ϵ-Best-Arm Identification in Pay-Per-Reward Multi-Armed Bandits

Sivan Sabato

Ben-Gurion University of the Negev

In short

- ϵ-best-arm identification in stochastic multi-armed-bandits
- **Pay-Per-Reward**: The cost of each arm pull is proportional to the expected future reward of that arm
- The **MAB-PPR** algorithm solves this problem
- The cost of MAB-PPR **does not depend on the number of arms**
- Linear dependence on total expected reward
- Can work with infinitely many arms

Motivation

- **Finding best crowd-worker**: Payment during testing is proportional to test scores
- **Finding best text to link to ad**: Payment during survey is proportional to click rates

In general: Finding the best arm when pull cost is proportional to arm quality

The setting

ϵ-best arm identification

- There are K arms, each with a reward distribution
- Goal: find an arm with an ϵ-best expected reward

Pay-per-reward

The cost of each arm-pull is **proportional to the expected reward**

Weight $:=$ **arm quality** $\equiv$ **current cost** $\equiv$ **expected reward**

Additional assumptions:

- Number of arms can be **unbounded or infinite**
- Each arm has a bounded weight and variance
- Bounded total weight of arms W
- Independence of arm pulls (bounded variance of rewards of sums of arms)

Main result

Theorem:
MAB-PPR Finds an ϵ-best arm w.p. $1-\delta$ with a cost of $O(W\log(1/\delta)/\epsilon^2)$.

Cost of **MAB-PPR** does not depend on the number of arms!

Challenges

Challenge: Reducing costs fast

- No. of arms is unbounded $\Rightarrow$ should not remove a constant fraction
- Must reduce total arm weight fast
- Must remove arms based on **unknown** arm weights

Challenge: Weights are unknown

- Remove a constant fraction of the **expected arm weights**
- Need to estimate total weights of **sets of arms**

Challenge: Unbounded instantaneous total cost

- Number of arms is unbounded
  $\Rightarrow$ total instantaneous reward is unbounded
  $\Rightarrow$ need estimator for heavy tails
- We use the **Median-of-Means**

The Median-of-Means estimator (Alon et al. 1999)

- Concentration of estimates using only bounded variance
- If $\hat{w}$ is the MoM δ-estimator from n samples, then w.p. $1-\delta$,
  $|w-\hat{w}| \leq \sqrt{6\sigma^2\log(1/\delta)/n}$.

Sum of median-of-means $\neq$ Median-of-means of the sum

Approach

Estimating sums of arm weights

- Cannot combine individual weight estimates
- Cannot estimate all possible sums
- Instead,
  1. Use **two batches of arm pulls in each iteration**
  2. **Estimate** individual arm weights
  3. **Order** arms by estimated weight
  4. Estimate only **prefix sums** of ordering
- Select a prefix of the arms to remove

Analysis (I)

Lemma

In each iteration, at least a constant fraction of the **arm weights** is removed.

Lemma

In each iteration, at least one ϵ-heaviest arm remains.

Analysis (II)

Lemma: Few large weight estimates

If μ is sum of arms and $\hat{w}_i$ is the δ-estimator for δ from n samples, then w.p. $1-\delta$, then

$$[\text{No. arms such that } \hat{w}_i \geq \gamma W] \leq \frac{4}{\gamma}(1+\sqrt{6\sigma^2\log(1/\delta)/(W^2 n)}.$$

The Algorithm: MAB-PPR

Input: ϵ, δ **and access to arm-pulling**

Output: ϵ**-heaviest arm**

Initialization

$S \leftarrow$ set of all arms

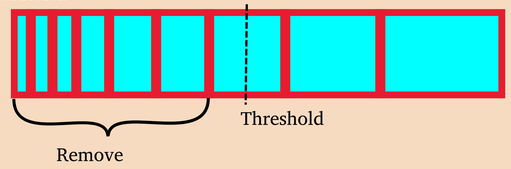
Iterative arm removal

In each iteration:

1. Pull all arms in S for $L_1(\epsilon,\delta)$ times
2. Estimate arm weights using Median-of-Means
3. Pull all arms in S for $L_2(\epsilon,\delta)$ times
4. Order the arms by increasing estimated weight
5. Estimate **prefix sums** of the arms using Median-of-Means
6. If the total sum is small, return some arm from S and terminate.
   Else, **remove arms with smallest estimated weight**

Removing arms

Remove arms with estimated total weight smaller than threshold



7. Update S, ϵ, δ.

Stop iterating when $|S| < N_{\text{final}}$.

Arm selection

- Pull all arms in S for $L_3(\epsilon,\delta)$ times
- Estimate arm weights using Median-of-Means
- Return the arm with maximal estimated weight