

Budgeted Learning for Search Ads and Other Applications

Michael Wunder

April 30, 2009

Budgeted Multi-Arm Bandit

Motivation

Problem Definition and Approach

Linear Programming Formulation

Exploration

Budgeted Multi-Arm Bandit: Ratio Index

Gittins Index

Ratio Index

Game Theory in Search Auctions

Motivation

There are a large number of applications for this problem.

- ▶ Search Advertising: Ad position?

Motivation

There are a large number of applications for this problem.

- ▶ Search Advertising: Ad position?
- ▶ Search Advertising: Advertisers would like to discover how advertising with certain keywords in certain markets affect revenues. This stage of advertising represents when the advertiser would like data not available from the search engine provider. However, they have a budget to explore the many possibilities open to them. How should they proceed?

Motivation

There are a large number of applications for this problem.

- ▶ Search Advertising: Ad position?
- ▶ Search Advertising: Advertisers would like to discover how advertising with certain keywords in certain markets affect revenues. This stage of advertising represents when the advertiser would like data not available from the search engine provider. However, they have a budget to explore the many possibilities open to them. How should they proceed?
- ▶ Keyword \rightarrow bandit

Overview of Budgeted Multi-Arm Bandit

- ▶ Each arm $i \in 1 \dots n$ has a prior distribution R_i , the reward received from choosing that arm, over K values $0 \leq \{a_1, a_2, \dots, a_k\} \leq 1$

Overview of Budgeted Multi-Arm Bandit

- ▶ Each arm $i \in 1 \dots n$ has a prior distribution R_i , the reward received from choosing that arm, over K values $0 \leq \{a_1, a_2, \dots, a_k\} \leq 1$
- ▶ There is a cost c_i for choosing an arm i

Overview of Budgeted Multi-Arm Bandit

- ▶ Each arm $i \in 1 \dots n$ has a prior distribution R_i , the reward received from choosing that arm, over K values
 $0 \leq \{a_1, a_2, \dots, a_k\} \leq 1$
- ▶ There is a cost c_i for choosing an arm i
- ▶ The total cost of pulling arms must not exceed some budget C , or can pull an arm no more than C times if the costs are 1

Overview of Budgeted Multi-Arm Bandit

- ▶ Each arm $i \in 1 \dots n$ has a prior distribution R_i , the reward received from choosing that arm, over K values
 $0 \leq \{a_1, a_2, \dots, a_k\} \leq 1$
- ▶ There is a cost c_i for choosing an arm i
- ▶ The total cost of pulling arms must not exceed some budget C , or can pull an arm no more than C times if the costs are 1
- ▶ Arms are assumed to be independent

Two approaches

We will look at two separate approximation methods for this problem.

- ▶ A greedy method that utilizes stochastic packing (Guha et al. [2])

Two approaches

We will look at two separate approximation methods for this problem.

- ▶ A greedy method that utilizes stochastic packing (Guha et al. [2])
- ▶ A Ratio method similar to the Gittins index (Goel et al. [1])

Two approaches

We will look at two separate approximation methods for this problem.

- ▶ A greedy method that utilizes stochastic packing (Guha et al. [2])
- ▶ A Ratio method similar to the Gittins index (Goel et al. [1])
- ▶ Both methods are constant approximations in relation to OPT, where OPT is the best *exploration* policy, not the best policy given the true data, as before.

State Space

- ▶ For a single arm, $\Sigma = \binom{h+K}{K}$

State Space

- ▶ For a single arm, $\Sigma = \binom{h+K}{K}$
- ▶ For all arms Σ^n

State Space

- ▶ For a single arm, $\Sigma = \binom{h+K}{K}$
- ▶ For all arms Σ^n
- ▶ However, it can be shown that the state space for a single arm can be reduced to $\Sigma = O\left(\left(\frac{\log n}{\epsilon^2}\right)^{\frac{1}{\epsilon}}\right)$ with loss of ϵ from the optimal, using Hoeffding bounds.

State Space

- ▶ For a single arm, $\Sigma = \binom{h+K}{K}$
- ▶ For all arms Σ^n
- ▶ However, it can be shown that the state space for a single arm can be reduced to $\Sigma = O\left(\left(\frac{\log n}{\epsilon^2}\right)^{\frac{1}{\epsilon}}\right)$ with loss of ϵ from the optimal, using Hoeffding bounds.
- ▶ Further, to discover an approximately optimal exploration policy, we do not have to define an action for every joint state, which is $\Omega(\Sigma^n)$. Instead a polynomial size LP is $O(nK\Sigma)$.

Linear Programming Relaxation

- ▶ Maximize $\sum_{i=1}^n \sum_{u \in T_i} x_u \zeta_i(u)$, where $\zeta = E[\theta_i(u)]$ is the reward for state u , given the following constraints
- ▶ $\sum_{i=1}^n c_i (\sum_{u \in T_i} z_u) \leq C$, which ensures that the expected cost per arm is less than or equal to the budget across all the arms.
- ▶ $\sum_{i=1}^n \sum_{u \in T} x_u \leq 1$ which ensures there exists a distribution over states to be exploited
- ▶ $\sum_{v: u \in D(v)} z_u p_{vu} = w_u, \forall i, u \in T_i$ where $D(v)$ indicates the child node of v , ρ_i is the root for arm i , p_{vu} is the probability of transitioning from v to u
- ▶ $x_u + z_u \leq w_u, \forall u \in T_i, \forall i$
- ▶ $x_u, z_u, w_u \in [0, 1], \forall u \in T_i, \forall i$

Exploration Policy \mathcal{A}_i

Exploration Policy \mathcal{A}_i using (w_u^*, x_u^*, z_u^*) to set \mathcal{E}_i if exploitation has been selected for arm i .

- ▶ Randomly select $q \in [0, w_u^*]$. q belongs to one of three intervals.

Exploration Policy \mathcal{A}_i

Exploration Policy \mathcal{A}_i using (w_u^*, x_u^*, z_u^*) to set \mathcal{E}_i if exploitation has been selected for arm i .

- ▶ Randomly select $q \in [0, w_u^*]$. q belongs to one of three intervals.
- ▶ If $q \geq 0$ and $q \leq z_u^*$, then play the arm.

Exploration Policy \mathcal{A}_i

Exploration Policy \mathcal{A}_i using (w_u^*, x_u^*, z_u^*) to set \mathcal{E}_i if exploitation has been selected for arm i .

- ▶ Randomly select $q \in [0, w_u^*]$. q belongs to one of three intervals.
- ▶ If $q \geq 0$ and $q \leq z_u^*$, then play the arm.
- ▶ If $q > z_u^*$ and $q \leq z_u^* + x_u^*$, then set $\mathcal{E}_i = 1$ and $X_i = \zeta_i(u)$.

Exploration Policy \mathcal{A}_i

Exploration Policy \mathcal{A}_i using (w_u^*, x_u^*, z_u^*) to set \mathcal{E}_i if exploitation has been selected for arm i .

- ▶ Randomly select $q \in [0, w_u^*]$. q belongs to one of three intervals.
- ▶ If $q \geq 0$ and $q \leq z_u^*$, then play the arm.
- ▶ If $q > z_u^*$ and $q \leq z_u^* + x_u^*$, then set $\mathcal{E}_i = 1$ and $X_i = \zeta_i(u)$.
- ▶ If $q \geq z_u^* + x_u^*$ and $q \leq w_u^*$, then set $\mathcal{E}_i = 0$ and $X_i = 0$.

GreedyViolate

Find the following values for each arm:

$$p_i = E[\mathcal{E}_i]$$

$$\eta_i = E[X_i]$$

$$\nu_i = E[C_i]/C$$

GreedyViolate

Now play each arm in decreasing order of $\frac{\eta_j}{p_j + \nu_j}$ according to policy \mathcal{A} .

When \mathcal{A}_j terminates after playing arm j , check if the next arm would violate the budget constraint. If it would, play this last arm for exploitation. If $\mathcal{E}_j = 1$, also exploit this arm. Otherwise, continue playing arms in sorted order.

GreedyViolate

Theorem

GreedyViolate achieves exploitation gain \mathcal{G}_{GV} of at least $\frac{OPT}{4}$.

- ▶ First renumber the arms according to order of arm playing and choose k such that $\sum_{i=1}^k \eta_i \geq \gamma^*/2 = \sum_i \eta_i/2$.

GreedyViolate

Theorem

GreedyViolate achieves exploitation gain \mathcal{G}_{GV} of at least $\frac{OPT}{4}$.

- ▶ First renumber the arms according to order of arm playing and choose k such that $\sum_{i=1}^k \eta_i \geq \gamma^*/2 = \sum_i \eta_i/2$.
- ▶ Use a knapsack packing method. Consider the arms as items with profit $\frac{\eta_i}{\gamma^*/2}$ and size $s_i = \nu_i + p_i$. (Now the expected cost and profit of each arm are normalized.)

GreedyViolate

Theorem

GreedyViolate achieves exploitation gain \mathcal{G}_{GV} of at least $\frac{OPT}{4}$.

- ▶ First renumber the arms according to order of arm playing and choose k such that $\sum_{i=1}^k \eta_i \geq \gamma^*/2 = \sum_i \eta_i/2$.
- ▶ Use a knapsack packing method. Consider the arms as items with profit $\frac{\eta_i}{\gamma^*/2}$ and size $s_i = \nu_i + p_i$. (Now the expected cost and profit of each arm are normalized.)
- ▶ Then $\sum_{i=1}^k \nu_i \geq 1$ and $\sum_{i=1}^{k-1} s_i \leq 1$. The profit when r is the size of the filled knapsack is $\phi(r)$.

GreedyViolate

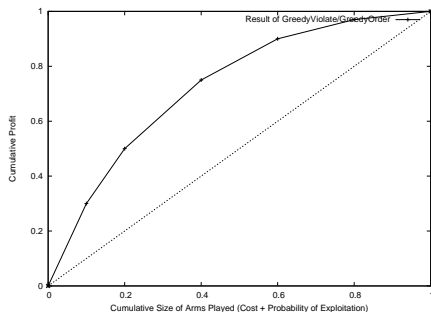


Figure: Curve formed by GreedyViolate/ GreedyOrder

GreedyViolate

Proof.

If we plot the cumulative profit against the exploration cost and exploitation threshold, the curve is concave. The area under the triangle including the origin, (1,0), and (1,1) is $1/2$, and this triangle corresponds to the stopping point of GreedyViolate. Since the area represents profit, and the area of the curve must be greater than $1/2$, the exploitation gain is greater than $\frac{OPT}{4}$.

$$\frac{1}{2} \leq AOC \leq \frac{\mathcal{G}_{GV}}{\gamma^*/2}$$

$$OPT \leq \gamma^*$$

$$\mathcal{G}_{GV} \geq \frac{OPT}{4}$$

GreedyOrder

GreedyOrder works the same as GreedyViolate, except that it stops the current arm if further exploration puts it over budget.

Claim: GreedyOrder achieves gain of at least $\frac{OPT}{4}$, while remaining under the budget constraints.

Proof.

GreedyOrder will continue playing the current arm because it is currently the best one selected for exploitation. The only difference between GreedyOrder and GreedyViolate is that GreedyViolate plays this arm during the exploration phase. Therefore, the same approximation holds for both, except that GreedyOrder meets the budget condition. □

Gittins Index

- ▶ Using only the state of the arm and a discount factor, compute a score for an arm, independently of all the other arms
- ▶ Separating the computation makes decision-making more efficient

Ratio Index (Goel et al. [1])

- ▶ P_{uv} is the probability of moving from state u to state v , h is exploration budget
- ▶ $\eta(u) = \sum_{v \in T} P_{uv} \eta(v)$
- ▶ Policy π returns an arm i to explore or terminates given state S
- ▶ $\mathcal{C}(\pi) = \frac{\sum_{u \in T} z_u^\pi}{h} + \sum_{u \in T} x_u^\pi$
- ▶ $\mathcal{P}(\pi) = \sum_{u \in T} x_u^\pi \eta(u)$
- ▶ Ratio Index defined as $r(u, h) = \max_{\pi} \frac{\mathcal{P}(\pi)}{\mathcal{C}(\pi)}$

Greedy Ratio Index Algorithm

The greedy algorithm always selects the arm with the maximum ratio index, $r(u_i, h)$. *Claim:* This greedy algorithm gives a $O(1)$ approximation to the budgeted learning problem. *Lemma 2.1* The ratio index policy does not abandon any arm-state v with $r(v, h) > r(u, h)$, and it does not explore or exploit an arm where $r(v, h) < r(u, h)$.

Proof.

The first part is obvious from the greedy selection. The second is the result of an LP technique and the relationship between the marginal profit of this arm and the ratio index. The main idea is that v cannot be a transitional state under the conditions stated by the ratio calculation. □

Persistent Ratio Index Algorithm

- ▶ Play arm i until the policy chooses to exploit or abandon the arm. Repeat the process after abandoning an arm. Then, if the budget is about to be exceeded, exploit the arm with the highest ratio index.

Claim: The greedy algorithm G earns at least as much as the persistent algorithm.

Proof.

From Lemma 2.1 and by the martingale property, early termination can never increase profit. □

Persistent Ratio Index Algorithm, Continued

Claim: $\mathcal{P}_G \geq .22\mathcal{P}_{OPT}$.

Proof.

The marginal profit p of the arm picked by the persistent algorithm at step j is guaranteed to be more than $(\mathcal{P}(\pi^*) - p_{j-1})/2$, and the probability that the budget is not exceeded is $(1 - c_{j-1})$.

Therefore, the profit is more than that obtained by this differential process, where p^* is the expected profit of OPT:

$$\frac{dp}{dc} = \frac{p^* - p}{2}(1 - c)$$

Integrating from $c = 0$ to $c = 1$ gives the expected profit of $(1 - e^{-0.25})p^* = 0.22p^*$. □

Game Theory in Search Auctions

For Stochastic Optimization, we assume that the bid landscape is stable for short periods of time. Ultimately, the auction setting is a repeated game over time, although with many unknown players. As analysis of this subject extends further, research about how game theoretic methods and optimization interact may be a promising area.



A. Goel, S. Khanna, and B. Null.

The ratio index for budgeted learning with applications.
SODA, 2009.



S. Guha and K. Munagala.

Approximation algorithms for budgeted learning problems.
STOC, 2007.