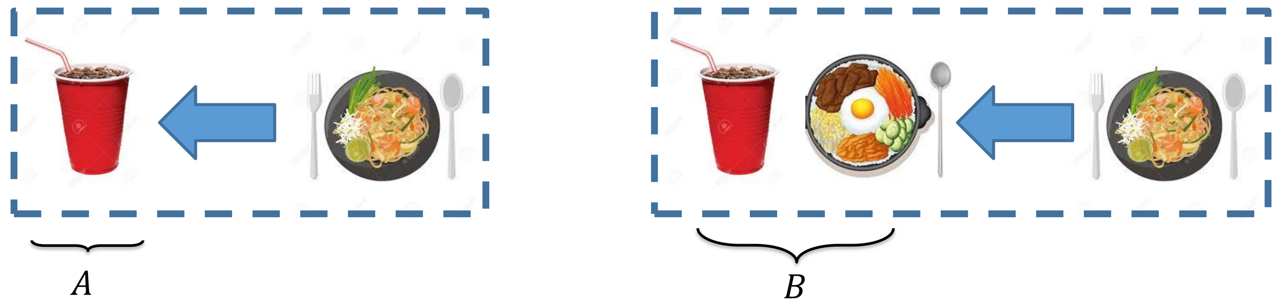
# Adversarially Robust Submodular Maximization under Knapsack Constraints

Dmitry Avydukhin[1], Slobodan Mitrović [2], Grigory Yaroslavtsev[1], Samson Zhou[1]
[1]Indiana University, [2]MIT

## SUBMODULAR FUNCTIONS

- Ground Set $V$ (items, sets, vertices)
- Set function $f: 2^V \to \mathbb{R}$ with diminishing returns property

$$\forall\, A \subseteq B \subseteq V, e \notin B,\ f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$$

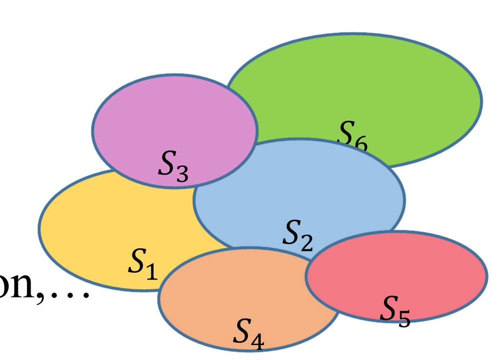



$$f(A \cup \{e\}) - f(A) \quad \geq \quad f(B \cup \{e\}) - f(B)$$

- Oracle access to $f$: Given a subset $S \subseteq V$, returns $f(S)$

## APPLICATIONS

- $E = \{e_1, e_2, \ldots, e_n\}, V \subseteq 2^E$
- $S = \{s_{i_1}, s_{i_2}, \ldots, s_{i_k}\} \in V,\ f(S) = |\cup_{s_i \in S}\, s_i|$
- $S^* = \arg\max_{|S| \leq k} f(S)$, $f$ is a submodular function
- Coverage, viral marketing, document summarization,…



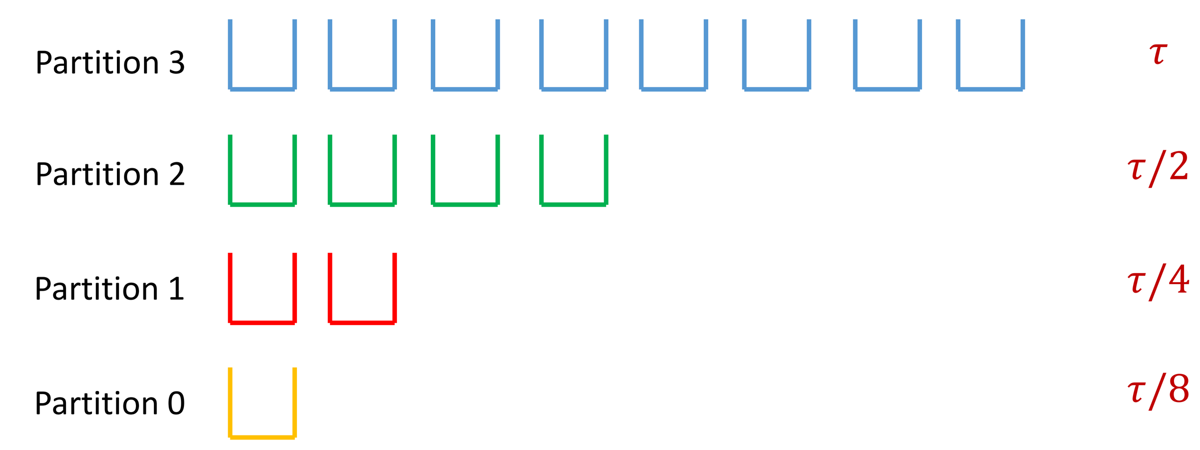

## CONSTRAINED SUBMODULAR OPTIMIZATION

- Cardinality constraint: $S^* = \arg\max_{|S| \leq k} f(S)$
- Knapsack constraint: Each element $e$ has cost $c(e)$, $S^* = \arg\max_{c(S) \leq K} f(S)$
- Multiple knapsacks: Costs $c_1(e), c_2(e), \ldots, c_d(e)$ and constraints $b_1, b_2, \ldots, b_d$, $S^* = \arg\max_{c_1(S) \leq b_1, \ldots, c_d(S) \leq b_d} f(S)$
- Adversarially robust submodular optimization: Given set $E$ that is removed from $V$ after summary is produced, (knapsack) $S^* = \arg\max_{c(S) \leq K, S \cap E = \emptyset} f(S)$

- Movie recommendation: Want multiple genres of movies (multiple knapsacks), and removal of movies that have already been watched (robust)
- In big data models, do not want to compute the output from scratch
- Streaming model: items arrive sequentially, minimize space
- Distributed model: items across machines, minimize communication

## BACKGROUND

- Constrained submodular maximization is NP-hard
- Marginal gain: $f(e|A) \coloneqq f(A \cup \{e\}) - f(A)$
- Greedy algorithm for cardinality constraint: repeatedly take the item with the largest marginal gain $\to (1 - 1/e)$ - approximation [NWF78]
- Thresholding for cardinality constraint in the streaming model: take any items whose marginal gain exceeds $\frac{f(\mathrm{opt})}{2k} \to \frac{1}{2}$ - approximation [BMKK14]
  - Can make geometrically increasing guesses for $f(\mathrm{opt})$
  - Uses $O\left(\frac{1}{\epsilon} k \log n\right)$ space
- Marginal density: $\rho(e|A) \coloneqq \frac{f(A \cup \{e\}) - f(A)}{c(e)}$
- Thresholding for knapsack constraint in the streaming model: take any items whose marginal density exceeds $\frac{2f(\mathrm{opt})}{3k}$ OR the best single item $\to \frac{1}{3}$ - approximation [HKY17]
- Constant factor approximation for adversarially robust submodular maximization with cardinality constraint [BMSC17]
- Partitions and buckets approach:




- Intuition: Items in high partitions are more valuable
- Bad approximation if they are deleted, so we need more buckets
- Items in low partitions are not as valuable
- Still have good approximation if many buckets are full
- If many items are deleted from high partitions, but buckets in low partitions are not full, must still have captured "good" items

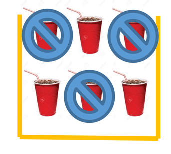
## RESULTS

- Streaming algorithm for single knapsack, robust to the removal of $m$ items
  - Constant factor approximation, outputs $\tilde{O}(K + m)$ elements in robust summary, and uses space $\tilde{O}(K^2 + mK)$
- Streaming algorithm for single knapsack, robust to the removal of size $M$
  - Constant factor approximation, outputs $\tilde{O}(K + M)$ elements in robust summary, and uses space $\tilde{O}(K + M)$
- Streaming algorithm for $d$ knapsacks, robust to the removal of $m$ items
  - $\Omega\left(\frac{1}{d}\right)$ approximation, outputs $\tilde{O}(K + m)$ elements in robust summary, and uses space $\tilde{O}(K^2 + mK)$
- Distributed algorithm for multiple knapsack, robust to the removal of $m$ items
  - $\Omega\left(\frac{1}{d}\right)$ approximation, two rounds of communication, $\tilde{O}\left((m + K)\sqrt{n}\right)$ storage per machine

## KNAPSACK ROBUSTNESS

- Initial idea: replace marginal gain with marginal density

- Problem: big items can't fit
- Hotfix: double the size of each bucket

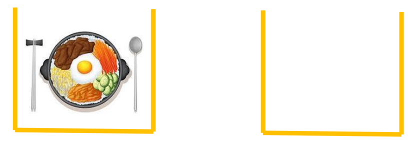


Remains good approximation

- Problem: number of buckets is based on the threshold, not size

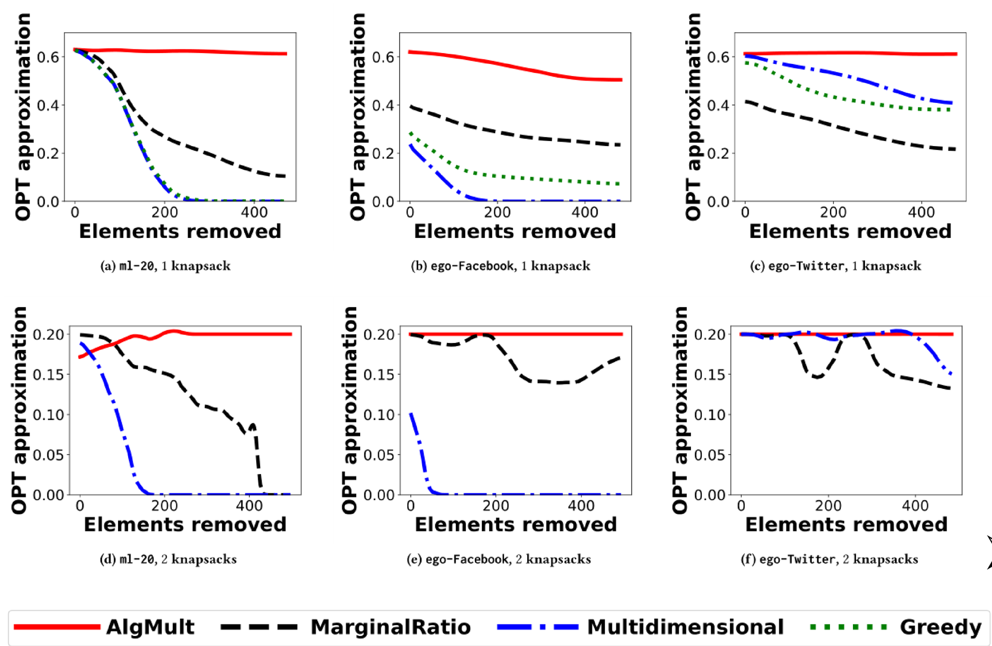
Bad approximation!

- Main idea: *Dynamic* bucketing scheme
- Each time element is added, allocate space proportional to its size



- Cap total number of items $\to$ Constant factor approximation
- Normalization for multiple knapsacks: rescale each row $i$ in cost matrix by $b_1/b_i$ so that all knapsack constraints are $K \coloneqq b_1$.
- Rescale all entries in cost matrix and constraint vector by minimum entry so that all costs are at least 1.
- "Marginal density": marginal gain divided by the *largest* cost (across all knapsacks)

## EXPERIMENTS




- Social network graphs from Facebook (4K vertices, 81K edges) and Twitter (88K vertices, 1.8M edges) collected by the Stanford Network Analysis Project (SNAP), MovieLens (27K movies, 200K ratings)
- Baselines: Offline Greedy, "Robustified" versions of streaming algorithms

| | ml-20, 1 knapsack | fb, 1 knapsack | twitter, 1 knapsack | ml-20, 2 knapsacks | fb, 2 knapsacks | twitter, 2 knapsacks |
|---|---|---|---|---|---|---|
| AlgMult | 641 | 378 | 401 | 1350 | 2745 | 4208 |
| MarginalRatio | 641 | 377 | 402 | 1350 | 2745 | 4209 |
| Multidimensional | 87 | 18 | 435 | 72 | 22 | 4221 |
| Greedy | 647 | 393 | 493 | - | - | - |

**Table 1: Sizes of robust summaries produced by the algorithms ($K = 10$).**

## REFERENCES

[BMKK14] Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. Streaming submodular maximization: Massive data summarization on the fly. KDD 2014.
[HKY17] Chien-Chung Huang, Naonori Kakimura, and Yuichi Yoshida. Streaming algorithms for maximizing monotone submodular functions under a knapsack constraint. APPROX 2017
[BMSC17] Ilija Bogunovic, Slobodan Mitrović, Jonathan Scarlett, and Volkan Cevher. Robust submodular maximization: A non-uniform partitioning approach. ICML 2017
[NWF78] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions - I.Math. Program. 14(1):265–294, 1978