Optimal Matrix Sketching over Sliding Windows

Hanyan Yin[1], Dongxie Wen[1], Jiajun Li[1], Zhewei Wei[1], Xiao Zhang[1],
Zengfeng Huang[2], Feifei Li[3]
[1] *Gaoling School of Artificial Intelligence, Renmin University of China*
[2] *School of Data Science, Fudan University* [3] *Alibaba Group*
Contact: zhewei@ruc.edu.cn

Motivation and Problem Statement

Matrix Sketching

- Many modern datasets are vast and rapid data streams, while computational and storage resources are limited.
- Matrix sketching: approximate large matrix $\boldsymbol{A} \in \mathbb{R}^{n \times d}$ with $\boldsymbol{B} \in \mathbb{R}^{\ell \times d}$, $\ell \ll n$.
- Row-update stream: each update receives $\boldsymbol{a}_i$, a row of $\boldsymbol{A}$.
- Covariance error: $\|\boldsymbol{A}^\top\boldsymbol{A} - \boldsymbol{B}^\top\boldsymbol{B}\|_2 \le \varepsilon\|\boldsymbol{A}\|_F^2$.
- Frequent Direction(FD)[Liberty 2013]: $\boldsymbol{B} \in \mathbb{R}^{\ell \times d}$ s.t. $\|\boldsymbol{A}^\top\boldsymbol{A} - \boldsymbol{B}^\top\boldsymbol{B}\|_2 \le \frac{1}{\ell}\|\boldsymbol{A}\|_F^2$.

Matrix Sketching over Sliding Windows

- Maintain (approximately) $\boldsymbol{A}_W^T\boldsymbol{A}_W$ for time/sequence-based window W.
- Applications: sliding window PCA; event detection; fault monitoring; differential privacy; online learning.
- Existing algorithms for matrix sketching over sliding windows were sub-optimal in terms of space complexity.

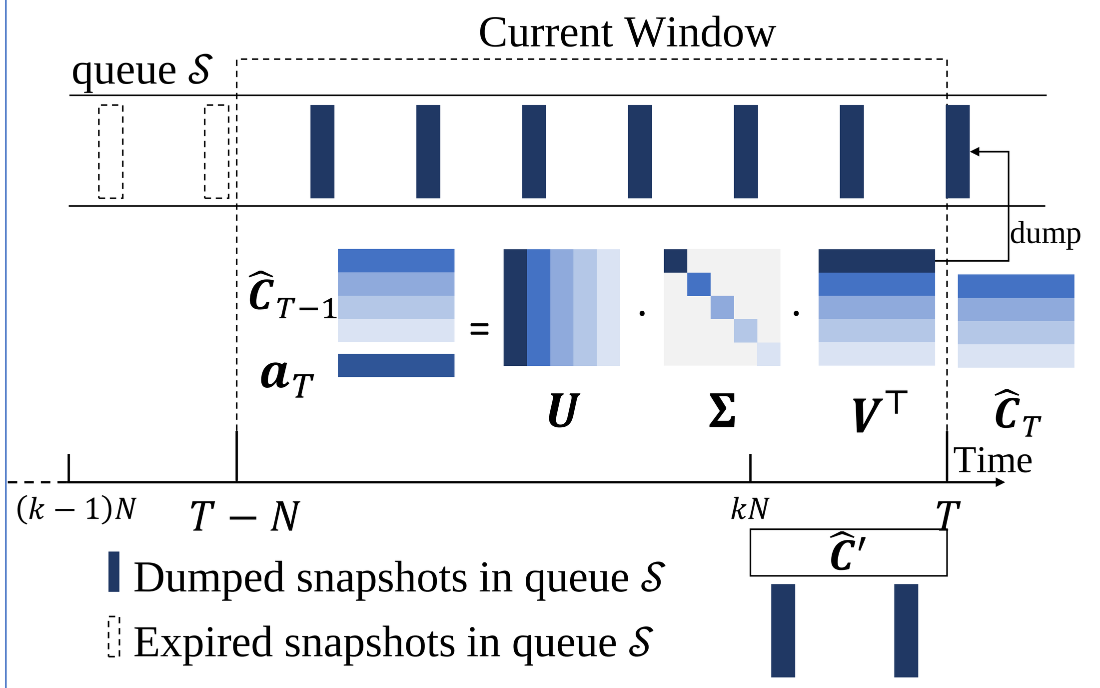
Sequence-based Normalized Matrix Sketching

Inspiration:

- Connection btw matrix sketching and item frequency problem [Liberty 2013]
- Extension of item frequency to sliding window model [Lee 2006]

Dump Snapshot Frequent Directions (DS-FD)

- Work for $\|\boldsymbol{a}_i\|_2 = 1$ and window size N.
- Maintain FD sketches $\boldsymbol{C}, \boldsymbol{C}'$ and queues $\mathcal{S}, \mathcal{S}'$.
- Expire the outdated elements in queues.
- Perform FD update $[\boldsymbol{\Sigma}, \boldsymbol{V}^\top] = \mathrm{FD}(\boldsymbol{C}, \boldsymbol{a}_i)$.
- If the top singular value $\sigma_1 > \theta = \varepsilon N$, save the top singular vector $\sigma_1 \cdot \boldsymbol{v}_1$ and current timestamp $T = i$ into queues.
- Restart every N steps.



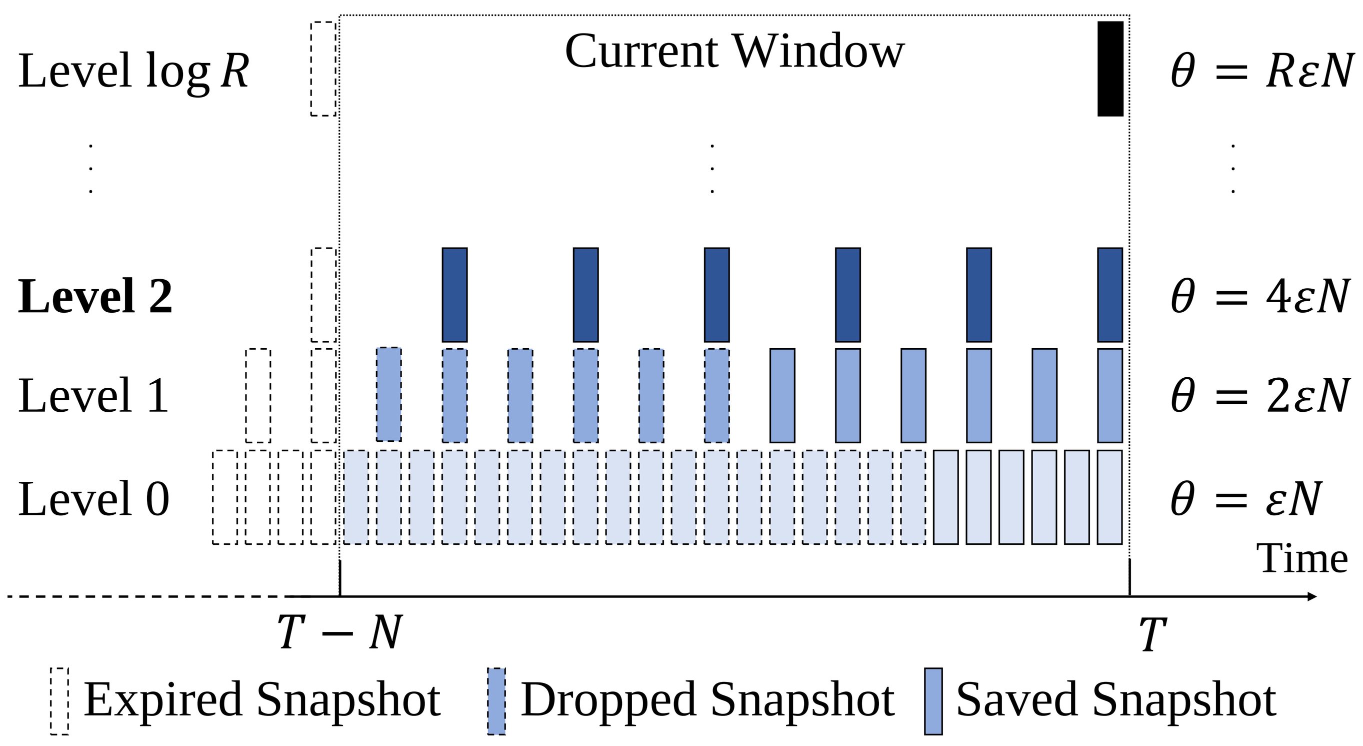
- Error bound: $\|\boldsymbol{A}_W^\top\boldsymbol{A}_W - \boldsymbol{B}_W^\top\boldsymbol{B}_W\|_2 \le \varepsilon N$
- Space: $O\left(\frac{d}{\varepsilon}\right)$
- Update time: $O(d\ell + \ell^3)$, $\ell = \min\left(\left\lceil\frac{1}{\varepsilon}\right\rceil, d\right)$

Unnormalized/Time-based Matrix Sketching

- Unnormalized rows: Work for $\|\boldsymbol{a}_i\| \in [1, R]$.

Insights:

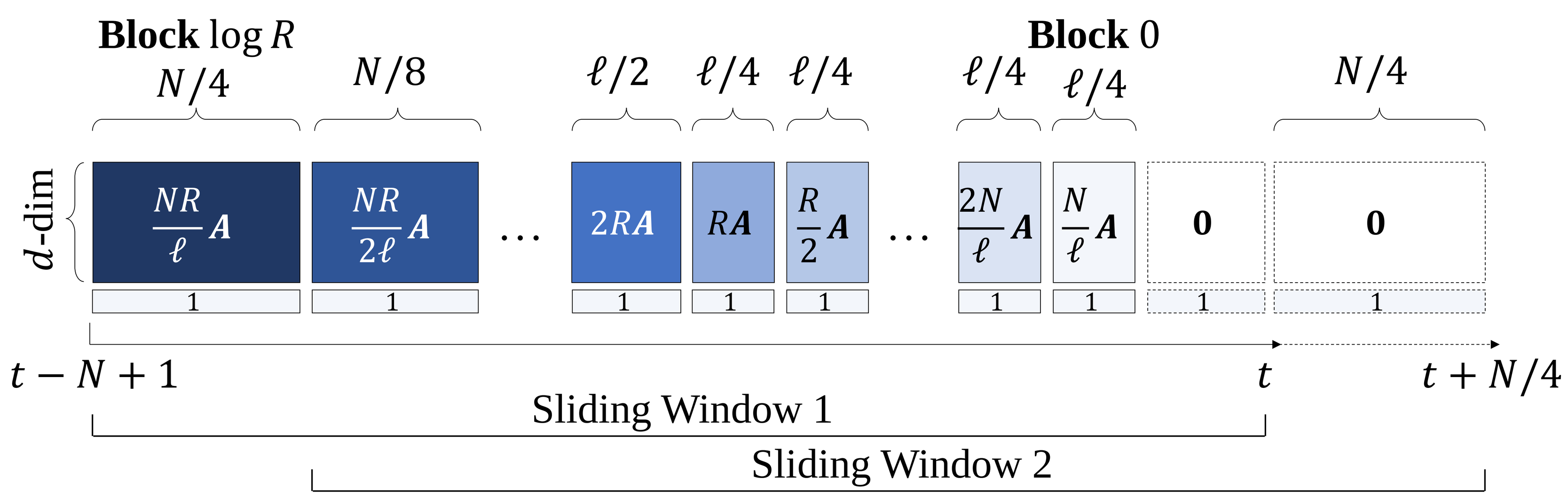
- Multiple DS-FD sketches with exponentially incremental dump threshold $\theta = \varepsilon N, 2\varepsilon N, \ldots, \varepsilon NR$.
- Retain only the most recent $O\left(\frac{1}{\varepsilon}\right)$ snapshots in each DS-FD sketch.



- Error bound: $\|\boldsymbol{A}_W^\top\boldsymbol{A}_W - \boldsymbol{B}_W^\top\boldsymbol{B}_W\|_2 \le \varepsilon\|\boldsymbol{A}_W\|_F^2$
- Space: $O\left(\frac{d}{\varepsilon}\log R\right)$

Space Complexity of DS-FD is Optimal

Theorem 6.1 Any deterministic algorithm which provides the covariance error bound $\|\boldsymbol{A}_W^\top\boldsymbol{A}_W - \boldsymbol{B}_W^\top\boldsymbol{B}_W\|_2 \le O(\varepsilon)\|\boldsymbol{A}_W\|_F^2$ must uses $O\left(\frac{d}{\varepsilon}\log R\right)$ bits.



Proof main idea: construct difficult adversarial input against algorithms.

sketch κ	Sequence-based		Time-based	
	normalized	unnormalized	normalized	unnormalized
DS-FD (This paper)	$O\left(\frac{d}{\varepsilon}\right)$	$O\left(\frac{d}{\varepsilon}\log R\right)$	$O\left(\frac{d}{\varepsilon}\log \varepsilon N\right)$	$O\left(\frac{d}{\varepsilon}\log \varepsilon NR\right)$
Lower bound (This paper)	$\Omega\left(\frac{d}{\varepsilon}\right)$	$\Omega\left(\frac{d}{\varepsilon}\log R\right)$	$\Omega\left(\frac{d}{\varepsilon}\log \varepsilon N\right)$	$\Omega\left(\frac{d}{\varepsilon}\log \varepsilon NR\right)$

Conclusion: the space complexity of DS-FD is optimal.

Experiments and Analysis

Baselines:

Sketches	Update	Space	Window
Sampling	$\frac{d}{\varepsilon^2}\log\log NR$	$\frac{d}{\varepsilon^2}\log NR$	Sequence & time
LM-FD	$d\log\varepsilon NR$	$\frac{d}{\varepsilon^2}\log\varepsilon NR$	Sequence & time
DI-FD	$\frac{d}{\varepsilon}\log\frac{R}{\varepsilon}$	$\frac{Rd}{\varepsilon}\log\frac{R}{\varepsilon}$	Sequence
DS-FD (Our Work)	$\left(\frac{d}{\varepsilon} + \frac{1}{\varepsilon^3}\right)\log\varepsilon NR$	$\frac{d}{\varepsilon}\log\varepsilon NR$	Sequence & time

Observations:

- DS-FD provides better space-error tradeoffs than Sampling, LM-FD and DI-FD.
- Empirical errors are always lower than the theoretical bound, i.e., $\|\boldsymbol{A}_W^\top\boldsymbol{A}_W - \boldsymbol{B}_W^\top\boldsymbol{B}_W\|_2 \le \varepsilon\|\boldsymbol{A}_W\|_F^2$.
- DS-FD effectively balances update and query times.

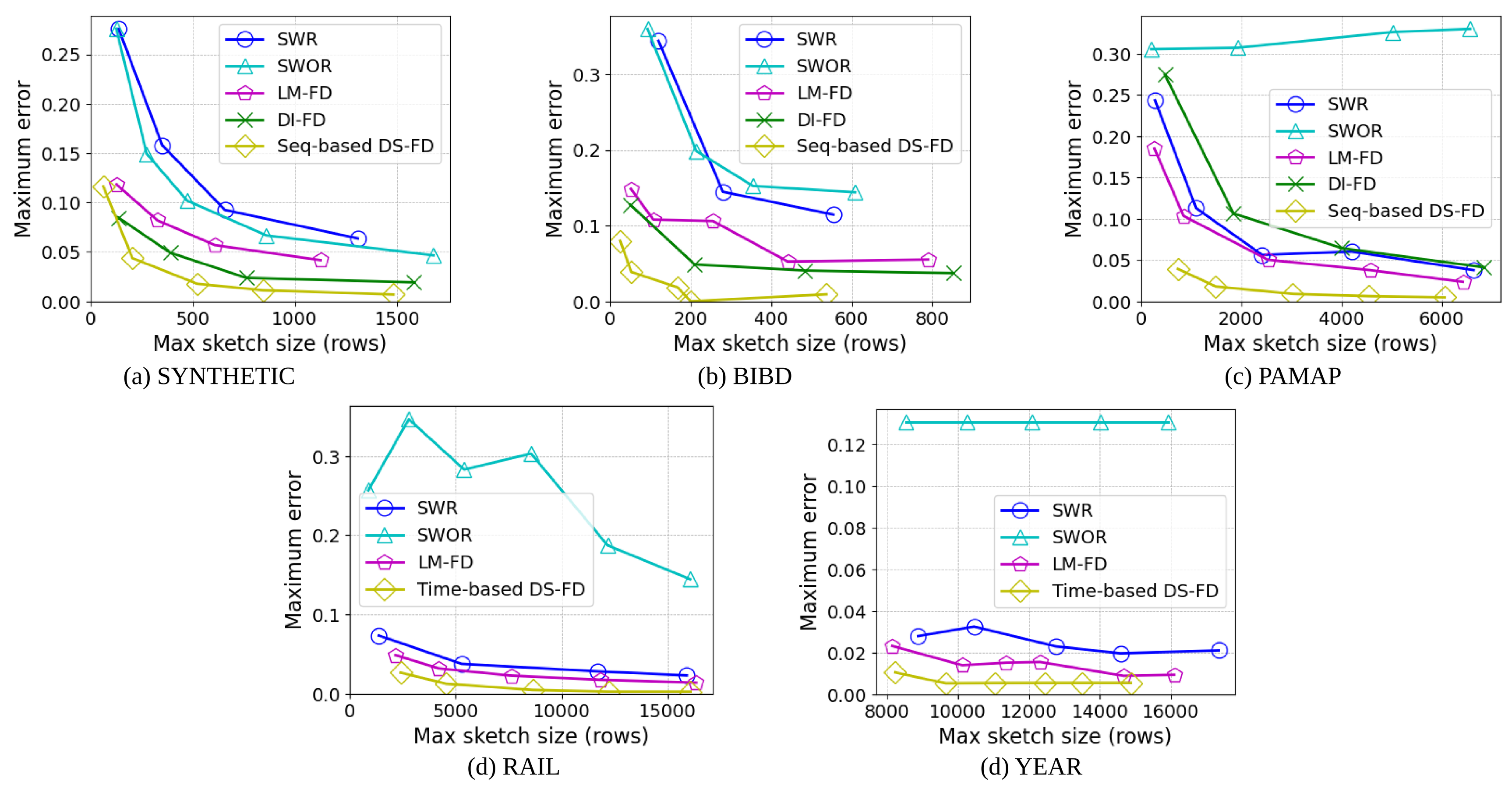


Table 4: Update time and query time of all methods with a relative error bound of $\varepsilon = 1/100$ on the BIBD dataset.

Time(ms)	Update time	Query Time
SWR	65.722	157.500
SWOR	3.143	291.936
LM-FD	0.061	3599.310
DI-FD	2.428	59.904
DS-FD	1.053	27.655