Fairness in Streaming Submodular Maximization

Marwa El Halabi, Slobodan Mitrovic, Ashkan Norouzi-Fard,
Jakab Tardos, Jakub Tarnawski

Why fairness?

- ML algorithms are used in sensitive domains: voting, hiring, criminal justice, access to credit, etc
- Evidence of bias and discrimination:

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn't Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

  - An algorithm called COMPAS used to predict if a criminal will reoffend produces much higher false positive rate for black people than white people [Angwin et al, 2016]
  - Under-representation of women in search results [Kay et al, 2015], e.g., for the search term "CEO", 11% of top 100 results on Google Images are women vs 27% in the ground truth
  - Gender and race bias in word embeddings [Caliskan et al, 2017, Bolukbasi et al, 2016], e.g., European American names are more associated with pleasant than unpleasant terms, compared to African American names, and female names are more associated with family than career words, compared to male names.

Image source: [Angwin et al, 2016]

Streaming submodular maximization

- Natural model for **data summarization**
- **Applications:** exemplar-based clustering, document and corpus summarization, video summarization, recommender systems
- **Streaming setting:** limited memory
- **Submodularity:** diminishing returns property

$$f(S \cup e) - f(S) \geq f(T \cup \{e\}) - f(T) \text{ for all } S \subseteq T$$

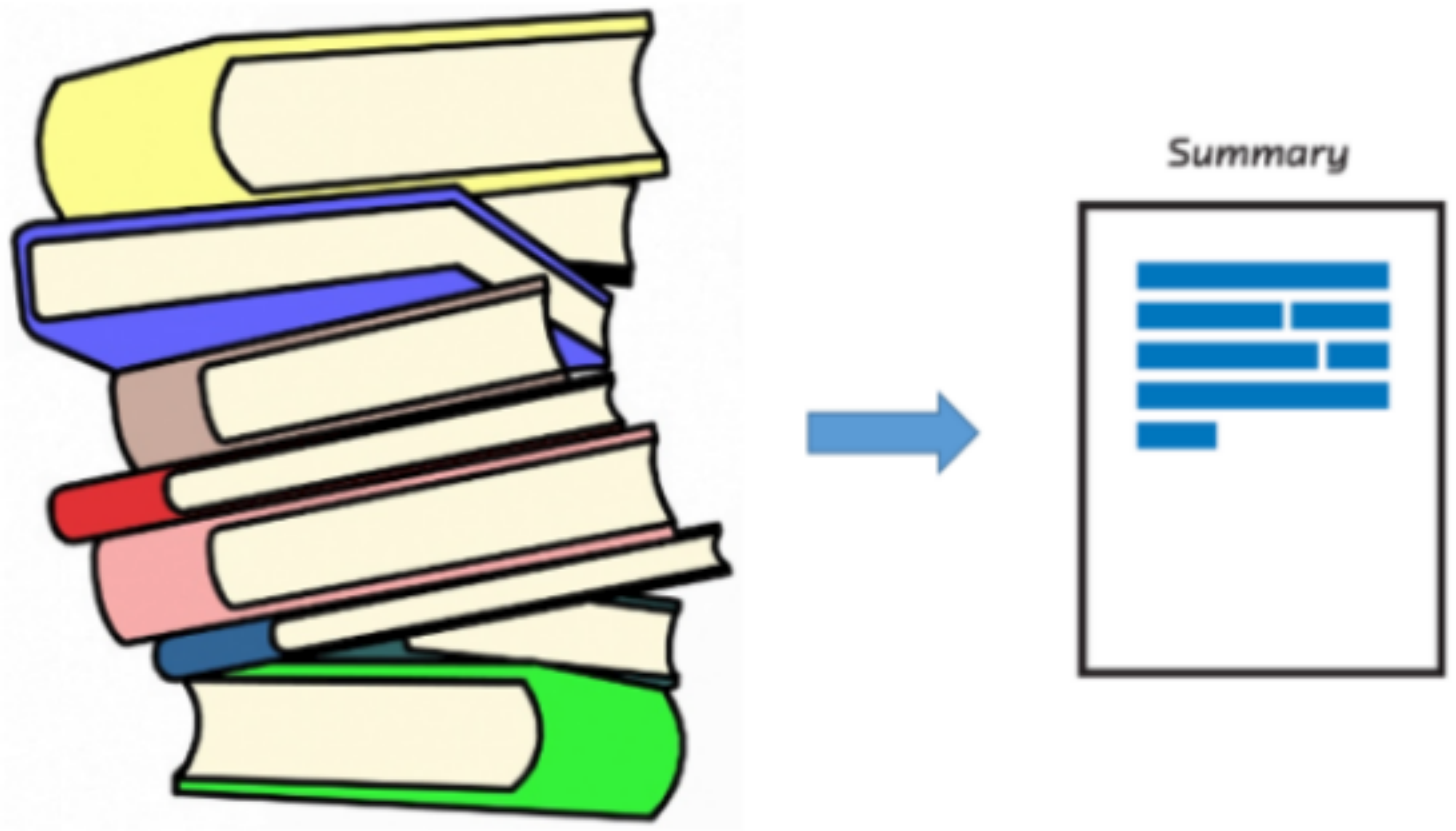
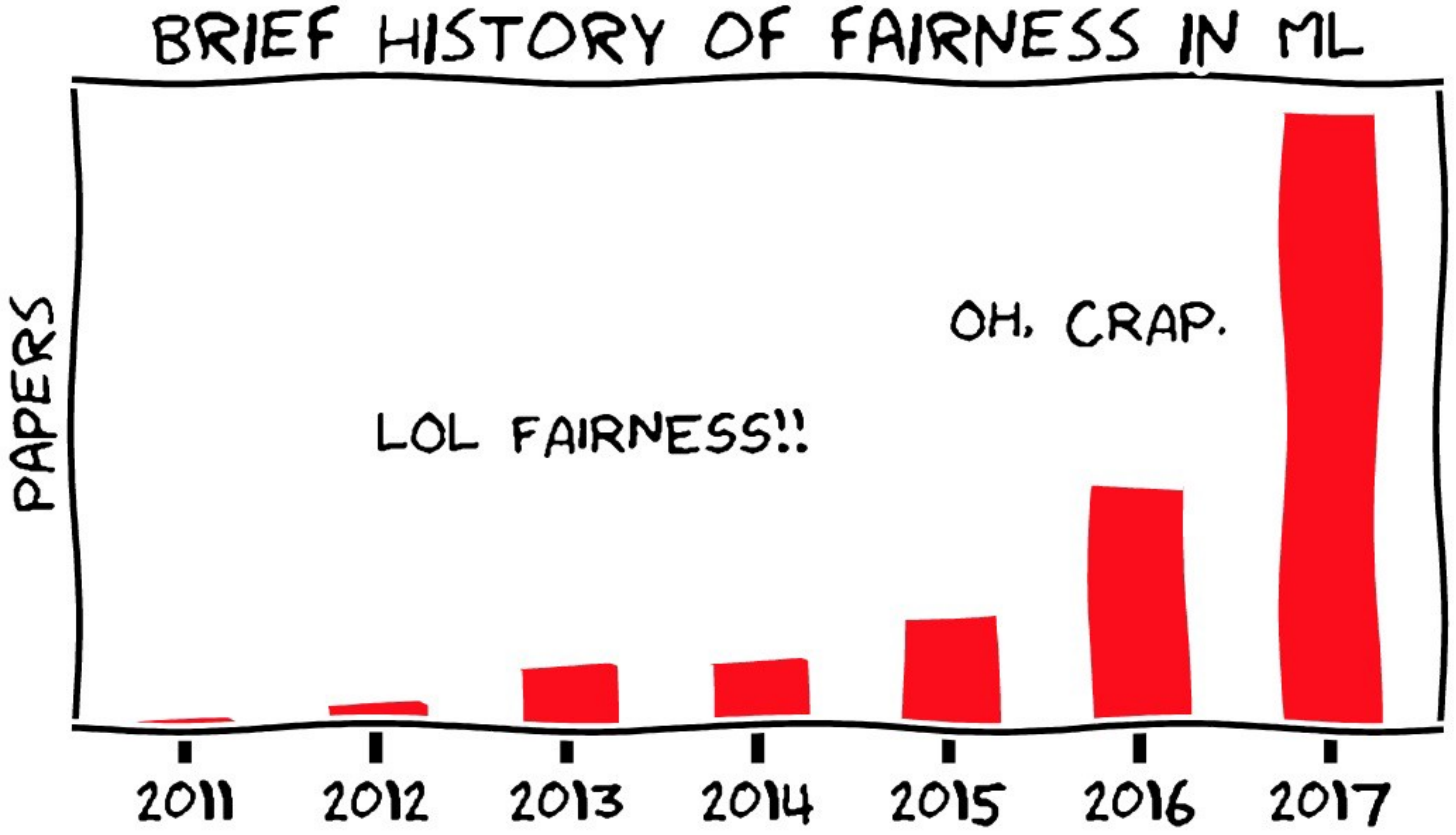


Image source: https://towardsdatascience.com/text-summarization-using-deep-learning-6e379ed2e89c

Related work



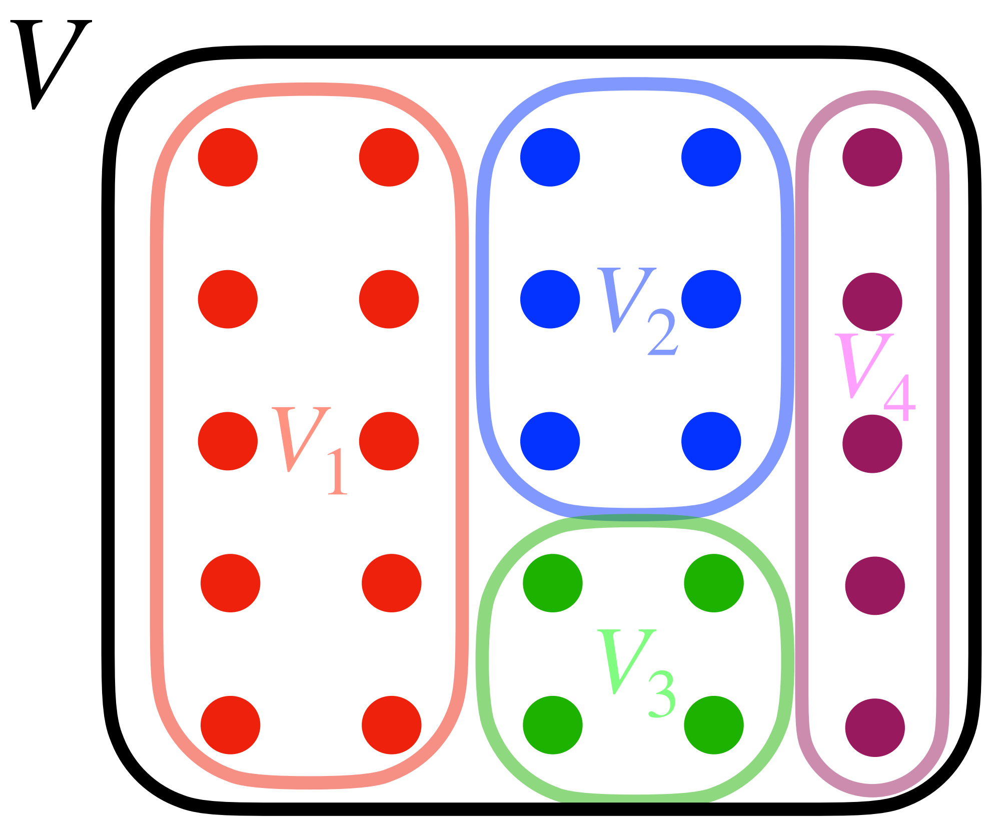
- Recent work on developing fair algorithms for fundamental problems, such as classification [Zafar et al, 2017], influence maximization [Tsang et al, 2019], ranking [Celis et al, 2018a], clustering [Chierichetti et al, 2017, Backurs et al, 2019, Jia et al, 2020], and diverse data summarization [Celis et al, 2018b].

- Fair submodular maximization only studied in **offline** setting for **monotone** objectives [Celis et al, 2017]

Image source: https://towardsdatascience.com/a-tutorial-on-fairness-in-machine-learning-3ff8ba1040cb

What does it mean to be fair?

- Solution should be "balanced" with respect to some **sensitive attribute** (e.g., race, gender) [Celis et al, 2017, Chierichetti et al, 2017, Celis et al, 2018b, Chierichetti et al, 2019]
- Given a set of n items (e.g., people) $V = \{1, \cdots, n\}$
- Each item is assigned a color encoding the sensitive attribute.
- $V_1, \cdots, V_C$ are the corresponding C disjoint color groups



A set $S \subseteq V$ is fair iff

$$\ell_c \leq |S \cap V_c| \leq u_c \text{ for all c}$$

Common choice: ℓ_c and $u_c \propto \frac{|V_c|}{n}$

Fair streaming submodular maximization

Given: ground set $V = V_1 \cup \cdots \cup V_c$, submodular function $f : 2^V \to \mathbb{R}_{\geq 0}$

$$\max_{S \in \mathcal{F}} f(S)$$

where $\mathcal{F} = \{S \subseteq V : |S| \leq k, |S \cap V_c| \in [\ell_c, u_c] \text{ for all } c = 1, \cdots, C\}$

Single-pass streaming setting: scan data stream only once, use limited memory ($m \ll n$)

Assumptions:

- f is normalized, i.e., $f(\emptyset) = 0$
- There exists a feasible solution, i.e., $\mathcal{F} \neq \emptyset \Rightarrow \sum_{c=1}^{C} \ell_c \leq k$
- Two settings: monotone $f(S) \geq f(T)$ for all $S \subseteq T$, and non-monotone

Fair streaming submodular maximization

Given: ground set $V = V_1 \cup \cdots \cup V_c$, submodular function $f : 2^V \to \mathbb{R}_{\geq 0}$

$$\max_{S \in \mathcal{F}} f(S)$$

where $\mathcal{F} = \{S \subseteq V : |S| \leq k, |S \cap V_c| \in [\ell_c, u_c] \text{ for all } c = 1, \cdots, C\}$

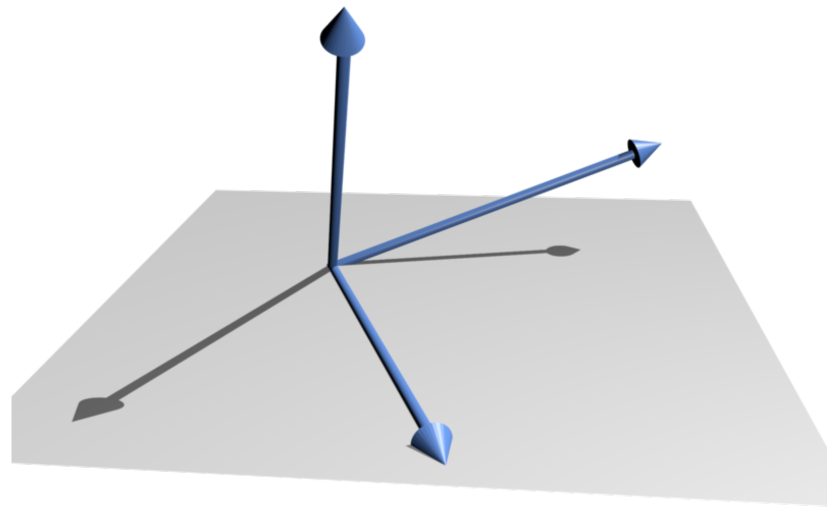
How hard is this problem?

- In offline setting, monotone objectives: $(1 - 1/e)$-approximation [Celis et al, 2017]
- In streaming setting, and for special case of cardinality constraint alone:
  - For monotone objectives: $(1/2 - \epsilon)$-approximation [Badanidiyuru et al, 2014]. This is tight [Feldman et al, 2020].
  - For non-monotone objectives: $1/5.82$-approximation [Feldman et al, 2018]

Relation to other problems

Idea: let's be lazy! **Can we reduce this problem to another well-studied problem?**

- Monotone case: Yes! We reduce this to submodular maximisation over **matroid** constraints
- Non-monotone case: Almost!



Matroid constraints:

- capture many natural constraints: cardinality $|S| \leq k$, partition matroid $|S \cap V_c| \leq u_c$
- existing efficient streaming algorithms:
  - $1/4$-approximation for monotone objectives [Chakrabarti et al, 2014]
  - $1/5.82$-approximation for non-monotone objectives [Feldman et al, 2018]

Relation to matroid constraints

Matroid constraints:

- capture many natural constraints: cardinality $|S| \le k$, partition matroid $|S \cap V_c| \le u_c$
- existing efficient streaming algorithms:
  - $1/4$-approximation for monotone objectives [Chakrabarti et al, 2014]
  - $1/5.82$-approximation for non-monotone objectives [Feldman et al, 2018]

$$\mathcal{F} = \{S \subseteq V : |S| \le k, |S \cap V_c| \in [\ell_c, u_c] \text{ for all } c = 1, \cdots, C\}$$

Is $\mathcal{F}$ a matroid? No

- Without lower bounds (i.e., $\ell_c = 0$), $\mathcal{F}$ is a laminar matroid

Monotone case: Reduction

$$\mathcal{F} = \{S \subseteq V : |S| \leq k, |S \cap V_c| \in [\ell_c, u_c] \text{ for all } c = 1, \cdots, C\}$$

- Without lower bounds (i.e., $\ell_c = 0$), $\mathcal{F}$ is a laminar matroid
- **Idea:** use matroid streaming algorithm then augment the solution with backup elements
- **Difficulty:** Solution might violate cardinality constraint!
- Define **extendable sets** $\tilde{\mathcal{F}} = \{S \subseteq V :$ there exists a feasible set $S' \in \mathcal{F}$ such that $S \subseteq S'\}$

A set S is extendable iff $|S \cap V_c| \leq u_c$ for all c and $\sum_{c=1}^{C} \max\{|S \cap V_c|, \ell_c\} \leq k$

Monotone case: Reduction

$$\mathcal{F} = \{S \subseteq V : |S| \leq k, |S \cap V_c| \in [\ell_c, u_c] \text{ for all } c = 1, \cdots, C\}$$

- Without lower bounds (i.e., $\ell_c = 0$), $\mathcal{F}$ is a laminar matroid
- **Idea:** use matroid streaming algorithm then augment the solution with backup elements
- **Difficulty:** Solution might violate cardinality constraint!
- Define **extendable sets** $\tilde{\mathcal{F}} = \{S \subseteq V :$ there exists a feasible set $S' \in \mathcal{F}$ such that $S \subseteq S'\}$

Key insight: $\tilde{\mathcal{F}}$ is a matroid!

Monotone case: Algorithm

$\mathscr{A}$: Streaming algorithm for monotone submodular maximisation over matroid constraint

Fair-Streaming algorithm:

1. Run $\mathscr{A}$ to construct an extendable set $S_{\mathscr{A}}$
2. In parallel: collect ℓ_c backup elements for every color c
3. At the end: augment $S_{\mathscr{A}}$ to a feasible set S using backup elements

Theorem: Fair-Streaming has the same approximation ratio, memory usage, and running time as $\mathscr{A}$.

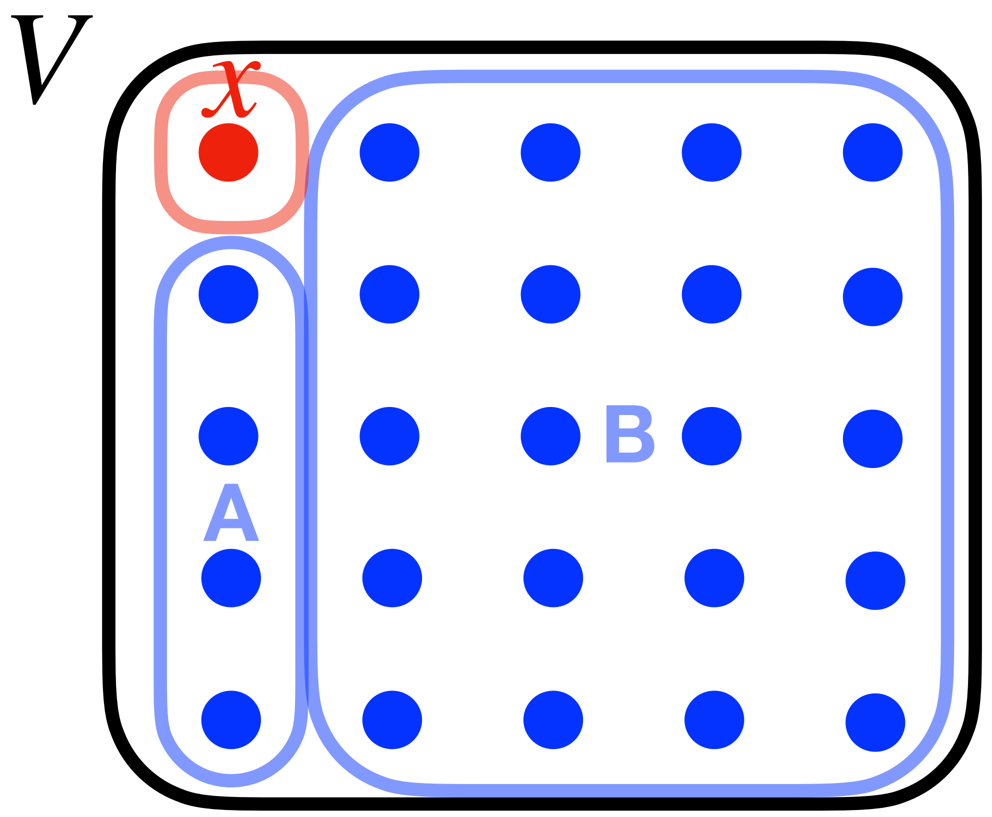
➡ $1/2$-approximation, $k^{O(k)}$ memory, using algorithm of [Huang et al, 2020]

➡ $1/4$-approximation, $O(k)$ memory, using algorithm of [Chakrabarti et al, 2014]

Non-monotone case: Hardness

Can we follow the same approach? Not exactly..

Difficulty: adding backup elements can drastically decrease solution value



$$F(S) = \begin{cases} |S| \text{ if } x \notin S \\ |S \cap A| \text{ if } x \in S \end{cases}$$

$$\ell_{\text{red}} = u_{\text{red}} = 1$$

$$|A| \ll |B|$$

Elements in A and B are indistinguishable before seeing x

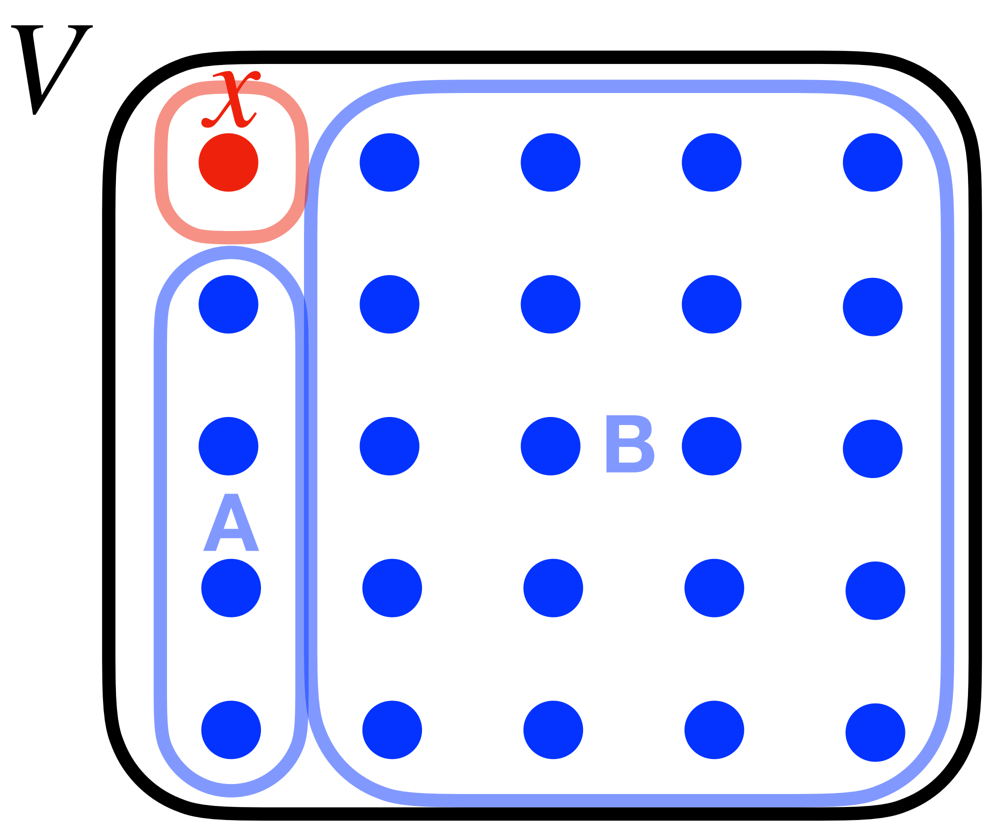
⇒ Any algorithm that does not store all of V will have $F(S) \approx 0$

$$\boxed{\textbf{Excess ratio: } q = 1 - \max_c \frac{\ell_c}{|V_c|}}$$

Non-monotone case: Hardness

Can we follow the same approach? Not exactly..

Difficulty: adding backup elements can drastically decrease solution value



$$F(S) = \begin{cases} |S| \text{ if } x \notin S \\ |S \cap A| \text{ if } x \in S \end{cases}$$

$\ell_{\text{red}} = u_{\text{red}} = 1$

$|A| \ll |B|$

Elements in A and B are indistinguishable before seeing x

$\Rightarrow$ Any algorithm that does not store all of V will have $F(S) \approx 0$

Excess ratio: $q = 1 - \max_c \frac{\ell_c}{|V_c|}$

Theorem: For any $\epsilon > 0$, and excess ratio $q \in [0,1]$, any $(q + \epsilon)$-approximation algorithm requires $\Omega(n)$ memory.

Non-monotone case: Reduction

Assumption: excess ratio $q = 1 - \max_c \frac{\ell_c}{|V_c|}$ is not too small

Extendable sets $\tilde{\mathscr{F}} = \{S \subseteq V :$ **there exists a feasible set** $S' \in \mathscr{F}$ **such that** $S \subseteq S'\}$

- **Idea:** use matroid streaming algorithm $\mathscr{A}$ to construct an extendable set $S_{\mathscr{A}}$, then augment the solution with backup elements

- **Difficulty:** adding backup elements can drastically decrease solution value

- **Helper Lemma:** If $g : 2^V \to \mathbb{R}_{\geq 0}$ is submodular, and $B \subseteq V$ is a random set where $e \in B$ with probability at most $1 - q \Rightarrow \mathbb{E}[g(B)] \geq q\ g(\emptyset)$ [Buchbinder et al, 2014]

➡ Apply helper lemma to $g(S) = f(S \cup S_{\mathscr{A}})$, and pick backup elements randomly

Non-monotone case: Algorithm

$\mathscr{A}$: Streaming algorithm for non-monotone submodular maximisation over matroid constraint

Fair-Sample-Streaming algorithm:

1. Run $\mathscr{A}$ to construct an extendable set $S_{\mathscr{A}}$
2. In parallel: sample without replacement ℓ_c backup elements for every color c, using **reservoir sampling**
3. At the end: augment $S_{\mathscr{A}}$ to a feasible set S using backup elements

Theorem: Fair-Sample-Streaming loses at most a factor q of the approximation ratio of $\mathscr{A}$, and has the same memory usage, and running time as $\mathscr{A}$

➡ $q/5.82$-approximation, $O(k)$ memory, using algorithm of [Feldman et al, 2018]

Empirical evaluation

Problem: $\max_{S \in \mathcal{F}} f(S)$ where $\mathcal{F} = \{S \subseteq V : |S| \leq k, |S \cap V_c| \in [\ell_c, u_c]$ for all $c = 1, \cdots, C\}$

Criteria:

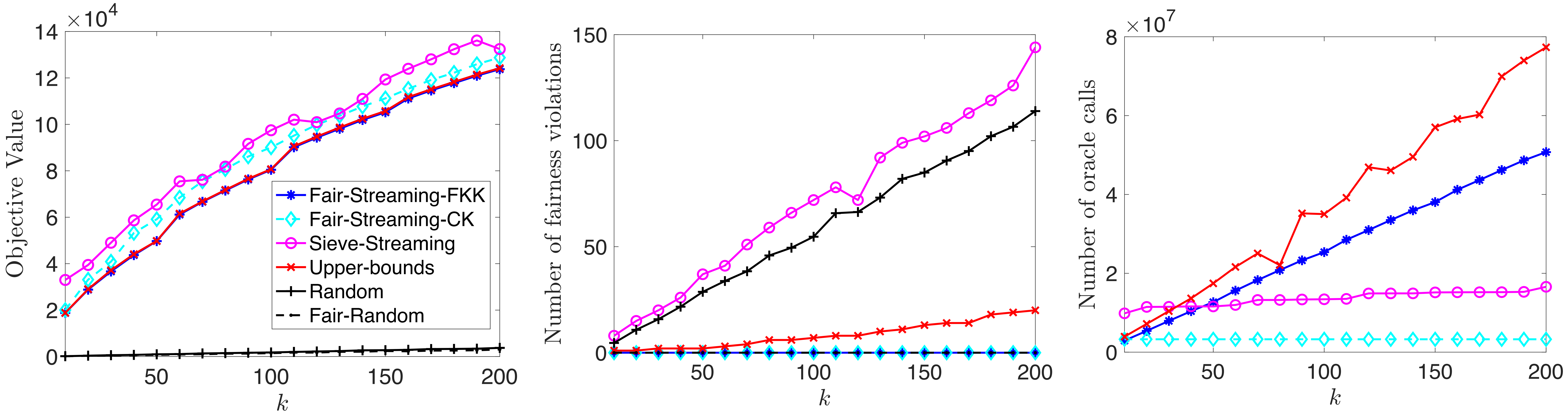
1. Objective value
2. Violation of fairness constraints: $\mathrm{err}(S) = \sum_{c \in [C]} \max\{|S \cap V_c| - u_c, \ell_c - |S \cap V_c|, 0\}$
3. Number of oracle calls

"Unfair" baselines:

- **Upper-Bounds:** streaming algorithm for matroid constraint [Feldman et al, 2018], applied to matroid defining upper bounds and cardinality constraint only; both monotone and non-monotone
- **Sieve-Streaming:** streaming algorithm for cardinality constraint [Badanidiyuru et al, 2014]; monotone

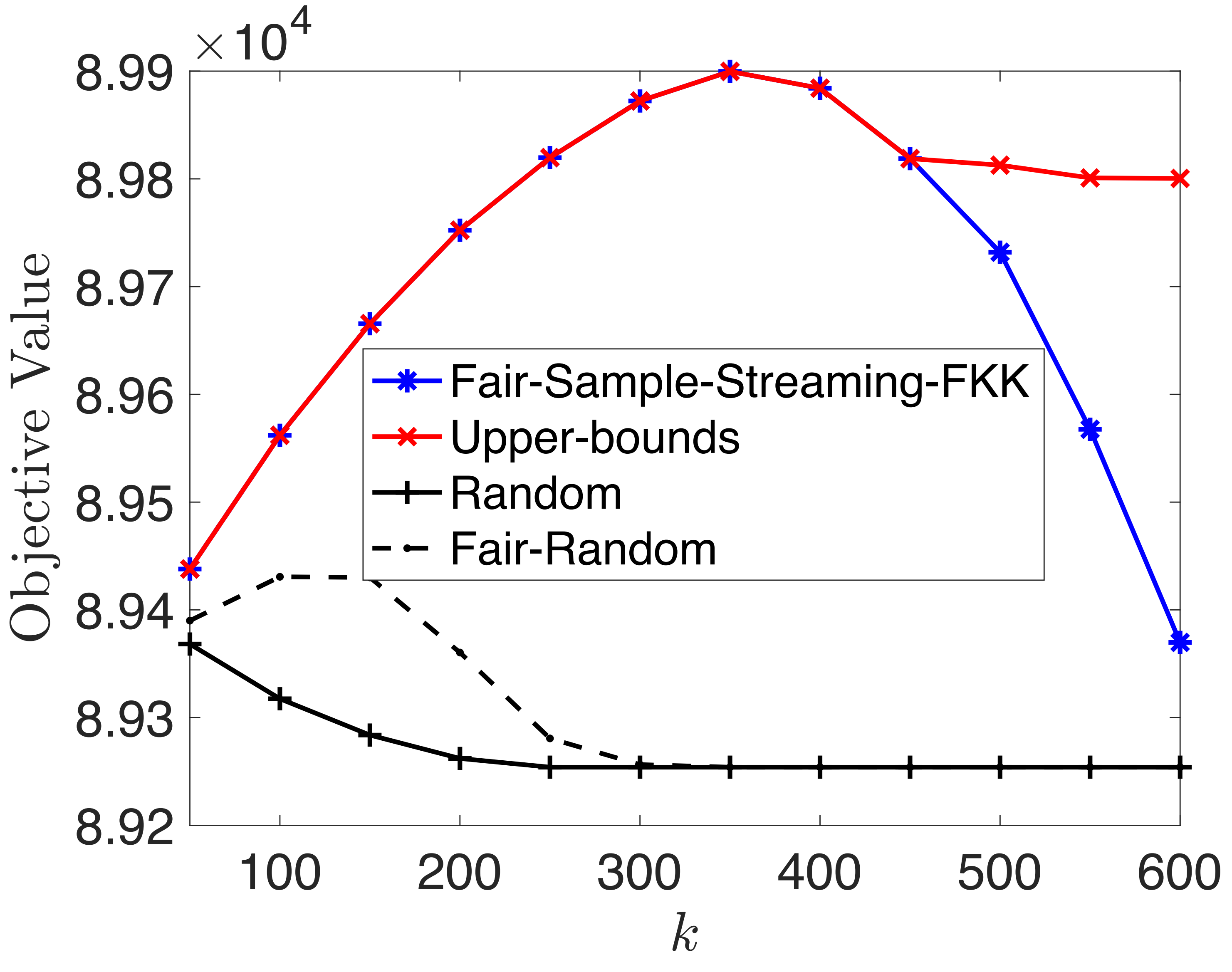
Social influence maximization

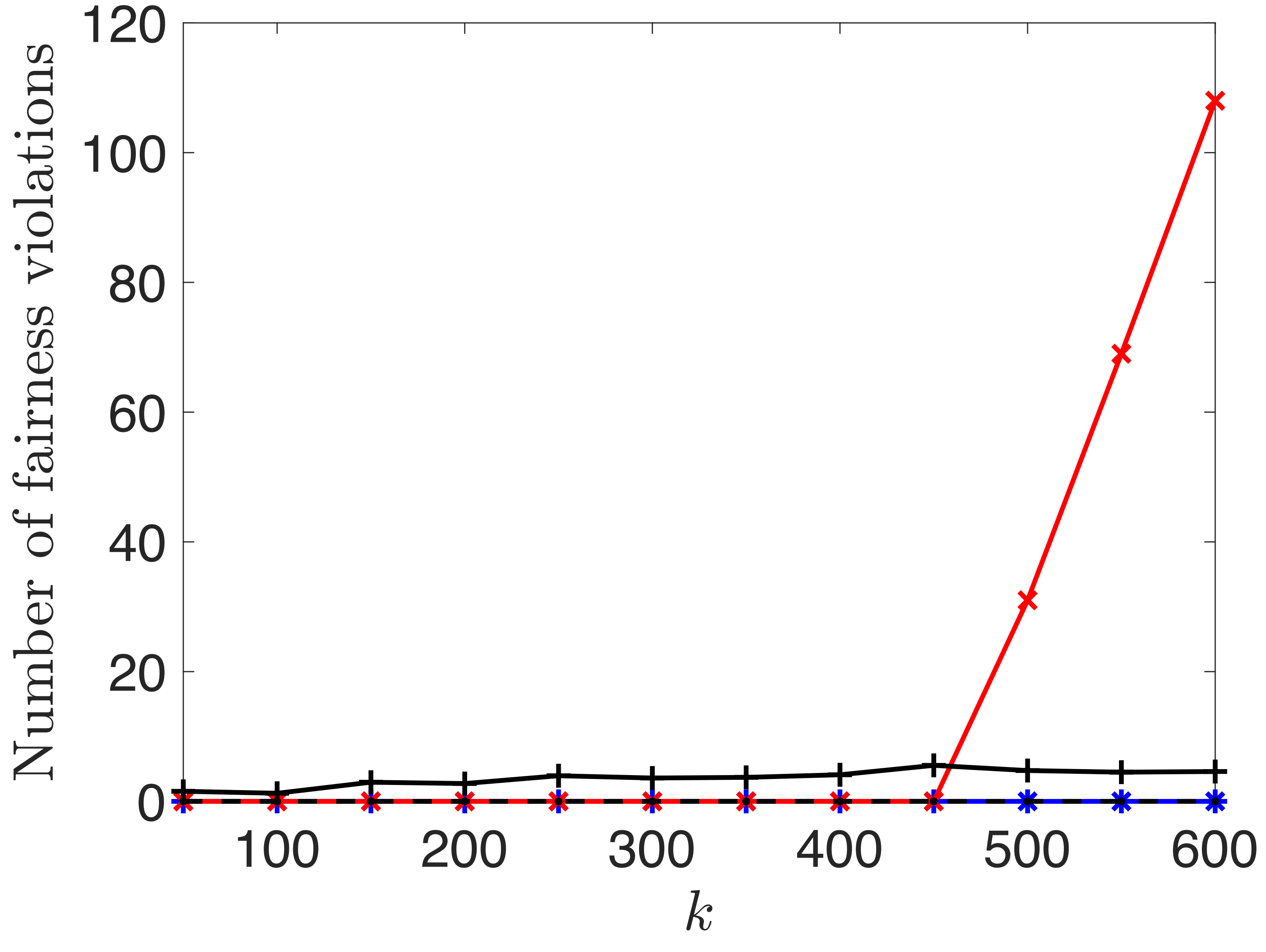
- Dataset: Pokec social network [Leskovec et al, 2014], $1\,632\,803$ nodes (users), $30\,622\,564$ edges (friendships)
- Objective (monotone submodular): $f(S) = |\cup_{v \in S} N(v)|$ where $N(v)$ is the set of neighbors of node v
- Sensitive attribute and bounds: age, $\ell_c = \max\{0, |V_c|/|V| - 0.05\} \cdot k, u_c = \min\{1, |V_c|/|V| + 0.05\} \cdot k$

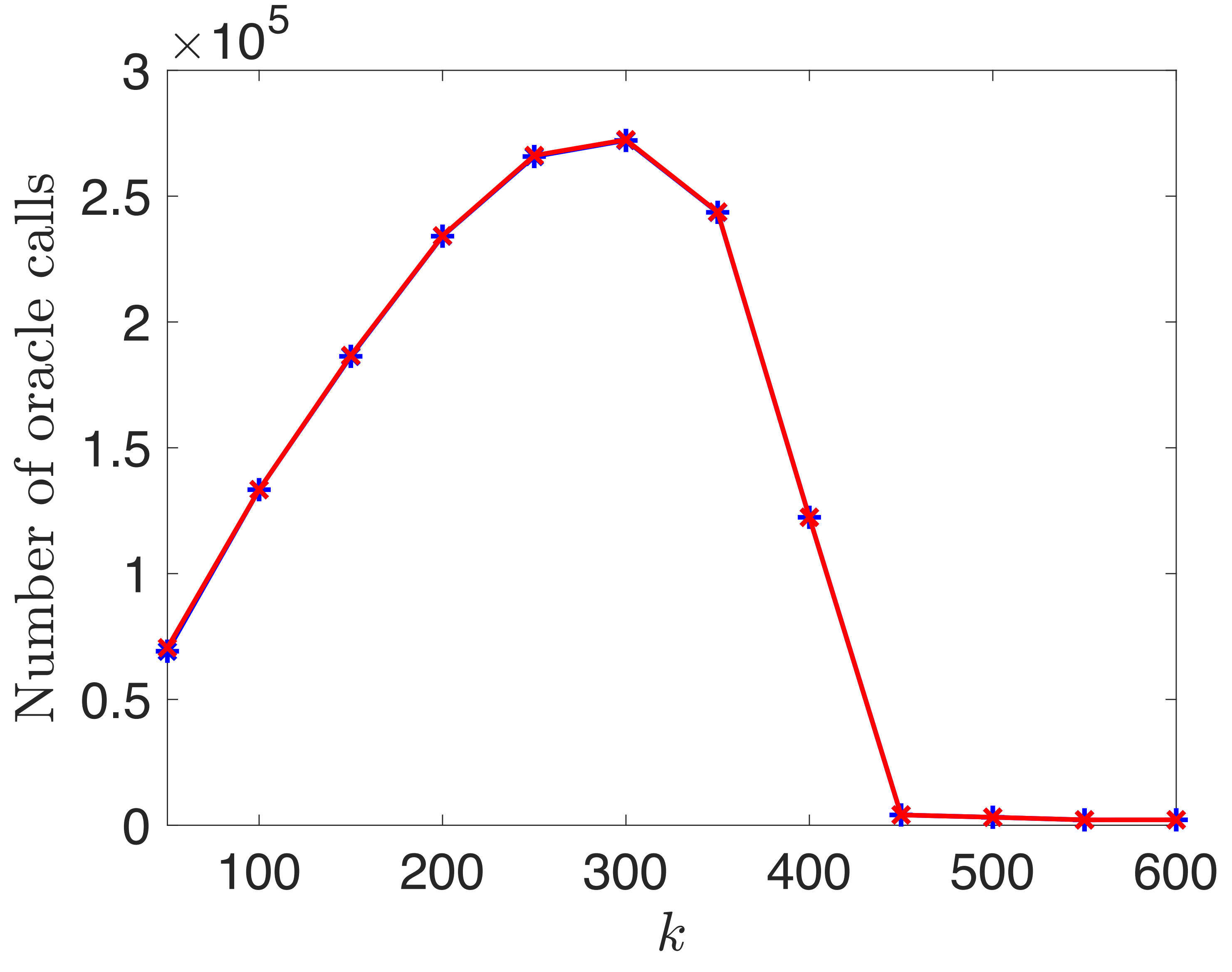


DPP-based summarization

- Dataset: Census Income dataset [Dua et al, 2017], $5\,000$ records, with 14 attributes (race, gender, income, etc)
- Objective (non-monotone submodular): $f(S) = \log\det(L_S)$ where L_S is principal submatrix of L indexed by S
- Sensitive attribute and bounds: race, $\ell_c = \lfloor 0.9 \frac{|V_c|}{|V|} k \rfloor, u_c = \lceil 1.1 \frac{|V_c|}{|V|} k \rceil$







Conclusion

✓ First streaming algorithms for fair submodular maximisation

✓ Price of fairness is limited

✓ Explicitly imposing fairness is necessary