ETH zürich

Efficiently Learning Fourier-Sparse Set Functions

Andisheh Amrollahi[1], Amir Zandieh[2], Andreas Krause[1], and Mikhail Kapralov[2]

[1]: ETH Zurich

[2]: EPFL

Learning & Adaptive Systems

Motivation

- Given: oracle access to some set function $\mathrm{G}: 2^V \to \mathbb{R}$?
- Goal: “learn” G ?
- Assumptions?

Sparsity in the Fourier domain

Fourier transforms on $\mathbb{Z}_N$

- $\mathbb{Z}_N = \{0, \dots, N-1\}$ (integers modulo N)
- Let $x_t : \mathbb{Z}_N \to \mathbb{C}$. The Fourier transform $\hat{x}_f : \mathbb{Z}_N \to \mathbb{C}$ is:

$$\hat{x}_f = \frac{1}{\sqrt{N}} \sum_{t=0}^{N-1} x_t e^{if\frac{2\pi}{N}t}, \qquad t, f \in \mathbb{Z}_N$$

- What if we know that $\hat{x}_f$ is sparse i.e there are only k non zero coefficients $\hat{x}_f$?

Computational and sampling complexities of the sparse Fourier transform

Algorithm	Time	Samples	Sparsity (k)	Model
[SODA'12] (Approx. Sparse)	$O(\log n \sqrt{nk \log n})$	$O(\log n \sqrt{nk \log n})$	$O(n/\log n)$	Worst Case
[SODA'12] (Approx. Sparse)	$O\left(\log n \sqrt[3]{k^2 n \log n}\right)$	$O\left(\log n \sqrt[3]{k^2 n \log n}\right)$	$O(n/\sqrt{\log n})$	Worst Case
[STOC'12] (Exactly Sparse)	$O(k \log n)$	$O(k \log n)$	$O(n)$	Worst Case
[STOC'12] (Approx. Sparse)	$O(k \log n \log(n/k))$	$O(k \log n \log(n/k))$	$O(n)$	Worst Case
[Allerton'13] (Exactly Sparse)	$O(k \log k + k(\log\log n)^{O(1)})$	$O(k)$	$O(n)$	Worst Case
[Allerton'13] (Exactly Sparse)	$O(k \log k)$	$O(k)$	$O(\sqrt{n})$	Average Case
[Allerton'13] (Approx. Sparse)	$O(k \log^2 n)$	$O(k \log n)$	$\Theta(\sqrt{n})$	Average Case
[SODA'14] (Approx. Sparse)	$O(k \log^2 n (\log\log n)^{O(1)})$	$O(k \log n (\log\log n)^{O(1)})$	$O(n)$	Worst Case
[FOCS'14] (Approx. Sparse)	$O(n \log^{O(1)} n)$	$O(k \log n)$	$O(n)$	Worst Case
Lower Bounds Exactly Sparse: Approx. Sparse:		$\Omega(k)$ $\Omega(k \log(n/k) / \log\log n)$		

Fourier transform on $\mathbb{Z}_2^n$ (Walsh-Hadamard transform)

- $\mathbb{Z}_2 = \{0,1\}$

$$\hat{x}_f = \frac{1}{\sqrt{2}} \sum_{t \in \mathbb{Z}_2} x_t e^{if\frac{2\pi}{2}t} = \sum_{t \in \mathbb{Z}_2} x_t (-1)^{ft}$$

$$\Psi_{\mathrm{f}}(\mathrm{t}) := (-1)^{<f,t>}$$

- $\mathbb{Z}_2^n = \{0,1\}^n$, $x_t : \mathbb{Z}_2^n \to \mathbb{R}$, $\hat{x}_f : \mathbb{Z}_2^n \to \mathbb{R}$

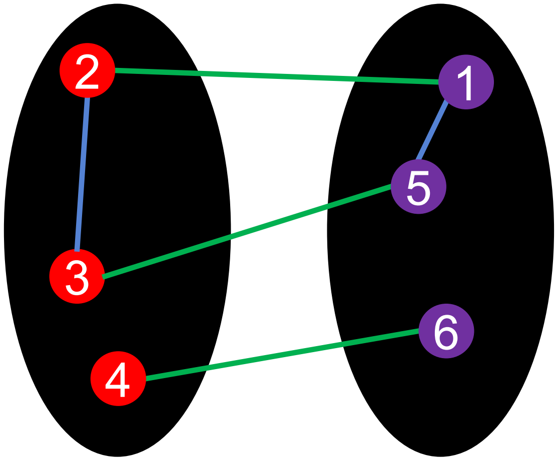
$$\hat{x}_f = \frac{1}{\sqrt{2^n}} \sum_{t \in \mathbb{Z}_2^n} x_t (-1)^{<f,t>}, f, t \in \mathbb{Z}_2^n$$

$$x_t = \frac{1}{\sqrt{2^n}} \sum_{f \in \mathbb{Z}_2^n} \hat{x}_f \boxed{(-1)^{<f,t>}}, f, t \in \mathbb{Z}_2^n$$

Examples: Cut functions

- Cut functions: $\mathrm{x_t}: \mathbb{Z}_2^6 \to \mathbb{R}$

$t = (0, 1, 1, 1, 0, 0),\ x_t = 3$



$A = \{2, 3, 4\}$ $V \backslash A = \{1, 5, 6\}$

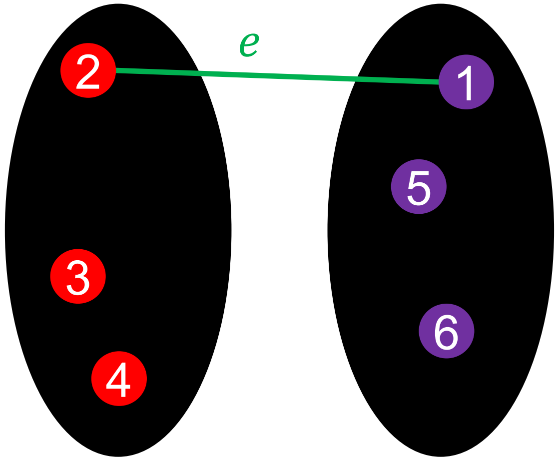
Examples: Cut functions

- Cut functions: $x_t\colon \mathbb{Z}_2^6 \to \mathbb{R}$

$$x_t = \frac{1}{2}\left(1 - (-1)^{<f_e, t>}\right)$$

$$f_e = (1, 1, 0, 0, 0, 0)$$

- $t_A = (0, 1, 1, 1, 0, 0)$



$A = \{2, 3, 4\}$ $\quad V \backslash A = \{1, 5, 6\}$

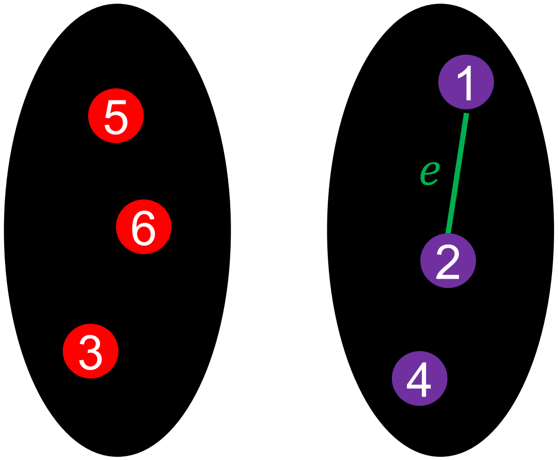
Examples: Cut functions

- Cut functions: $\mathrm{x_t}: \mathbb{Z}_2^6 \to \mathbb{R}$

$$x_t = \frac{1}{2}(1 + (-1)^{<f_e, t>})$$

$$f_e = (1, 1, 0, 0, 0, 0)$$

- $t_A = (0, 0, 1, 1, 0, 1)$



$A = \{5, 6, 3\}$ $V \backslash A = \{1, 2, 4\}$

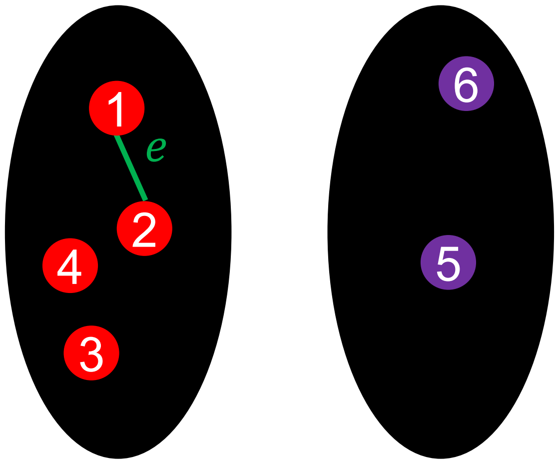
Examples: Cut functions

- Cut functions: $x_t\colon \mathbb{Z}_2^6 \to \mathbb{R}$

$$x_t = \frac{1}{2}\left(1 - (-1)^{<f_e, t>}\right)$$

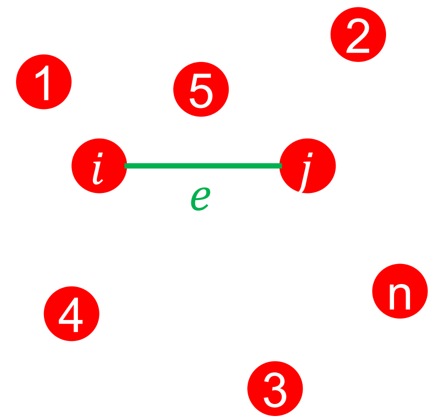
$$f_e = (1, 1, 0, 0, 0, 0)$$

- $t_A = (1, 1, 1, 1, 0, 0)$



$A = \{1, 2, 3, 4\}$ $V \backslash A = \{5, 6\}$

Examples: Cut functions



- Cut functions: $x_t\colon \mathbb{Z}_2^n \to \mathbb{R}$

$$x_t = \sum_{e \in E} \frac{1}{2}\left(1 + (-1)^{<f_e, t>}\right)$$

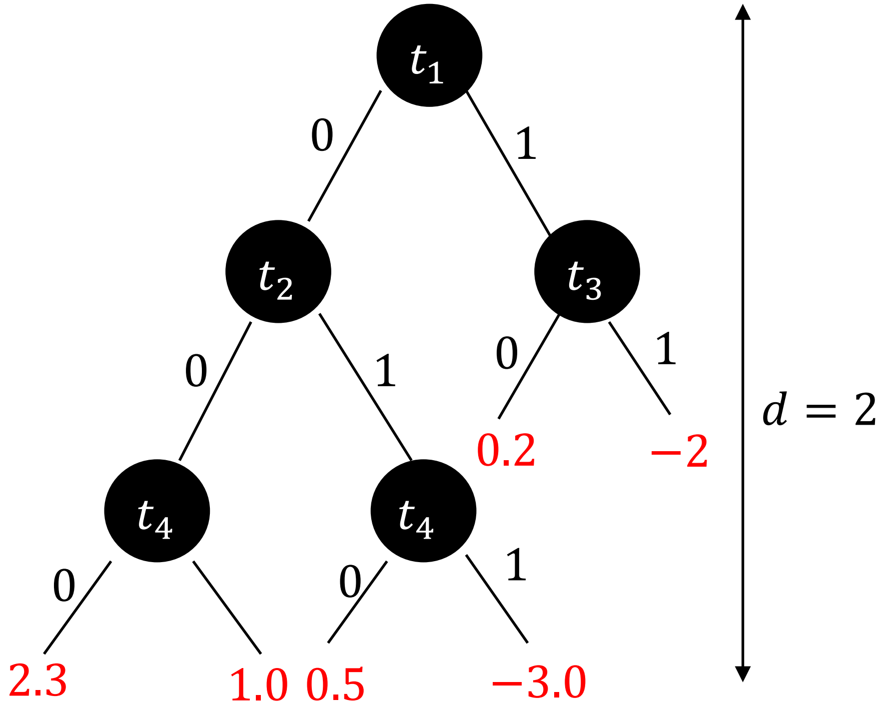
$$f_e = (0, \ldots, 1, \ldots, 1, \ldots, 0)$$

i j

Sparsity $(k) \leq |E| + 1$
Frequency degree $(d) \leq 2$

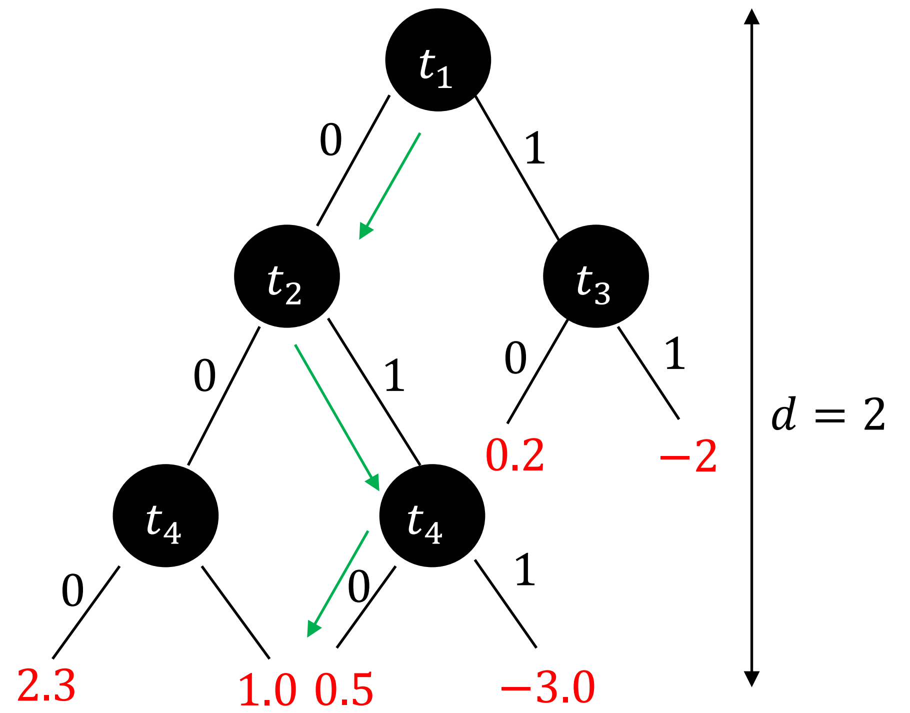
Examples: Decision trees

- $\mathrm{x_t}: \mathbb{Z}_2^4 \to \mathbb{R}$



Examples: Decision trees

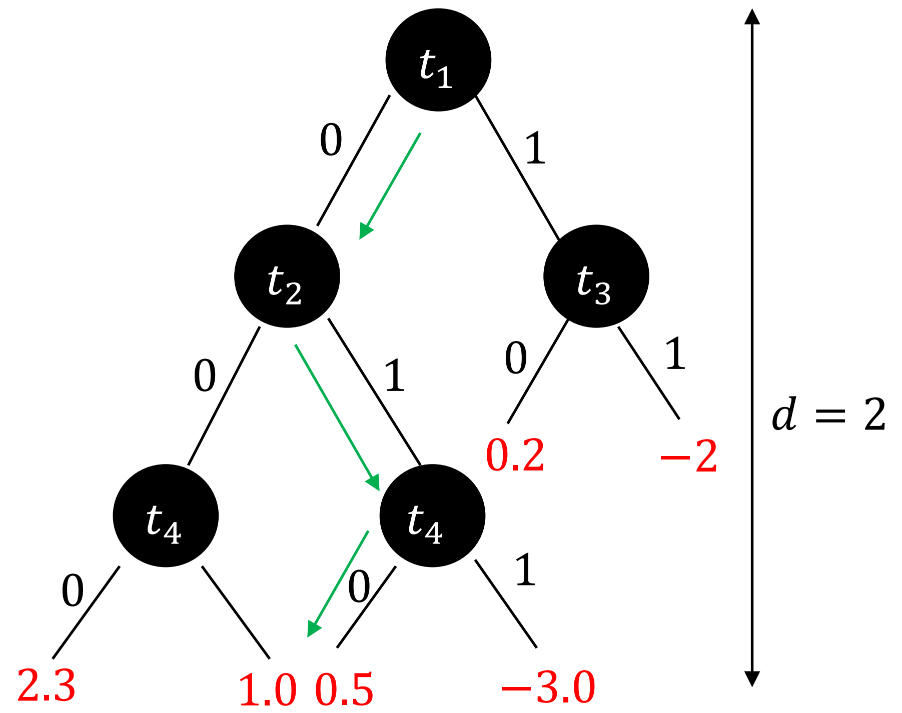
- $x_t : \mathbb{Z}_2^4 \to \mathbb{R}$
- $t = (0,1,1,0),\ x_t = 0.5$



Examples: Decision trees

- $x_t: \mathbb{Z}_2^4 \to \mathbb{R}$
- $t = (0,1,1,0)$, $x_t = 0.5$
- Sparsity(k) $\leq 2^{depth}$

  Frequency degree (d) $\leq depth$



Examples: Other

- Frequency degree d captures order of dependency
- Sums of set functions each dependent on at most d variables

Results: exactly sparse

	Our results: Computation	Our results: Samples	Previous results: Computation	Previous results: Samples
Exactly sparse	$O(n\,k \log k)$	$O(nk)$	$O(n\,k \log k)$ [1]	$O(nk)$ [1]
Assumption	**None**		Random support and $k = n^{\alpha}, 0 < \alpha < 1$	

	Our results: Computation	Our results: Samples	Previous results: Computation	Previous results: Samples
Exactly sparse+ low degree	$\boldsymbol{O(n\,k\,\log k + n\,k\,d \log n)}$	$\boldsymbol{O(k\,d \log n)}$	$\Omega(n^d)$ [2]	$O(k\,d^4 \log(n)^4)$ [2] $O(k \log k\ \ d \log n)$ [3]
Assumption	None		None	

[1]: R. Scheibler et al. , "A Fast Hadamard Transform for Signals With Sublinear Sparsity in the Transform Domain," in *IEEE Transactions on Information Theory*, 2015.
[2]: S. Peter, and A. Krause. "Learning Fourier sparse set functions." *Artificial Intelligence and Statistics,* 2012.
[3]: I. Haviv, and O. Regev. “The restricted isometry property of subsampled Fourier matrices”. *Geometric Aspects of Functional Analysis, 2017*

Results: approximately sparse

	Our results: Computation	Our results: Samples	Previous results: Computation	Previous results: Samples
Approx. sparse	$\boldsymbol{O(k \log k\ \ n \log n)}$	$O(k\ n \log n)$	$O(k\ n^2)$ [4]	$\boldsymbol{O(kn)}$ [4]
Assumption	**None**		Random, binary valued support and noise is Gaussian	
Approx. sparse + Low degree	$\boldsymbol{O\big(kn \log k + k \log k\ \ d \log n \cdot \log(d \log n)\big)}$	$\boldsymbol{O(kd \log n \cdot \log(d \log n))}$	$\Omega(n^d)$ [2]	$O(k\ d^4 \log(n)^4)$ [2] $O(k \log k\ \ d \log n)$ [3]
Assumptions	None		None	

[2]: P. Sobbe, and A. Krause. "Learning Fourier sparse set functions." *Artificial Intelligence and Statistics,* 2012.
[3]: I. Haviv, and O. Regev. "The restricted isometry property of subsampled Fourier matrices". *Geometric Aspects of Functional Analysis, 2017*
[4]:X. Li, J. K. Bradley, S. Pawar and K. Ramchandran, "The SPRIGHT algorithm for robust sparse Hadamard Transforms," *IEEE International Symposium on Information Theory,* 2014.

Techniques: hashing

$$\sigma = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}_{n \times b}$$

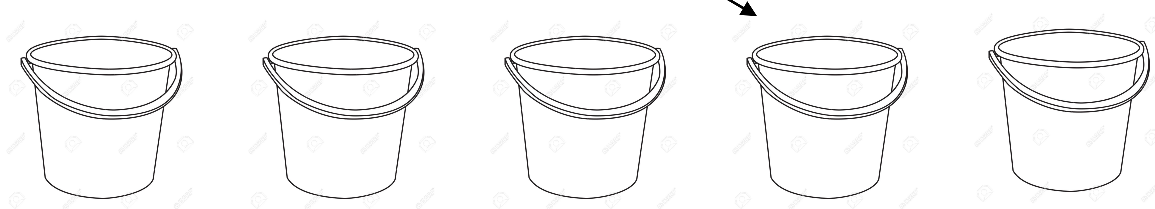
- $\mathrm{x_t}: \mathbb{Z}_2^n \to \mathbb{R}$, $\hat{x}_f: \mathbb{Z}_2^n \to \mathbb{R}$
- $\sigma \in \mathbb{Z}_2^{n \times b}$, $b < n$, elements chosen independently and uniformly at random
- Hash function $h(f) \coloneqq \sigma^T f$. $h: \mathbb{Z}_2^n \to \mathbb{Z}_2^b$ ⟶ Pairwise independent
- $B = 2^b$ buckets ⟶ $B \approx k$ (sparsity)
- $\mathrm{y_t} := \mathrm{x_{\sigma t}}: \mathbb{Z}_2^b \to \mathbb{R}$, $\hat{y}_f: \mathbb{Z}_2^b \to \mathbb{R}$
- $\hat{y}_f = \sum_{\tilde{f}: h(\tilde{f}) = f} \hat{x}_{\tilde{f}}$, $f \in \mathbb{Z}_2^b, \tilde{f} \in \mathbb{Z}_2^b$

Techniques: hashing

$$h(\tilde{f}) = \sigma^\top \tilde{f} = \begin{bmatrix} 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tilde{f}$$

$$\hat{y}_{(1,1,0)} = \hat{x}_{(1,0,0,0,0)} + \hat{x}_{(1,1,0,0,0)} + \hat{x}_{(1,0,1,1,0)}$$

$$all\ \tilde{f}\ s.t\ h(\tilde{f}) = (1,1,0)$$



(0,0,0) (0,0,1) ... (1,1,0) (1,1,1)

Techniques: hashing

- $Let\, a \in \mathbb{Z}_2^n$
- $\mathrm{y_t} := \mathrm{x}_{\sigma \mathrm{t+a}} : \mathbb{Z}_2^b \rightarrow \mathbb{R}$, $\hat{y}_f : \mathbb{Z}_2^b \rightarrow \mathbb{R}$
- $\hat{y}_f = \sum_{\tilde{f}:h(\tilde{f})=f} \hat{x}_f \, (-1)^{<a,f>}$, $f \in \mathbb{Z}_2^b, \tilde{f} \in \mathbb{Z}_2^b$
- Repeat for a fixed σ and different values of a

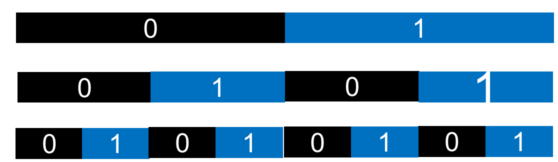
Techniques: Compressive Sensing over finite fields

- Given linear measurements of type $< a_i, f >$ can one recover f ?
- Just set $a_i = e_i = (0, \dots 0\, , 1, 0, \dots, 0)$ for $i \in [n]$

i'th coordinate

$f =$ 

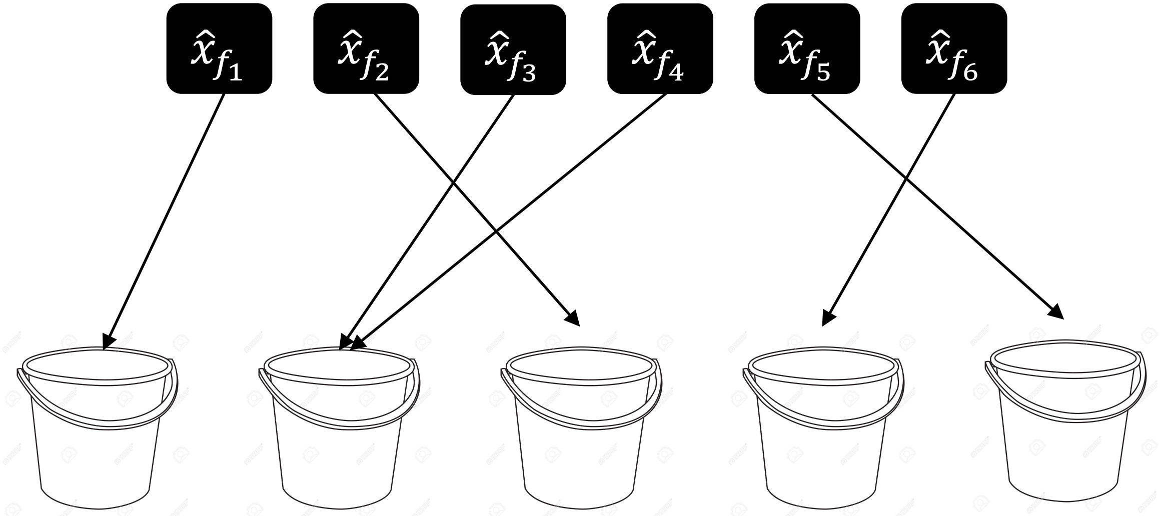
- What if frequency is of degree one ?
  - Binary search: $O(\log n)$ measurements



- What if frequency is of degree d ?
  - Hashing and binary search: $O(d \log n)$ measurements

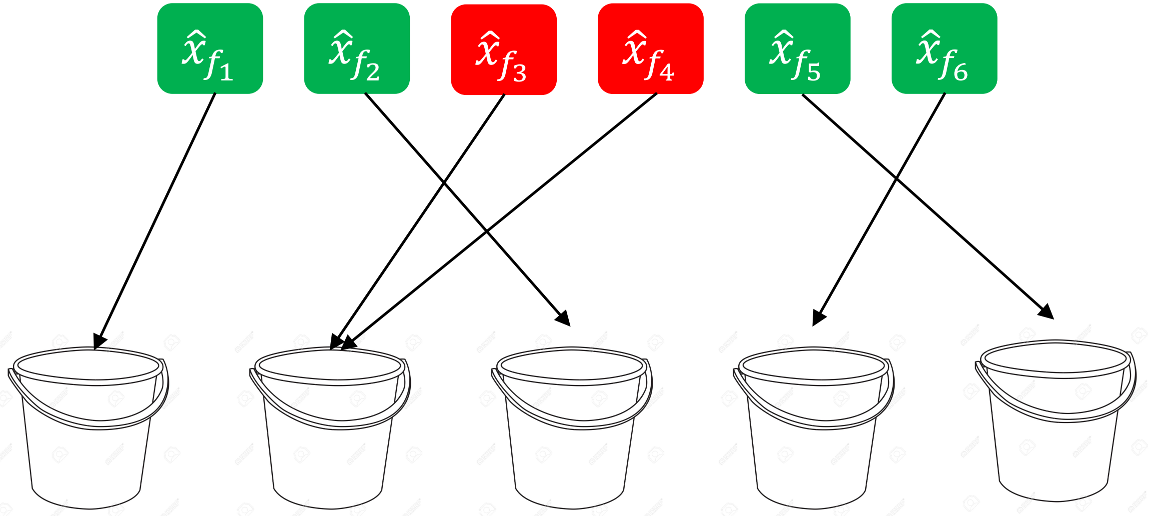
Techniques: peeling

- What if there are collisions?



Techniques: peeling

- What if there are collisions?



Techniques: peeling

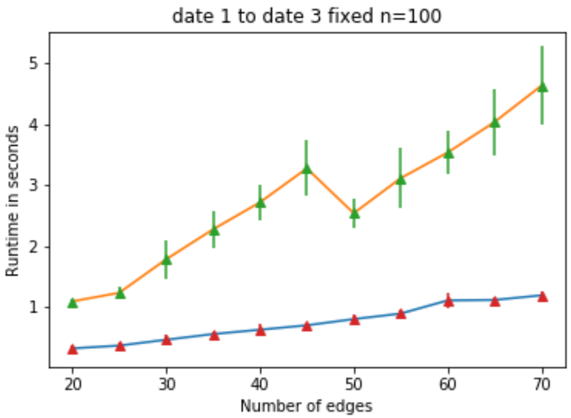
- What if there are collisions?

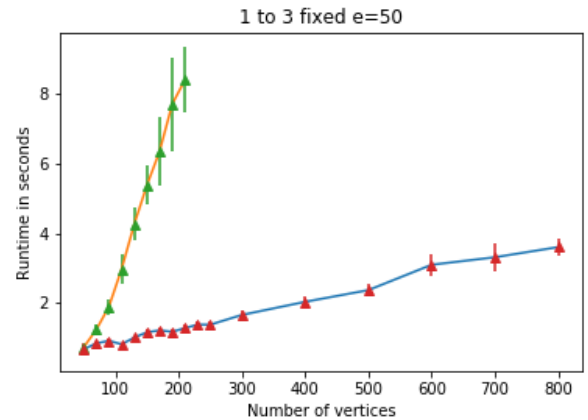
$$\hat{x}_{f_1} \quad \hat{x}_{f_2} \quad \hat{x}_{f_3} \quad \hat{x}_{f_4} \quad \hat{x}_{f_5} \quad \hat{x}_{f_6}$$

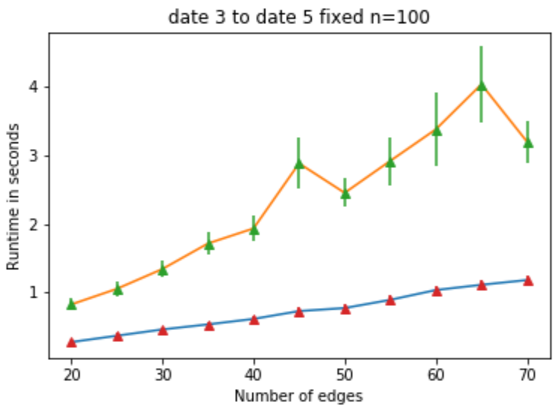
$$- \quad \hat{x}_{f_1} \quad \hat{x}_{f_2} \quad \hat{x}_{f'_3} \quad \hat{x}_{f'_4} \quad \hat{x}_{f_5} \quad \hat{x}_{f_6}$$

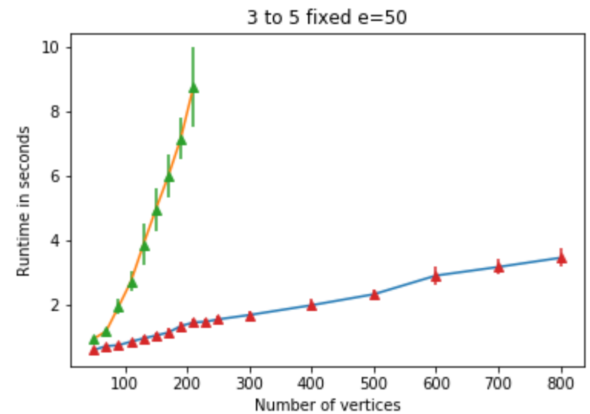
$$\hat{x}_{f_2} \quad \hat{x}_{f_3} \quad \hat{x}_{f'_2} \quad \hat{x}_{f'_3}$$

Experiments









Experiments

No. of vertices	CS method		Our method	
	Runtime	Samples	Runtime	Samples
70	1.14	767	0.85	6428
90	1.88	812	0.92	6490
110	3.00	850	0.82	6491
130	4.31	880	1.01	7549
150	5.34	905	1.16	7942
170	6.13	927	1.22	7942
190	7.36	947	1.18	7271
210	8.24	965	1.28	7271
230	$*$	$*$	1.38	7942
250	$*$	$*$	1.38	7271
300	$*$	$*$	1.66	8051
400	$*$	$*$	2.06	8794
500	$*$	$*$	2.42	8794
600	$*$	$*$	3.10	9646
700	$*$	$*$	3.35	9646
800	$*$	$*$	3.60	9646