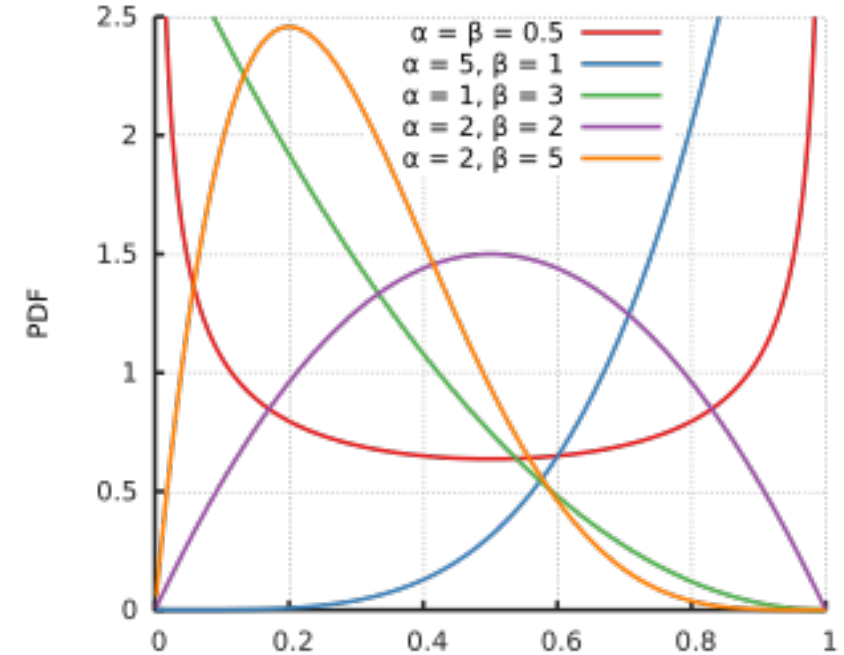


Bernoulli rewards, Beta priors

Uniform distribution $Beta(1,1)$

$Beta(\alpha, \beta)$ prior $\Rightarrow$ Posterior

- $Beta(\alpha + 1, \beta)$ if you observe 1
- $Beta(\alpha, \beta + 1)$ if you observe 0



Start with $Beta(1,1)$ prior belief for every arm

In round t ,

- For every arm i , sample $\theta_{i,t}$ independently from posterior $Beta(S_{i,t} + 1, F_{i,t} + 1)$
- Play arm $i_t = \max_i \theta_{i,t}$
- Observe reward and update the Beta posterior for arm i_t



Gaussian prior, Gaussian likelihood

Standard normal prior $N(0,1)$

Gaussian likelihood $N(\mu, 1)$ of reward

Posterior after n independent observations: $N\left(\hat{\mu}, \frac{1}{n+1}\right)$

- $\hat{\mu}$ is the empirical mean

Start with $N(0, v^2)$ prior belief for every arm

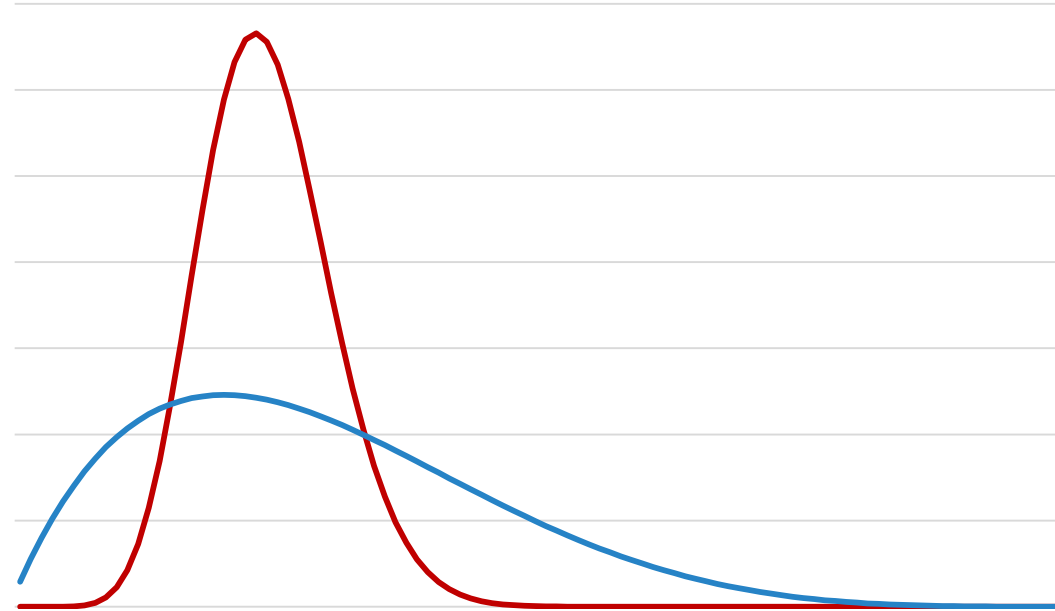
In round t ,

- For every arm i , sample $\theta_{i,t}$ independently from posterior $N\left(\hat{\mu}_i, \frac{v^2}{n_i+1}\right)$
- Play arm $i_t = \max_i \theta_{i,t}$
- Observe reward and update empirical mean $\hat{\mu}_i$ and number of plays n_i for arm i_t



Intuition

- Maintain Bayesian posteriors for unknown parameters
- With more trials posteriors concentrate on the true parameters
 - Mode captures MLE: enables exploitation
- Less trials means more uncertainty in estimates
 - Spread/variance captures uncertainty: enables exploration



Regret bounds

Optimal instance-dependent bounds for Bernoulli rewards

- $\text{Regret}(T) \leq \ln(T)(1 + \epsilon) \sum_i \frac{\Delta_i}{KL(\mu^* || \mu_i)} + O\left(\frac{N}{\epsilon^2}\right)$
 - [A. and Goyal 2012, 2013, Kaufmann et al. 2013]
 - Matches *asymptotic instance wise lower bound* [Lai Robbins 1985]

Arbitrary bounded reward distribution (Beta and Gaussian priors)

- $\text{Regret}(T) \leq O(\ln(T) \sum_i \frac{1}{\Delta_i})$ [A. and Goyal 2013]

Instance-independent bounds (Beta and Gaussian priors)

- $\text{Regret}(T) \leq O(\sqrt{NT \ln T})$ [A. and Goyal 2013]
 - Lower bound $\Omega(\sqrt{NT})$

Prior and likelihood mismatch allowed! – worst case regret bounds



Lower bounds

- [Lai and Robbins 1985](#) [Informal] For *any* given instance of the MAB problem, any “reasonable algorithm” will play a suboptimal arm at least $\Omega(N \log(T))$ times for large T
- Worst case bound: For every algorithm, there exists an instance with $\Omega(\sqrt{NT})$ regret
- Note the linear/sublinear dependence on number of arms

How to scale? Handling large number of arms

- Many practical problems involve large number of arms
 - Products to recommend
 - Ads to display
- Utilize similarity? Learning about one product informs about similar products
- Supervised learning based models
 - Each arm i represented by a (feature) vector x_i dimension d
 - Arms with similar features have similar rewards

Linear and convex bandits

Parametric models mapping features to outcome

- Linear bandit (linear regression): mean reward is linear in x_i

$$\mu_i = E[r_t | i] = x_i \cdot \theta$$

- Convex bandit: mean reward is concave in x

$$\mu_i = E[r_t | i] = f(x_i, \theta)$$

Or mean cost is convex in x

- Features of each arm are known. Only need to learn the unknown parameter θ of dimension d

Some algorithms

Linear bandits

- LinUCB [Auer 2002] Regret bound $\tilde{O}(d\sqrt{T})$
- Thompson Sampling [A. and Goyal 2013] Regret bound $\tilde{O}(d^{3/2}\sqrt{T})$
- Worst case lower bound on regret of any algorithm $\Omega(d\sqrt{T})$

Convex bandits

- Successive elimination type algorithm [Agarwal, Foster, Hsu, Kakade, Rakhlin 2011]
Regret bound $\tilde{O}(\text{poly}(d)\sqrt{T})$
 - $O(\sqrt{T})$ for $d = 1$
- Several more recent works with better dependence on dimension.

Linear and Convex bandits

$$\mu_i = E[r_t | i] = f(x_i, \theta)$$

- Learning complexity depends on dimension of feature/parameter vector and not the number of arms
- Continuous but structured space of arms can be more efficient to learn than discrete

Talk outline

- ✓ The Multi-Armed Bandit (MAB) problem
 - Some algorithmic techniques (UCB, TS)
 - Handling large number arms with linear and convex models
- **The Reinforcement Learning problem (RL)**
 - Limits on scalability without structure
- Using bandit techniques for **structured** RL problems
 - Problem #1: Inventory management
 - Problem #2: Dynamic Pricing

The reinforcement learning problem

- Response to pulling an arm (action) depends on the **current state** of the system
- Rounds $t = 1, \dots, T$
 - *Observe* **current state**, take an action, observe reward and **new state**
- Solution concept: Optimal **policy**
 - Mapping from state to action
- Need to learn reward distributions **and state transition distribution** to find optimal decision making policy
- Applications: inventory management, autonomous vehicle control, robot navigation, personalized medical treatments

The reinforcement learning problem

Sequential decision making: Rounds $t = 1, \dots, T$.

In round t ,

- observe state s_t
- take action $a_t = \pi_t(s_t)$,
- observe reward $E[r_t] = \mu_{s_t, a_t}$, and next state s_{t+1} with probability $\mathbf{P}_{s_t, a_t}(s_{t+1})$

Goal: minimize regret

$$\text{Regret}(M, T) = \sum_{t=1}^T \mu(s_t^*, a_t^*) - \sum_{t=1}^T \mu(s_t, a_t)$$

$(s_t^*, a_t^*, t = 1, \dots, T)$ is the trajectory of best single stationary policy π^*

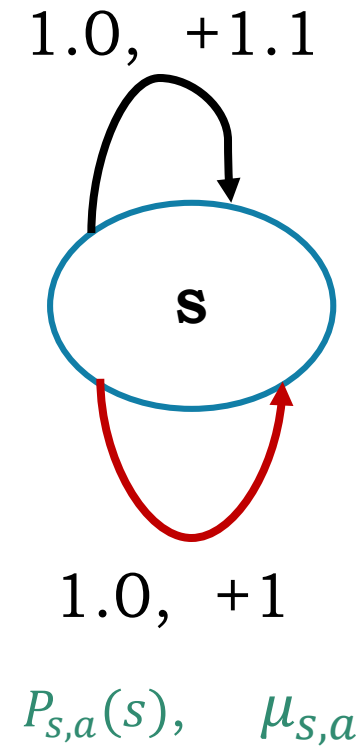
The need for exploration

Recall: Multi-armed bandit problem

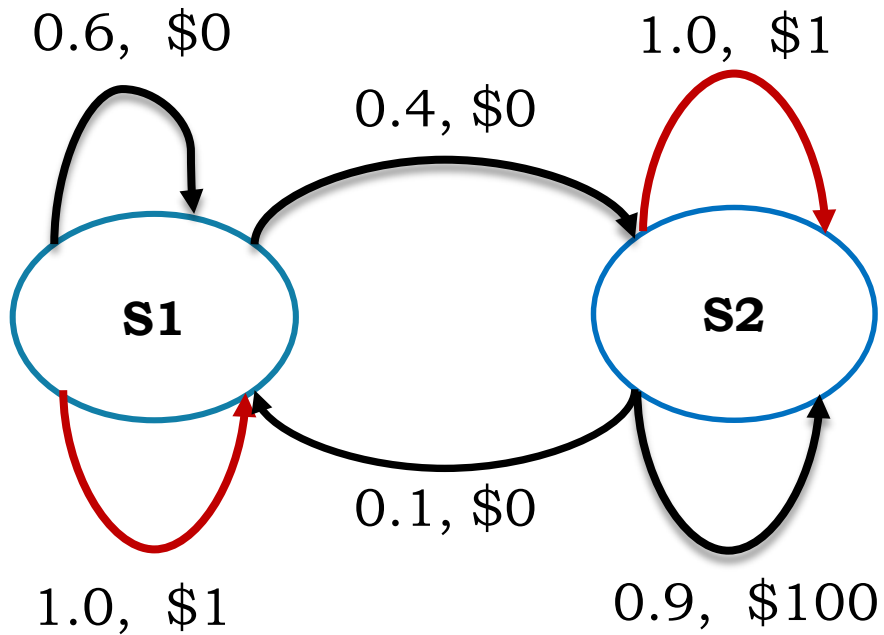
- Single state MDP
- Solution concept: optimal action
- Uncertainty in rewards
 - Random rewards with unknown mean μ_1, μ_2
- Exploit only: use the current best estimate (MLE/empirical mean) of unknown mean to pick arms
 - Initial few trials can mislead into playing red action forever

1.1, 1, 0.2,

1, 1, 1, 1, 1, 1, 1,



The need for exploration



- Uncertainty in rewards, state transitions
- Unknown reward distribution, transition probabilities
- Exploration-exploitation:
 - Explore actions/states/policies, learn reward distributions and transition model
 - Exploit the (seemingly) best policy

Summary of *recent* work: episodic setting

T is total number of steps, H steps per episode

Optimistic Value iteration [Azar, Osband, Munos, 2017]

- Model based algorithm
- Worst-case regret bound $\tilde{O}(H\sqrt{SAT})$

Optimistic Q-learning [Jin et al. 2018]

- Worst-case regret bound $\tilde{O}(\sqrt{H^3SAT})$

Posterior sampling known prior setting [Osband, Russo, and Van Roy 2013, Osband and Van Roy, 2016, 2017]

- Model based algorithm
- **Bayesian regret** bound of $\tilde{O}(H\sqrt{SAT})$

Summary of recent work: non-episodic setting

Communicating MDP diameter D , optimal policy is stationary

Upper confidence bound based algorithms [Jaksch, Ortner, Auer, 2010] [Bartlett, Tewari, 2012]

- Worst-case regret bound $\tilde{O}(DS\sqrt{AT})$

Optimistic Posterior Sampling [A. Jia 2017]

- Worst-case regret bound $\tilde{O}(DS\sqrt{AT})$

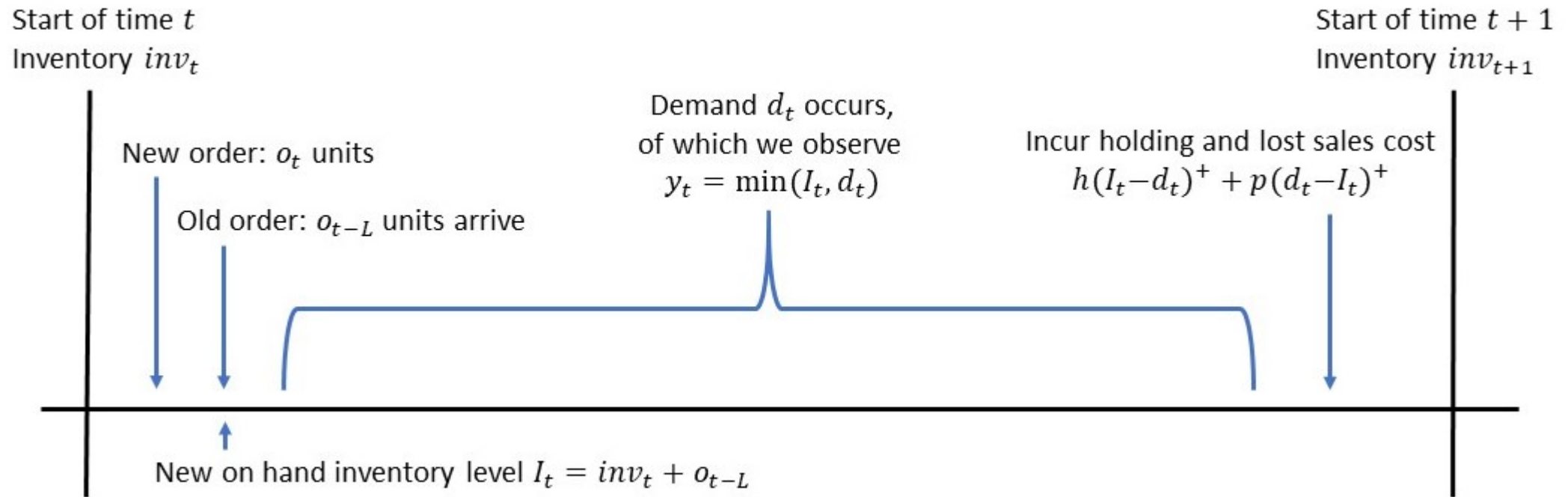
Lower bound $\Omega(\sqrt{DSAT})$

- **Note the linear/sublinear dependence on number of states and actions**
 - Can we do better for large but continuous structured state spaces?

Talk outline

- ✓ The Multi-Armed Bandit (MAB) problem
 - Some algorithmic techniques (UCB, TS)
 - Handling large number arms with linear and convex models
- ✓ The Reinforcement Learning problem (RL)
 - Limits on scalability without structure
- **Using bandit techniques for structured RL problems**
 - Problem #1: Inventory management [A. and Jia 2019]
 - Problem #2: Dynamic Pricing [A., Yin, and Zeevi 2021]

#1: Inventory control problem



Censored demand, lost sales, lead times



Inventory control problem: learning an MDP

In each round $t = 1, \dots, T$

- Observe inventory I_t , past $L - 1$ orders $(o_{t-L+1}, \dots, o_{t-1})$ [L-dim state]
- Decide new order o_t [1-dim action]
- Observe sales $y_t = \min(d_t, I_t)$ where $d_t \sim F$ [censored demand]
- Incur cost $\bar{C}_t = h(I_t - y_t) + p(\textcolor{red}{d}_t - y_t)$
- New inventory $I_{t+1} = I_t - y_t + o_{t-L+1}$ past orders $(o_{t-L+2}, \dots, o_t)$ [next state]

➡ Unknown demand distribution F

Learning a (continuous state and action) MDP



Challenges and relaxation

Positive lead time: past L orders are relevant,

- Number of states are exponential in L
- Makes the optimization problem difficult even under known demand distribution

Relaxed Goal

- Minimize regret against best in a benchmark class of policies

$$Regret(T) = E\left[\sum_{t=1}^T C_t \mid s_0\right] - \min_{\pi \in \Pi} E\left[\sum_{t=1}^T C_t^\pi \mid s_0\right]$$

- Interesting benchmark class Π : **base-stock policies**



Base-stock policies

- Base-stock policy π^x : Order the minimum amount required to bring new inventory position to base-stock level x

Given on hand inventory I_t , and outstanding orders $o_{t-1}, o_{t-2}, \dots, o_{t-L+1}$, order

$$o_t = \left(x - \sum_{i=1}^{L-1} o_{t-i} - I_t \right)^+$$

- Some attractive properties [Janakiraman, Roundy 2004]
 - Asymptotic Optimality
 - **Convexity of average cost:** Assume demand distribution F is such that $F(0) > 0$. Then, asymptotic average cost $g(x)$ under policy π^x is independent of the starting state and convex in x .



Thinking like a convex bandit

- In stochastic convex bandit, play $x_t \in R$, observe cost $c_t = g(x_t) + \text{noise}$
 - Minimize Regret $= \sum_{t=1}^T c_t - T \left(\min_{x \in [0, U]} g(x) \right)$
 - **Immediate cost** function $g(x)$ is unknown but convex function in x
- Inventory control: Play a policy π^{x_t} , observe sales $y_t = \min(d_t, I_t)$ (depends on state)
 - Minimize $E \left[\sum_{t=1}^T C_t \right] - T \left(\min_{x \in [0, U]} g(x) \right)$
 - **Long-run average cost** $g(x)$ is convex in x
- We need to play a single π^x longer to see long-run average...
 - use stochastic convex bandit with slow changing x



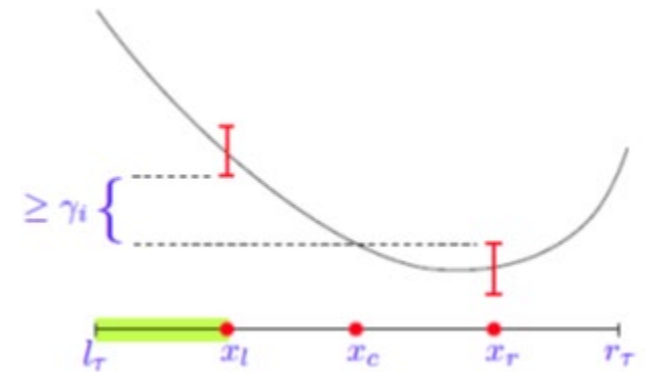
Stochastic convex bandit algorithm

[Agarwal, Foster, Hsu, Kakade, Rakhlin 2011]

- Successive elimination type algorithm for 1-d convex bandit to bound regret

$$\sum_{t=1}^T c_t - T \left(\min_x g(x) \right) \leq \tilde{O}(\sqrt{T})$$

- Epoch setup: Changes decision slowly
 - A few decision points (x_ℓ, x_c, x_r) played repeatedly
 - Samples used to estimate a confidence interval on $g(x)$
 - $\log(T)/\gamma^2$ samples suffice for a confidence interval of γ
 - Halve γ in every (doubling) epoch
 - eliminate a constant fraction of the continuous decision space



Our algorithm

In each epoch

- Play policies π^x instead of decision points $x \in (x_\ell, x_c, x_r)$
- Use empirical average of observed costs to estimate confidence bounds on the asymptotic cost g^x
- eliminate a constant fraction of the continuous decision space

From our key lemmas:

- Do $O(\log(T) / \gamma^2)$ samples still suffice for a confidence interval of γ ?



Two main lemmas: #1

Concentration lemma for MDPs in terms of bias [A. and Jia 2019]

- Let policy π be such that
 - the asymptotic average cost $g^\pi(s)$ is independent of the starting state s
 - the **bias** is bounded by H for all states
- Let $C_1, C_2, \dots, C_N$ are the observed costs on executing policy π starting from any state s_0 . Then with probability $1 - \delta$

$$\left| g^\pi - \frac{1}{N} \sum_i^N C_i \right| \leq H \sqrt{\frac{2 \log(1/\delta)}{N}}$$

- **Implication:** $H \log(1/\delta) / \gamma^2$ samples suffice for a confidence interval of γ
 - $O(H\sqrt{T})$ regret from regret analysis of stochastic convex bandit



Two main lemmas: #2

Bias lemma for inventory control MDP [A. and Jia 2019]

- In inventory control MDP, for any s, s' , and any base-stock policy π^x , bias is uniformly bounded as:

$$v^x(s) - v^x(s') \leq 36 \max(h, p) Lx \leq 36 \max(h, p) LU := H$$



Regret against the best base-stock policy

Almost black box reduction to stochastic convex bandit to compete with best single base-stock level

Main result [Agrawal, Jia 2019] Given any demand distribution F such that $F(0) > 0$. Then, there is an online learning algorithm for the inventory control problem with regret bounded as:

$$\text{Regret}(T) \leq \tilde{O}(DU^2 + LU\sqrt{T}) \cdot \max(h, p)$$

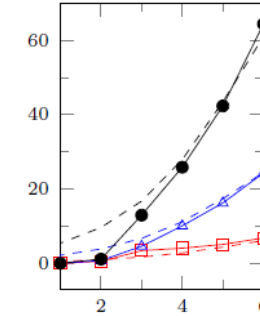
- Important: directly applying general RL results would give regret linear in number of states which is exponential in L
 - Previously best known bounds were exponential in L

[Huh, Janakiraman, Muckstadt, Rusmevichientong 2009a MOR, 2009b MS], [Zhang, Chao, and Shi. 2017]

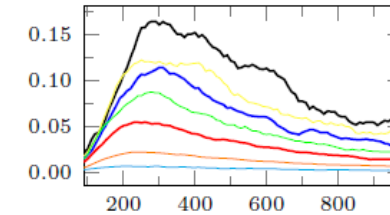


#2 Dynamic pricing under stateful demand models

- Word-of-mouth effects
 - Popular products get more popular
 - Demand depends on past popularity/sales/reviews
 - Non-linearity in sales growth
- Impacts pricing & advertising strategies:
 - Initial discounts to boost initial adoption
 - Sending products free to highly rated reviewers



New Product reviews on Amazon



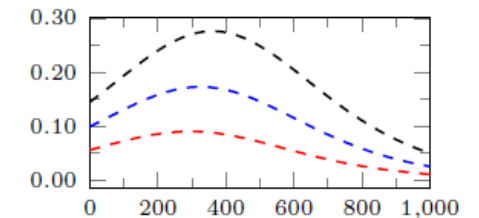
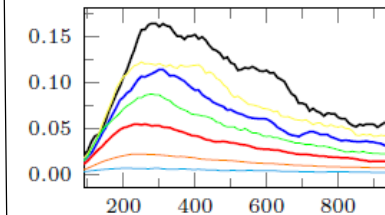
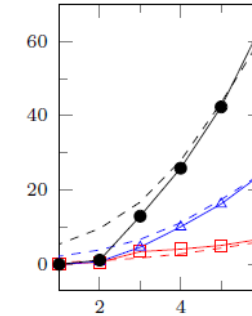
New answers for topics on Stack Overflow



Bass diffusion model [Bass 1969]

- Potential buyers (market size) m , Time horizon T
- Two parameter model
 - Coefficient of innovation α , Coefficient of imitation β
- $X_t \in [0,1]$ fraction of market captured

$$\frac{dX_t}{dt} = \underbrace{(\alpha + \beta X_t)}_{\text{Innovation imitation effects}} \underbrace{(1 - X_t)}_{\text{Proportion of market not penetrated}}$$



Bass model fitted to Stack
overflow and Amazon data

Learning to price under stochastic Bass model

At any instant $t \in [0, T]$ (continuous time)

- observe state $S_t \in \{0, \dots, m\}$ number of customers at time t [1-dim state]
 - $X_t = S_t/m$
- Set price p_t [1-dim action]
- Stochastic state transition $S_{t+dt} = S_t + N_t$ [next state]
$$N_t \sim \text{Poisson}(m e^{-p_t}(\alpha + \beta X_t)(1 - X_t)dt)$$
- reward $N_t p_t$ [reward]

Goal

- Maximize total revenue $\text{Regret}(T) = \text{OPT}(T) - \int_0^T N_t p_t$
 - $\text{OPT}(T)$ is expected total revenue of optimal stochastic (non-stationary) policy.

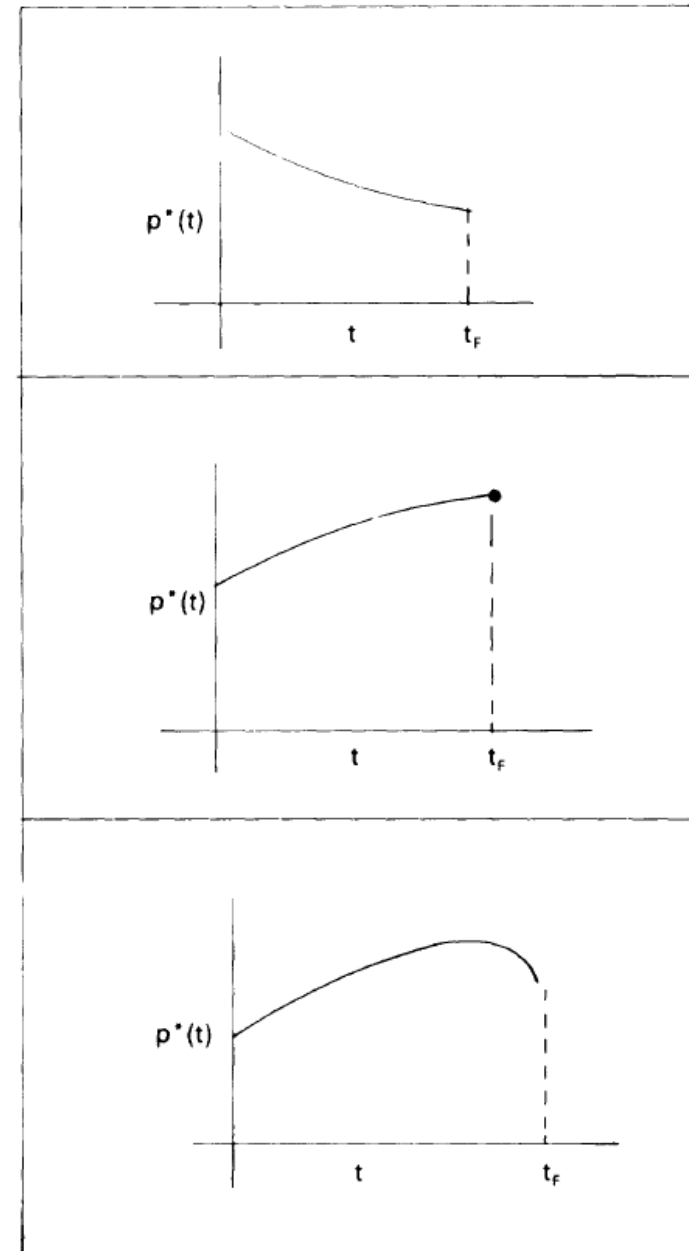
Optimal pricing policy

(For the deterministic model, no noise)

$$p^*(t) = 1 + \log\left[\frac{(\alpha + \beta X_t^*)(1 - X_t^*)}{(\alpha + \beta X_T^*)(1 - X_T^*)}\right]$$

- Highly non-stationary
 - Manipulates price based on time left
 - Tradeoff between market captured and price per sale

Plots taken from
[Dolan and Jeuland,
Journal of Marketing 1981]



$\beta < \alpha$

$\beta \gg \alpha$
Small horizon T

$\beta \gg \alpha$
Large horizon T

Challenges and opportunities

- Optimal policy is highly non-stationary
- Continuous time
- Continuous (although low dimensional) state

- ✓ Low dimensional parameter space
- ✓ Concavity of value function

Concavity property [A., Yin, Zeevi 2020]

- **Lemma:** For any t , the deterministic value function $V_t^{\text{det}}(X)$ is concave in adoption level (state) X

- **Implication:** for value functions at any time t , and state $X = S/m$

$$V_t^{\text{stoch}}(X) \leq V_t^{\text{det}}(X)$$

- **Deterministic optimal policy is good enough:** $\text{OPT}^{\text{stoch}}(T) \leq \text{OPT}^{\text{det}}(T)$
 - $\text{OPT}^{\text{det}}(T)$ is optimal revenue in the deterministic Bass model
- **Algorithm design and analysis using bandit techniques:** Estimate the model parameters using UCB techniques and imitate deterministic optimal policy

Result: Matching upper and lower bounds

- For our algorithm, with probability $1 - \delta$

$$\text{Regret}(T) \leq O\left(m^{\frac{2}{3}}\left(\frac{1}{\alpha} + \frac{1}{\beta} + \frac{\beta}{\alpha^2}\right) \log(m) \log\left(\frac{T}{\delta}\right)\right)$$

- Given any* algorithm fix any $\alpha, \beta, T = \Theta\left(\frac{1}{\alpha + \beta}\right)$, then there exists a market $\alpha, \beta \pm \frac{(\alpha + \beta)^2}{\beta}$ with

$$\text{Regret}(T) \geq \Omega\left(m^{\frac{2}{3}}\left(\frac{\alpha}{\alpha + \beta}\right)^{\frac{4}{3}}\right)$$

Take away and future directions

- **Take away:** Structural properties of OM problems can be used to simplify learning in MDPs
 - Convexity properties for efficient optimization
 - Learn simple parametric policies
 - Bandit techniques
- **Future work:** Learning more general state-dependent models for OM problems
 - Contextual versions
 - Is concavity of value function crucial?
 - Which other OM problems have such structural properties?