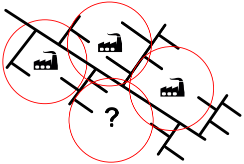
Very Fast Streaming Submodular Function Maximization

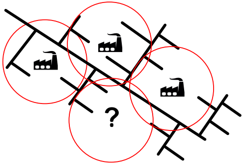
Sebastian Buschjäger, Philipp Jan-Honysz, Lukas Pfahler, and Katharina Morik

13th - 17th September 2021

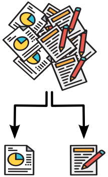
TU Dortmund University - Artifical Intelligence Group - Collaborative Research Center 876



(a) Facility location / coverage

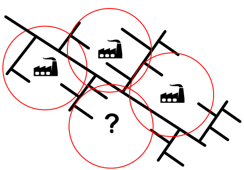


(a) Facility location / coverage

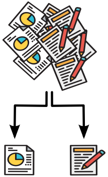


(b) Summarization

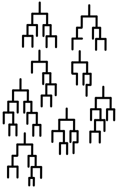
Submodular Function Maximization



(a) Facility location / coverage

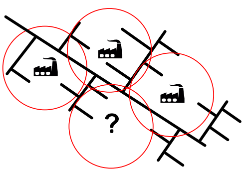


(b) Summarization

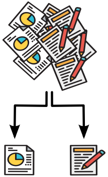


(c) Determinantal Point Processes

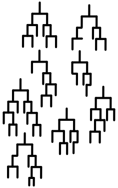
Submodular Function Maximization



(a) Facility location / coverage



(b) Summarization



(c) Determinantal Point Processes

$$\max_{S \subseteq V, |S| \leq K} f(S)$$

where $f\colon 2^V \to \mathbb{R}_{\geq 0}$ is a submodular set function

Gain Let $f: V \to \mathbb{R}$ and let $e \in V$ and $S \subseteq V$:

$$\Delta_f(e|S) = f(S \cup \{e\}) - f(S)$$

Gain Let $f: V \rightarrow \mathbb{R}$ and let $e \in V$ and $S \subseteq V$:

$$\Delta_f(e|S) = f(S \cup \{e\}) - f(S)$$

Submodularity f is submodular iff for all $A \subseteq B \subseteq V$ and $e \in V \setminus B$ it holds that

$$\Delta_f(e|A) \geq \Delta_f(e|B)$$

Gain Let $f : V \to \mathbb{R}$ and let $e \in V$ and $S \subseteq V$:

$$\Delta_f(e|S) = f(S \cup \{e\}) - f(S)$$

Submodularity f is submodular iff for all $A \subseteq B \subseteq V$ and $e \in V \setminus B$ it holds that

$$\Delta_f(e|A) \geq \Delta_f(e|B)$$

Example

Often V is big: Stream V and consume one $e \in V$ at a time

Algorithm	Approximation Ratio	Memory	Queries per Element	Stream	Ref.
Greedy	$1 - 1/\exp(1)$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✗	[23]
StreamGreedy	$1/2 - \varepsilon$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✗	[13]
PreemptionStreaming	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✓	[4]
IndependentSetImprovement	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	[8]
Sieve-Streaming	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[2]
Sieve-Streaming++	$1/2 - \varepsilon$	$\mathcal{O}(K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[16]
Salsa	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	(✓)	[24]
QuickStream	$1/(4c) - \varepsilon$	$\mathcal{O}(cK \log K \log(1/\varepsilon))$	$\mathcal{O}(\lceil 1/c \rceil + c)$	✓	[18]
ThreeSieves	$(1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	this paper

Often V is big: Stream V and consume one $e \in V$ at a time

Algorithm	Approximation Ratio	Memory	Queries per Element	Stream	Ref.
Greedy	$1 - 1/\exp(1)$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✗	[23]
StreamGreedy	$1/2 - \varepsilon$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✗	[13]
PreemptionStreaming	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✓	[4]
IndependentSetImprovement	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	[8]
Sieve-Streaming	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[2]
Sieve-Streaming++	$1/2 - \varepsilon$	$\mathcal{O}(K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[16]
Salsa	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	(✓)	[24]
QuickStream	$1/(4c) - \varepsilon$	$\mathcal{O}(cK \log K \log(1/\varepsilon))$	$\mathcal{O}(\lceil 1/c \rceil + c)$	✓	[18]
ThreeSieves	$(1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	this paper

Streaming Submodular Function Maximization: Available algorithms

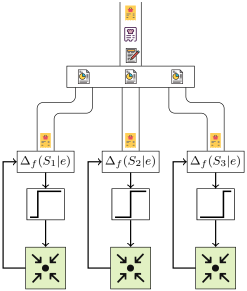
Often V is big: Stream V and consume one $e \in V$ at a time

Algorithm	Approximation Ratio	Memory	Queries per Element	Stream	Ref.
Greedy	$1 - 1/\exp(1)$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✗	[23]
StreamGreedy	$1/2 - \varepsilon$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✗	[13]
PreemptionStreaming	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✓	[4]
IndependentSetImprovement	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	[8]
Sieve-Streaming	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[2]
Sieve-Streaming++	$1/2 - \varepsilon$	$\mathcal{O}(K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[16]
Salsa	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	(✓)	[24]
QuickStream	$1/(4c) - \varepsilon$	$\mathcal{O}(cK \log K \log(1/\varepsilon))$	$\mathcal{O}(\lceil 1/c \rceil + c)$	✓	[18]
ThreeSieves	$(1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	this paper

Often V is big: Stream V and consume one $e \in V$ at a time

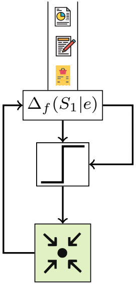
Algorithm	Approximation Ratio	Memory	Queries per Element	Stream	Ref.
Greedy	$1 - 1/\exp(1)$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✗	[23]
StreamGreedy	$1/2 - \varepsilon$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✗	[13]
PreemptionStreaming	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✓	[4]
IndependentSetImprovement	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	[8]
Sieve-Streaming	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[2]
Sieve-Streaming++	$1/2 - \varepsilon$	$\mathcal{O}(K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[16]
Salsa	$1/2 - \varepsilon$	$\mathcal{O}(K \log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	(✓)	[24]
QuickStream	$1/(4c) - \varepsilon$	$\mathcal{O}(cK \log K \log(1/\varepsilon))$	$\mathcal{O}(\lceil 1/c \rceil + c)$	✓	[18]
ThreeSieves	$(1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	this paper

Streaming Submodular Function Maximization: Sieve Streaming



(a) Sieve Streaming

- Add if $\Delta_f(e|S_i) \geq v_i$
- Use $v_i \in O = \{(1+\varepsilon)^i \mid i \in \mathbb{Z}, m \leq (1+\varepsilon)^i \leq K \cdot m\}$
- Maintain multiple summaries and thresholds
- Memory: $O(K \log K/\varepsilon)$
- Solution: $f(S) \geq (1/2 - \varepsilon) f(OPT)$



(a) Three Sieves

- Add if $\Delta_f(e|S) \geq v$
- Start with a large $v_i \in O$ and gradually decrease it
- Maintain a single summary and threshold
- Memory: $O(K)$
- Solution: $f(S) \geq (1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$

A probabilistic-view of item selection

How to determine when to lower the threshold?

A probabilistic-view of item selection

How to determine when to lower the threshold?

1) $\Delta_f(e|S) \geq v$
Add e to the summary and continue with large threshold.

2) $\Delta_f(e|S) < v$
Do not add e. Would a smaller v have been better?

How to determine when to lower the threshold?

1) $\Delta_f(e|S) \geq v$
Add e to the summary and continue with large threshold.

2) $\Delta_f(e|S) < v$
Do not add e. Would a smaller v have been better?

Differently put What is the probability $p(e|f, S, v)$ to find an item e in the future that exceeds v?

A probabilistic-view of item selection

How to determine when to lower the threshold?

1) $\Delta_f(e|S) \geq v$
Add e to the summary and continue with large threshold.

2) $\Delta_f(e|S) < v$
Do not add e. Would a smaller v have been better?

Differently put What is the probability $p(e|f, S, v)$ to find an item e in the future that exceeds v?

Basic idea Estimate $p(e|f, S, v)$ on the fly and lower threshold once its unlikely to be out-valued

But p is estimated from data and has its own confidence.

A probabilistic-view of item selection

But p is estimated from data and has its own confidence.

1) $\Delta_f(e|S) \geq v$
Add e to the summary and S changes. Start new estimation of $p(e|f, S, v)$

2) $\Delta_f(e|S) < v$
Do not add e and update estimate of $p(e|f, S, v)$.

A probabilistic-view of item selection

But p is estimated from data and has its own confidence.

1) $\Delta_f(e|S) \geq v$
Add e to the summary and S changes. Start new estimation of $p(e|f, S, v)$

2) $\Delta_f(e|S) < v$
Do not add e and update estimate of $p(e|f, S, v)$.

We only observe rejections for p

Rule of Three The $\alpha = 0.95$ - confidence interval of p after T negative and **no positive** coin flips is

$$p \in \left[0, \frac{3}{T}\right]$$

The Rule of Three and Three Sieves

Rule of Three The $\alpha = 0.95$ - confidence interval of p after T negative and **no positive** coin flips is

$$p \in \left[0, \frac{3}{T}\right]$$

Example After $T = 1000$ rejections the 95% confidence interval is $p \in [0, 0.003]$

The Rule of Three and Three Sieves

Rule of Three The $\alpha = 0.95$ - confidence interval of p after T negative and **no positive** coin flips is

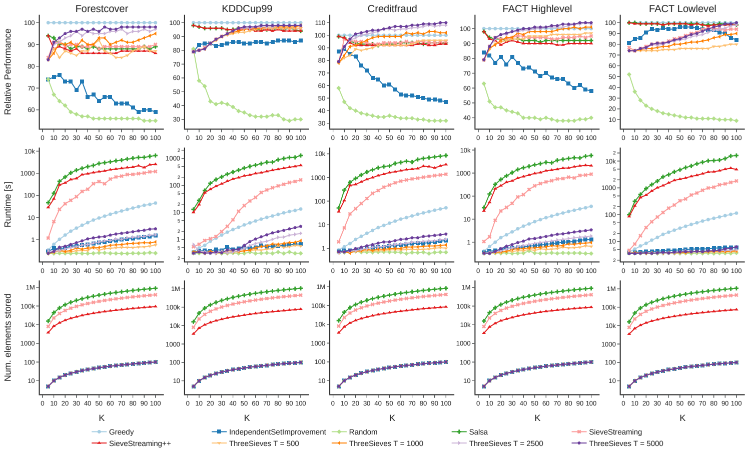
$$p \in \left[0, \frac{3}{T}\right]$$

Example After $T = 1000$ rejections the 95% confidence interval is $p \in [0, 0.003]$

Algorithmic idea

- Start with largest available threshold and set $t = 0$
- If $\Delta_f(e|S) \geq v$ add e to S and set $t = 0$
- If $\Delta_f(e|S) < v$ increase t by one
- If $t \geq T$ lower threshold by rule-of-three and set $t = 0$

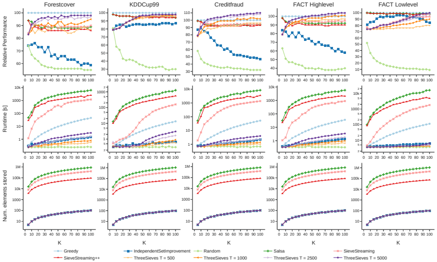
Experiments



Very Fast Streaming Submodular Function Maximization under Memory Constraints

Algorithm	Approximation Ratio	Memory	Queries per Element	Stream	Ref.
Greedy	$1-1/\exp(1)$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✗	[23]
StreamGreedy	$1/2-\varepsilon$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✗	[13]
PreemptionStreaming	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(K)$	✓	[4]
IndependentSetImprovement	$1/4$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	[8]
Sieve-Streaming	$1/2-\varepsilon$	$\mathcal{O}(K\log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[2]
Sieve-Streaming++	$1/2-\varepsilon$	$\mathcal{O}(K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	✓	[16]
Salsa	$1/2-\varepsilon$	$\mathcal{O}(K\log K/\varepsilon)$	$\mathcal{O}(\log K/\varepsilon)$	(✓)	[24]
QuickStream	$1/(4c)-\varepsilon$	$\mathcal{O}(cK\log K\log(1/\varepsilon))$	$\mathcal{O}(\lceil 1/c\rceil+c)$	✓	[18]
ThreeSieves	$(1-\varepsilon)(1-1/\exp(1))$ with prob. $(1-\alpha)^K$	$\mathcal{O}(K)$	$\mathcal{O}(1)$	✓	this paper

(a) Theoretical Results



(b) Experimental results

Submodular Streaming Maximization https://github.com/sbuschjaeger/SubmodularStreamingMaximization/

"Very Fast Streaming Submodular Function Maximization" @ ECML/PKDD 2021