Sparse Fourier Transform
(lecture 3)

Michael Kapralov[1]

[1] IBM Watson

MADALGO'15

Given $x \in \mathbb{C}^n$, compute the Discrete Fourier Transform of x:

$$\widehat{x}_i = \sum_{j \in [n]} x_j \omega^{ij},$$

where $\omega = e^{2\pi i/n}$ is the n-th root of unity.

Given $x \in \mathbb{C}^n$, compute the Discrete Fourier Transform of x:

$$\widehat{x}_i = \sum_{j \in [n]} x_j \omega^{ij},$$

where $\omega = e^{2\pi i/n}$ is the n-th root of unity.

Goal: find the top k coefficients of $\widehat{x}$ approximately

In last lecture:

- exactly k-sparse: $O(k \log n)$ runtime and samples

Given $x \in \mathbb{C}^n$, compute the Discrete Fourier Transform of x:

$$\widehat{x}_i = \sum_{j \in [n]} x_j \omega^{ij},$$

where $\omega = e^{2\pi i/n}$ is the n-th root of unity.

Goal: find the top k coefficients of $\widehat{x}$ approximately

In last lecture:

- exactly k-sparse: $O(k \log n)$ runtime and samples
- approximately k-sparse: $O(k \log^2 n)$ runtime and samples

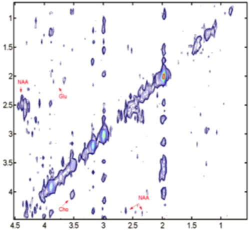
This lecture:

- approximately k-sparse: $O(k \log n)$ samples (optimal)

Sample complexity

Sample complexity=number of samples accessed in time domain.
In some applications at least as important as runtime

Shi-Andronesi-Hassanieh-Ghazi-Katabi-Adalsteinsson'
ISMRM'13

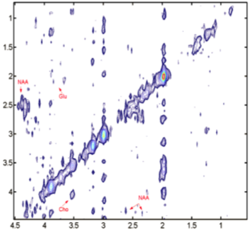


Sample complexity

Sample complexity=number of samples accessed in time domain.

In some applications at least as important as runtime

Shi-Andronesi-Hassanieh-Ghazi-Katabi-Adalsteinsson'
ISMRM'13



Given access to $x \in \mathbb{C}^n$, find $\hat{y}$ such that

$$||\hat{x} - \hat{y}||^2 \le C \cdot \min_{k-\text{sparse } \hat{z}} ||\hat{x} - \hat{z}||^2$$

Use smallest possible number of samples?

Uniform bounds (for all):

Candes-Tao'06
Rudelson-Vershynin'08
Cheraghchi-Guruswami-Velingker'12
Bourgain'14
Haviv-Regev'15

Deterministic, $\Omega(n)$ runtime

$O(k \log^2 k \log n)$

Non-uniform bounds (for each):

Goldreich-Levin'89
Kushilevitz-Mansour'91, Mansour'92
Gilbert-Guha-Indyk-Muthukrishnan-Strauss'02
Gilbert-Muthukrishnan-Strauss'05
Hassanieh-Indyk-Katabi-Price'12a
Hassanieh-Indyk-Katabi-Price'12b
Indyk-K.-Price'14

Randomized, $O(k \cdot \text{poly}(\log n))$ runtime

$O(k \log n \cdot (\log\log n)^C)$

Lower bound: $\Omega(k \log(n/k))$ for non-adaptive algorithms Do-Ba-Indyk-Price-Woodruff'10

Uniform bounds (for all):

Candes-Tao'06
Rudelson-Vershynin'08
Cheraghchi-Guruswami-Velingker'12
Bourgain'14
Haviv-Regev'15

Deterministic, $\Omega(n)$ runtime

$O(k\log^2 k\log n)$

Non-uniform bounds (for each):

Goldreich-Levin'89
Kushilevitz-Mansour'91, Mansour'92
Gilbert-Guha-Indyk-Muthukrishnan-Strauss'02
Gilbert-Muthukrishnan-Strauss'05
Hassanieh-Indyk-Katabi-Price'12a
Hassanieh-Indyk-Katabi-Price'12b
Indyk-K.-Price'14

Randomized, $O(k\cdot \text{poly}(\log n))$ runtime

$O(k\log n\cdot(\log\log n)^C)$

Lower bound: $\Omega(k\log(n/k))$ for non-adaptive algorithms Do-Ba-Indyk-Price-Woodruff'10

Theorem

There exists an algorithm for ℓ_2/ℓ_2 sparse recovery from Fourier measurements using $O(k\log n)$ samples and $O(n\log^3 n)$ runtime.

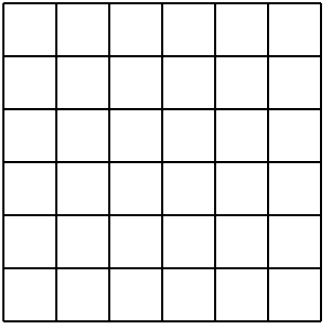
Optimal up to constant factors for $k \le n^{1-\delta}$.

Higher dimensional Fourier transform is needed in some applications

Given $x \in \mathbb{C}^{[n]}$, $N = n^d$, compute

$$\widehat{x}_j = \frac{1}{\sqrt{N}} \sum_{i \in [n]} \omega^{i^T j} x_i \quad \text{and} \quad x_j = \frac{1}{\sqrt{N}} \sum_{i \in [n]} \omega^{-i^T j} \widehat{x}_i$$

where ω is the n-th root of unity, and n is a power of 2.



Previous sample complexity bounds:

- $O(k \log^d N)$ in sublinear time algorithms
  - runtime $k \log^{O(d)} N$, for each
- $O(k \log^4 N)$ for any d
  - $\Omega(N)$ time, for all

This lecture:

Theorem

There exists an algorithm for ℓ_2/ℓ_2 sparse recovery from Fourier measurements using $O_d(k \log N)$ samples and $O(N \log^3 N)$ runtime.

Sample-optimal up to constant factors for any constant d.

What about sublinear time recovery?

Theorem

There exists an algorithm for ℓ_2/ℓ_2 *sparse recovery from Fourier measurements using* $O_d(k \log N (\log \log N)^2)$ *samples and* $O(k \log^{d+2} N)$ *runtime.*

This extends the result of Indyk-K.-Price'14 to higher dimensions

1. $O(k \log n)$ sample complexity in $O(n \log^3 n)$ time
   - extends to higher dimensions d
2. $O(k \log N (\log \log N)^2)$ sample complexity in $O(k \log^{d+2} N)$ time

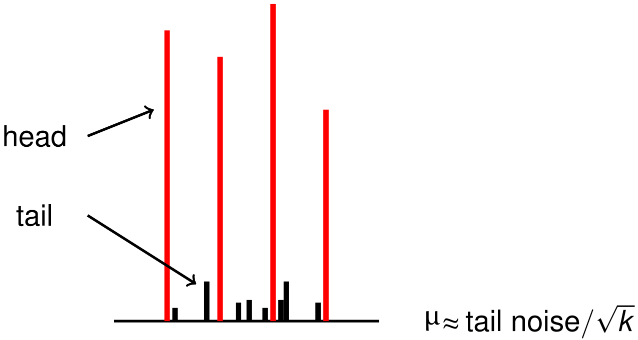
1. $O(k \log n)$ sample complexity in $O(n \log^3 n)$ time
   - extends to higher dimensions d
2. $O(k \log N (\log\log N)^2)$ sample complexity in $O(k \log^{d+2} N)$ time

Outline:

1. **ℓ_2/ℓ_2 sparse recovery guarantee**
2. Iterative recovery scheme
3. Sample-optimal algorithm in $O(N \log^3 N)$ time for $d = 1$
4. Experiments

ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\widehat{x}-\widehat{y}||^2 \leq C \cdot \min_{k-\text{sparse } \widehat{z}} ||\widehat{x}-\widehat{z}||^2$$



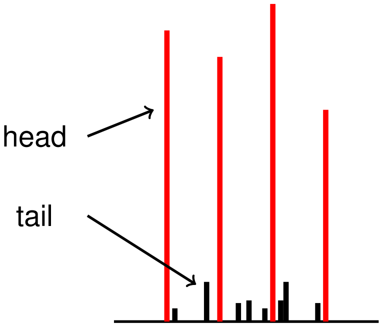
ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\hat{x} - \hat{y}||^2 \leq C \cdot \min_{k-\text{sparse } \hat{z}} ||\hat{x} - \hat{z}||^2$$

$|\hat{x}_1| \geq \ldots \geq |\hat{x}_k| \geq$
$|\hat{x}_{k+1}| \geq |\hat{x}_{k+2}| \geq \ldots$

$\text{Err}_k^2(\hat{x}) = \sum_{j=k+1}^{n} |\hat{x}_j|^2$

Residual error bounded by noise energy $\text{Err}_k^2(\hat{x})$



$\mu \approx \text{tail noise}/\sqrt{k}$

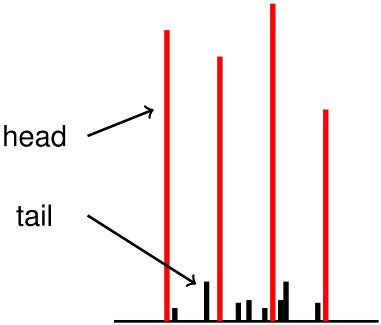
ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\widehat{x}-\widehat{y}||^2 \leq C \cdot \mathrm{Err}_k^2(\widehat{x})$$

$$|\widehat{x}_1| \geq \ldots \geq |\widehat{x}_k| \geq$$
$$|\widehat{x}_{k+1}| \geq |\widehat{x}_{k+2}| \geq \ldots$$

$$\mathrm{Err}_k^2(\widehat{x}) = \sum_{j=k+1}^{n} |\widehat{x}_j|^2$$

Residual error bounded by noise energy $\mathrm{Err}_k^2(\widehat{x})$



$\mu \approx$ tail noise$/\sqrt{k}$

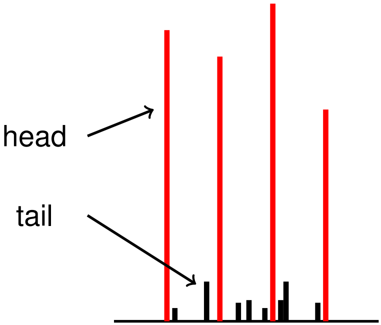
ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\widehat{x} - \widehat{y}||^2 \leq C \cdot \mathrm{Err}_k^2(\widehat{x})$$

$$|\widehat{x}_1| \geq \ldots \geq |\widehat{x}_k| \geq$$
$$|\widehat{x}_{k+1}| \geq |\widehat{x}_{k+2}| \geq \ldots$$

$$\mathrm{Err}_k^2(\widehat{x}) = \sum_{j=k+1}^{n} |\widehat{x}_j|^2$$

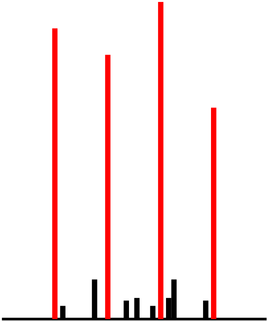
Residual error bounded by noise energy $\mathrm{Err}_k^2(\widehat{x})$



$\mu \approx$ tail noise$/\sqrt{k}$

ℓ_2/ℓ_2 sparse recovery guarantees:

$$\|\widehat{x} - \widehat{y}\|^2 \le C \cdot \mathrm{Err}_k^2(\widehat{x})$$



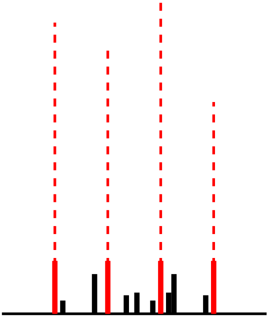
$\mu \approx$ tail noise$/\sqrt{k}$

Sufficient to ensure that most elements are below average noise level:

$$|\widehat{x}_i - \widehat{y}_i|^2 \le c \cdot \mathrm{Err}_k^2(\widehat{x})/k =: \mu^2$$

ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\widehat{x} - \widehat{y}||^2 \leq C \cdot \text{Err}_k^2(\widehat{x})$$



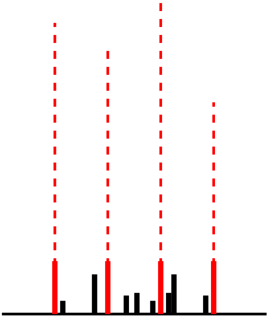
$\mu \approx \text{tail noise}/\sqrt{k}$

Sufficient to ensure that most elements are below average noise level:

$$|\widehat{x}_i - \widehat{y}_i|^2 \leq c \cdot \text{Err}_k^2(\widehat{x})/k = c \cdot \mu^2$$

ℓ_2/ℓ_2 sparse recovery guarantees:

$$||\widehat{x} - \widehat{y}||^2 \leq C \cdot \text{Err}_k^2(\widehat{x})$$



$\mu \approx \text{tail noise}/\sqrt{k}$

Sufficient to ensure that most elements are below average noise level:

$$|\widehat{x}_i - \widehat{y}_i| \leq c\mu$$

Outline:

1. ℓ_2/ℓ_2 sparse recovery guarantee
2. **Iterative recovery scheme**
3. Sample-optimal algorithm in $O(N \log^3 N)$ time for $d = 1$
4. Experiments

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\quad\triangleright$Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

PartialRecovery(x)
return dominant Fourier coefficients $\widehat{z}$ of x (approximately)

dominant coefficients$\approx |\widehat{x}_i| \geq c\mu$(above average noise level)

PARTIALRECOVERY(x)

return dominant Fourier coefficients $\widehat{z}$ of x (approximately)

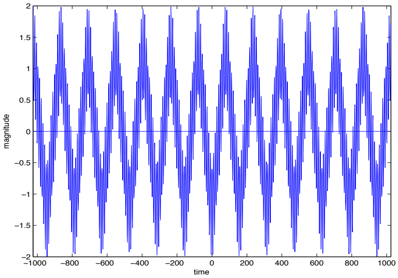
dominant coefficients $\approx |\widehat{x}_i| \geq c\mu$ (above average noise level)

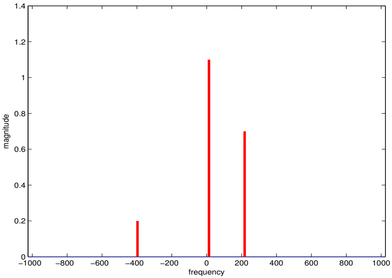
PARTIALRECOVERY(x)

return dominant Fourier coefficients $\widehat{z}$ of x (approximately)

dominant coefficients $\approx |\widehat{x}_i| \geq c\mu$ (above average noise level)

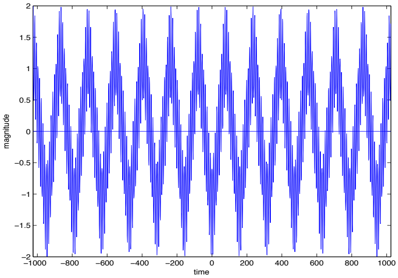
Recap of techniques from previous lectures

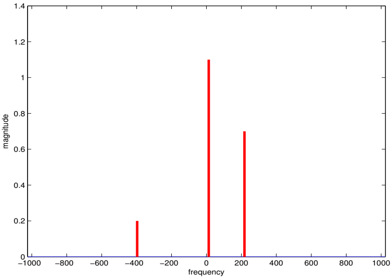

magnitude
time


magnitude
frequency

Task:approximate top k coeffs of $\hat{x}$ using few samples

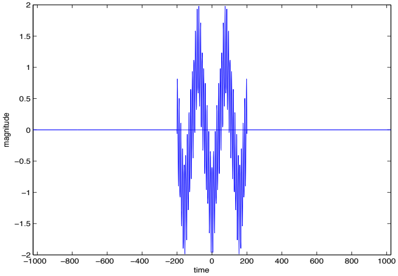
Natural idea: look at the value of the signal on the first $O(k)$ points

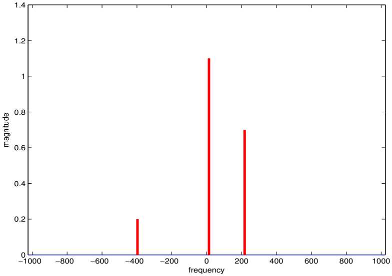




Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

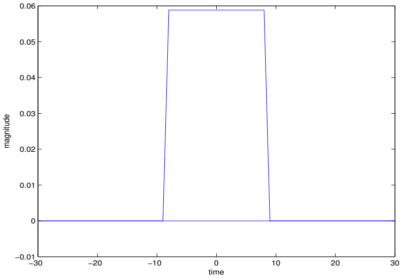


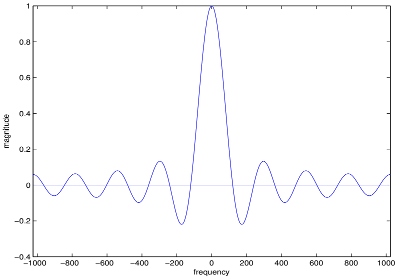


Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$

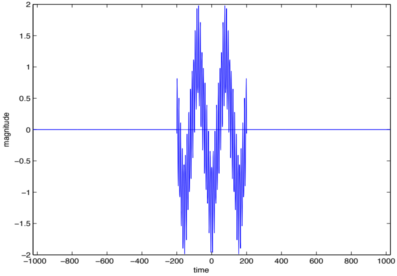


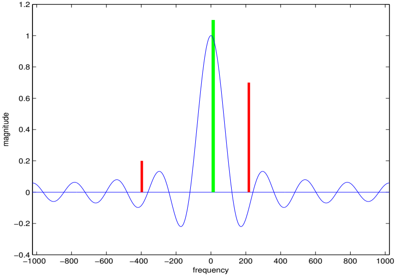


Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



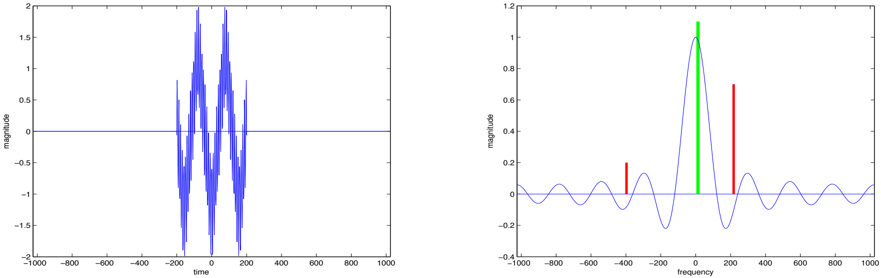


$$\widehat{(G \cdot x)}_f = \sum_{f' \in [n]} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$

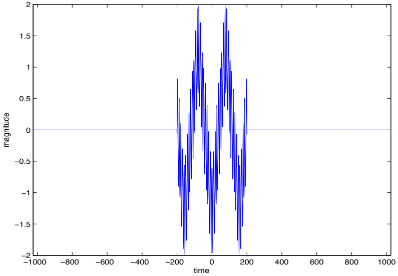


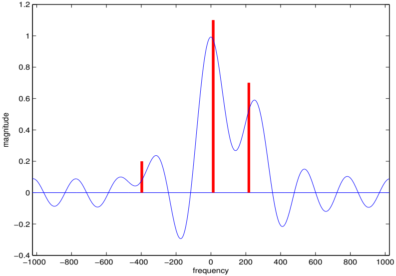
$$\widehat{(G \cdot x)}_f = \widehat{x}_f + \sum_{f' \in [n], f' \neq f} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



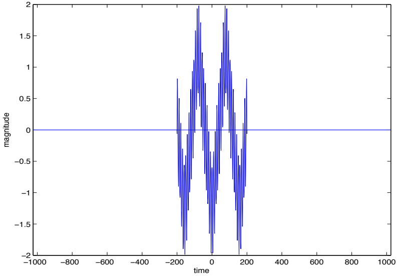


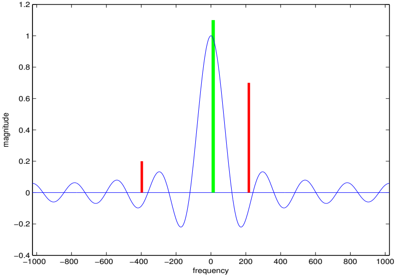
$$\widehat{(G \cdot x)}_f = \widehat{x}_f + \sum_{f' \in [n], f' \neq f} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



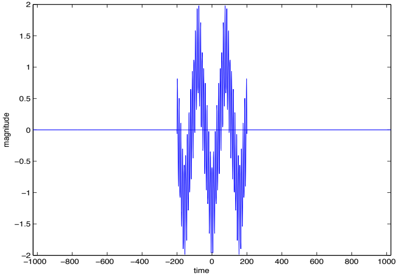


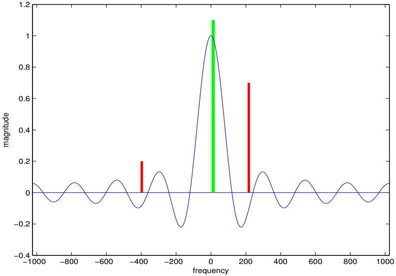
$$\widehat{(G \cdot x)}_f = \widehat{x}_f + \sum_{f' \in [n], f' \neq f} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\hat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \hat{x} * \widehat{G}$



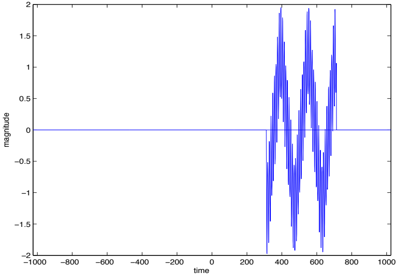


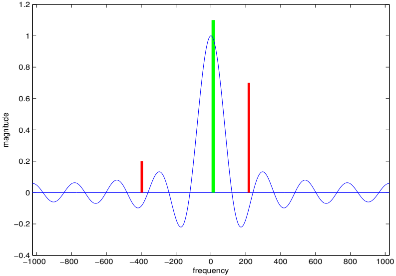
$$\widehat{(G \cdot x)}_f \omega^{-af} = \hat{x}_f + \sum_{f' \in [n] \setminus \{f\}} \omega^{a(f'-f)} \hat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



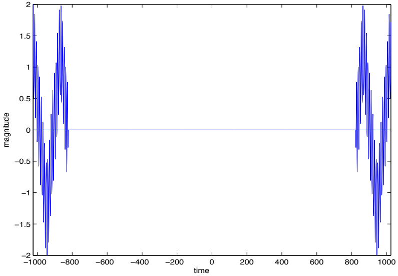


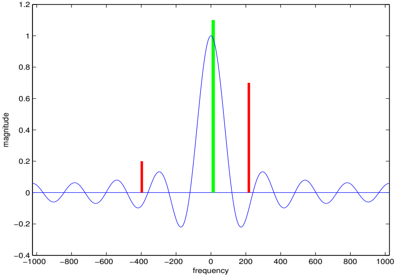
$$\widehat{(G \cdot x)}_f \omega^{-af} = \widehat{x}_f + \sum_{f' \in [n] \setminus \{f\}} \omega^{a(f'-f)} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



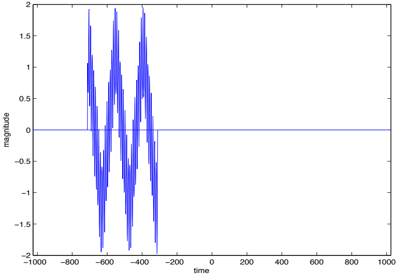


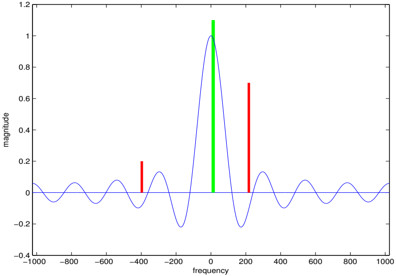
$$\widehat{(G \cdot x)}_f \omega^{-af} = \widehat{x}_f + \sum_{f' \in [n] \setminus \{f\}} \omega^{a(f'-f)} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



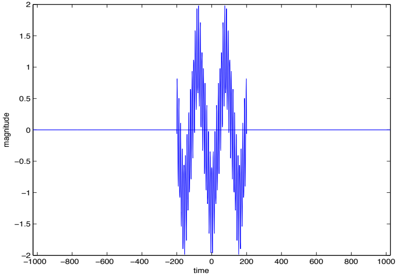


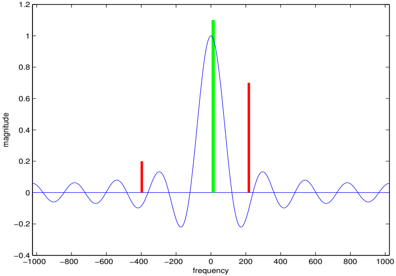
$$\widehat{(G \cdot x)}_f \omega^{-af} = \widehat{x}_f + \sum_{f' \in [n] \setminus \{f\}} \omega^{a(f'-f)} \widehat{x}_{f'} \widehat{G}_{f-f'}$$

Task:approximate top k coeffs of $\widehat{x}$ using few samples

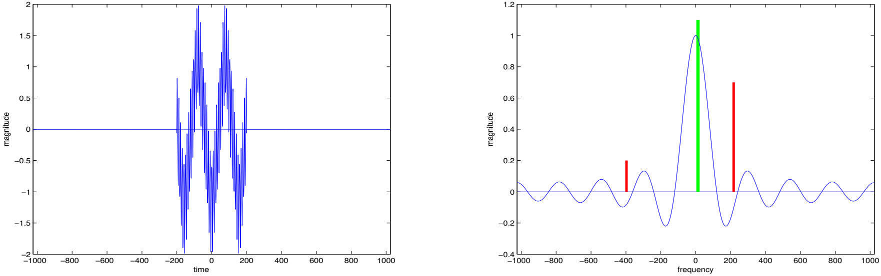
Natural idea: look at the value of the signal on the first $O(k)$ points

This convolves spectrum with sinc: $\widehat{(x \cdot G)} = \widehat{x} * \widehat{G}$



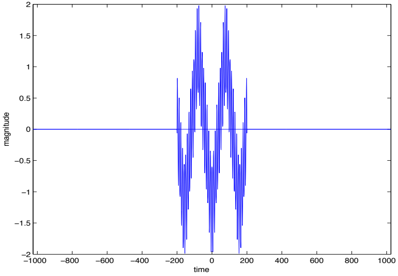


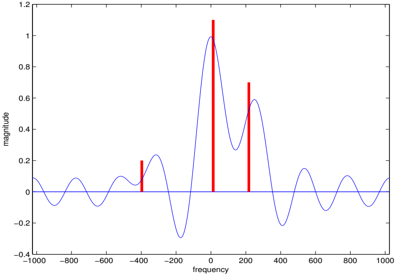
$$\widehat{(G \cdot x)}_f \omega^{-af} = \widehat{x}_f + \sum_{f' \in [n] \setminus \{f\}} \omega^{a(f'-f)} \widehat{x}_{f'} \widehat{G}_{f-f'}$$



$$\mathbf{E}_a\left[|\widehat{(G\cdot x)}_f\omega^{-af} - \hat{x}_f|^2\right] = \sum_{f'\in[n]\setminus\{f\}} |\hat{x}_{f'}|^2|\widehat{G}_{f-f'}|^2$$

- Expected error in terms of ℓ_2 norm (Parseval's indentity).
- Take median of independent trials

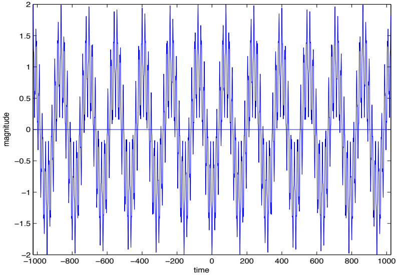


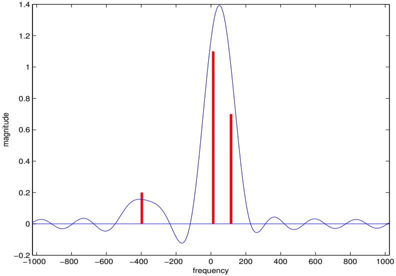


$$\mathbf{E}_a\left[|\widehat{(G\cdot x)}_f\omega^{-af} - \widehat{x}_f|^2\right] = \sum_{f'\in[n]\setminus\{f\}} |\widehat{x}_{f'}|^2|\widehat{G}_{f-f'}|^2$$

- Expected error in terms of ℓ_2 norm (Parseval's indentity).
- Take median of independent trials

What if two frequencies are close?





$$\mathbf{E}_a\left[|\widehat{(G\cdot x)}_f\omega^{-af} - \widehat{x}_f|^2\right] = \sum_{f'\in[n]\setminus\{f\}} |\widehat{x}_{f'}|^2|\widehat{G}_{f-f'}|^2$$

- Expected error in terms of ℓ_2 norm (Parseval's indentity).
- Take median of independent trials

Pseudorandom permutation

Gilbert-Muthukrishnan-Strauss'05:

Do a random invertible linear transformation of time domain:

$$(P_{\sigma,a,q}x)_i = x_{\sigma(i-a)}\omega^{\sigma q i}$$

This operation permutes the spectrum:

$$(\widehat{P_{\sigma,a,q}x})_{\pi_{\sigma,q}(i)} = \widehat{x}_i \omega^{a\sigma i},$$

where

$$\pi_{\sigma,a}(i) = \sigma(i-a) \mod n.$$

PARTIALRECOVERY(x)

return dominant Fourier coefficients $\widehat{z}$ of x (approximately)

Take $M = C \log n$ independent measurements:

$$y^j \leftarrow (P_{\sigma_j, a_j, q_j} x) \cdot G$$

Sample complexity= filter support $\times$ log n

Estimate each $f \in [n]$ as

$$\widehat{w}_f \leftarrow \text{median}\left\{\widehat{y}^j_{\pi_1(f)} \omega^{-a_1 \sigma_1 f}, \ldots, \widehat{y}^j_{\pi_M(f)} \omega^{-a_M \sigma_M f}\right\}$$
$$=: \text{median}\left\{\widetilde{y}^1_f, \ldots, \widetilde{y}^M_f\right\}.$$

Claim

If G =boxcar filter with support k/α, then with probability at least $1 - n^{-\Omega(C)}$

$$|\widehat{x}_f - \widehat{w}_f|^2 = O(\alpha) \cdot ||\widehat{x}||_2^2 / k$$

PARTIALRECOVERY(x)

return dominant Fourier coefficients $\widehat{z}$ of x (approximately)

Take $M = C \log n$ independent measurements:

$$y^j \leftarrow (P_{\sigma_j, a_j, q_j} x) \cdot G$$

Sample complexity= filter support $\times \log n$

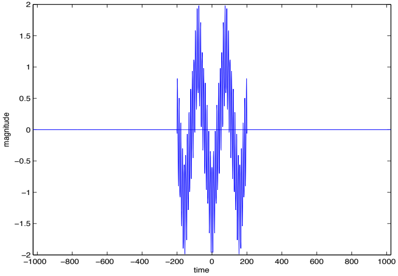
Estimate each $f \in [n]$ as

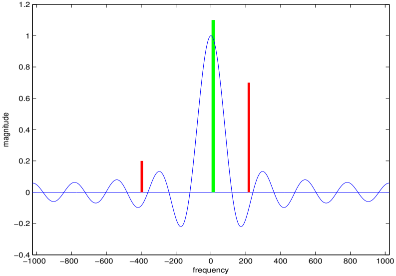
$$\widehat{w}_f \leftarrow \text{median}\left\{ \widehat{y}^j_{\pi_1(f)} \omega^{-a_1 \sigma_1 f}, \ldots, \widehat{y}^j_{\pi_M(f)} \omega^{-a_M \sigma_M f} \right\}$$
$$=: \text{median}\left\{ \widetilde{y}^1_f, \ldots, \widetilde{y}^M_f \right\}.$$

Claim

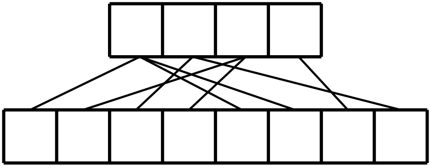
If G =boxcar filter with support k/α, then with probability at least $1 - n^{-\Omega(C)}$

$$|\widehat{x}_f - \widehat{w}_f|^2 = O(\alpha) \cdot ||\widehat{x}||_2^2 / k \gg \mu^2$$

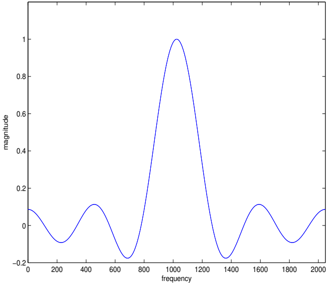


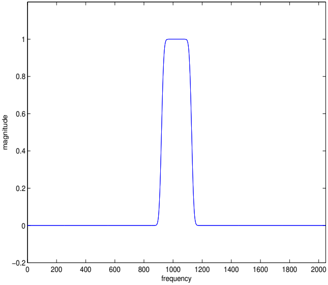


Like hashing heavy hitters into buckets (COUNTSKETCH, COUNTMIN), but buckets leak



Most work so far: make PartialRecovery step more efficient (better filters!)





Increases filter support to $k \log n$...

Outline:

1. ℓ_2/ℓ_2 sparse recovery guarantee
2. Iterative recovery scheme
3. **Sample-optimal algorithm in $O(N \log^3 N)$ time for $d = 1$**
4. Experiments

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$ Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$ Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

In most prior works sampling complexity is

samples per PartialRecovery step $\times$ number of iterations

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

In most prior works sampling complexity is

samples per PartialRecovery step $\times$ number of iterations

Lots of work on carefully choosing filters, reducing number of iterations:
Hassanieh-Indyk-Katabi-Price'12,
Ghazi-Hassanieh-Indyk-Katabi-Price-Shi'13, Indyk-K.-Price'14

- still lose $\Omega(\log\log n)$ in sample complexity (number of iterations)
- lose $\Omega((\log n)^{d-1} \log\log n)$ in higher dimensions

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$ Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

Our sampling complexity is

samples per PartialRecovery step × number of iterations

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$ ~~Takes random samples~~ of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

Our sampling complexity is

samples per PARTIALRECOVERY step ~~$\times$ number of iterations~~

Iterative recovery

Input: $x \in \mathbb{C}^n$
$\widehat{y}_0 \leftarrow 0$
For $t = 1$ to L

- $\widehat{z} \leftarrow \textsc{PartialRecovery}(x - y_{t-1})$ $\triangleright$ Takes random samples of $x - y$
- Update $\widehat{y}_t \leftarrow \widehat{y}_{t-1} + \widehat{z}$

Our sampling complexity is

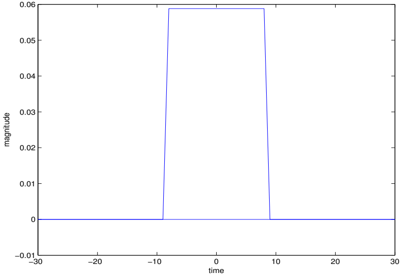
samples per PartialRecovery step ~~× number of iterations~~

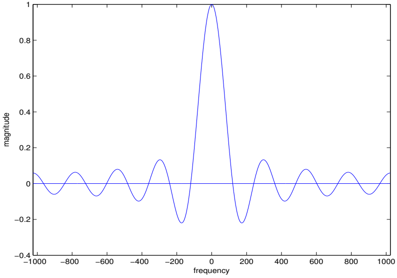
Can use very simple filters!

Our filter=boxcar convolved with itself $O(1)$ times

Filter support is $O(k)$ (=samples per measurement)

$O(k \log n)$ samples in PARTIALRECOVERY step



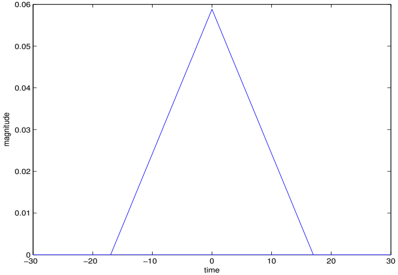


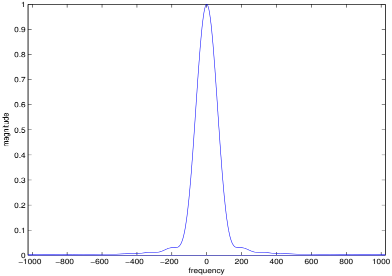
Can choose a rather weak filter, but do not need fresh randomness

Our filter=boxcar convolved with itself $O(1)$ times

Filter support is $O(k)$ (=samples per measurement)

$O(k \log n)$ samples in PARTIALRECOVERY step



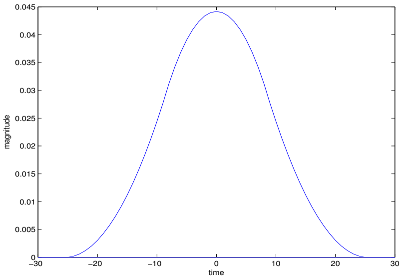


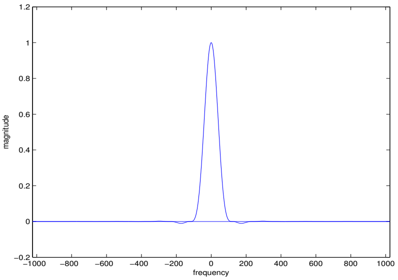
Can choose a rather weak filter, but do not need fresh randomness

Our filter=boxcar convolved with itself $O(1)$ times

Filter support is $O(k)$ (=samples per measurement)

$O(k \log n)$ samples in PARTIALRECOVERY step





Can choose a rather weak filter, but do not need fresh randomness

$G \leftarrow B * B * B$ ▷ Take samples of x

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$: ▷ Loop over thresholds

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$ ▷ Estimate, prune small elements

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$ ▷ Update samples

for $m = 1, \ldots, M$

End

$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

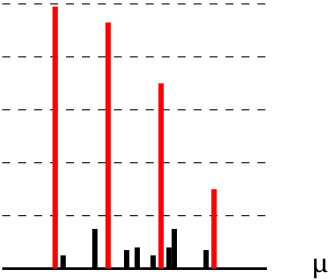
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

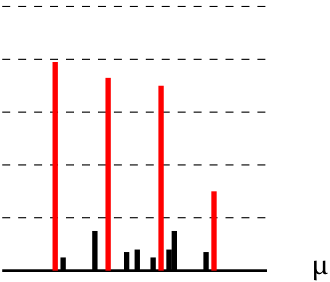
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

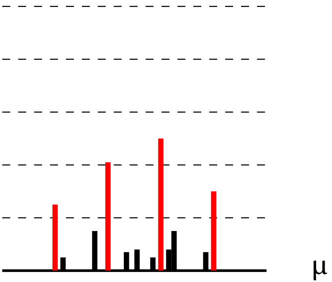
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

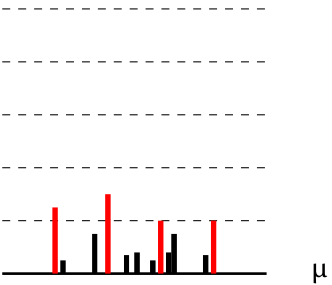
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

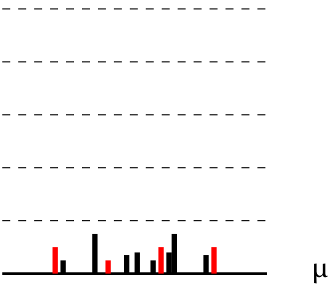
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



$G \leftarrow B * B * B$

Let $y^m \leftarrow (P_m x) \cdot G$

$m = 0, \ldots, M = C \log n$

$\widehat{z}_0 \leftarrow 0$

For $t = 1, \ldots, T = O(\log n)$:

For $f \in [n]$:

$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$

If $|\widehat{w}_f| < 2^{T-t}\mu/3$ **then**

$\widehat{w}_f \leftarrow 0$

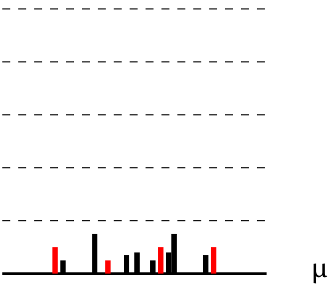
End

$\widehat{z}_{t+1} = \widehat{z}_t + \widehat{w}$

$y^m \leftarrow y^m - (P_m w) \cdot G$

for $m = 1, \ldots, M$

End



Main challenge: lack of fresh randomness. Why does median work?

Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$$

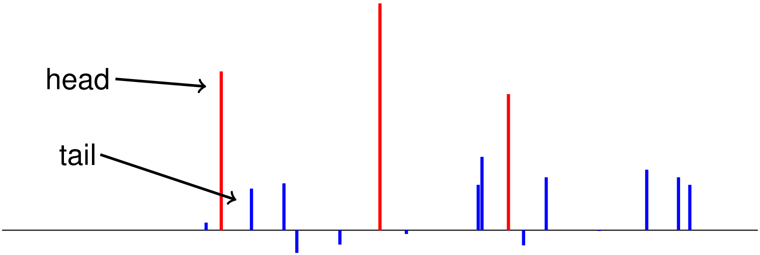
Main idea of analysis: split estimation error into two parts:

$$\left|\widetilde{y}_f - \widehat{x}_f\right| = \text{noise from head elements} + \text{tail noise}$$

Let S denote the set of heavy hitters:

$$S = \{ i \in [n] : |\widehat{x}_i| > \mu \} .$$

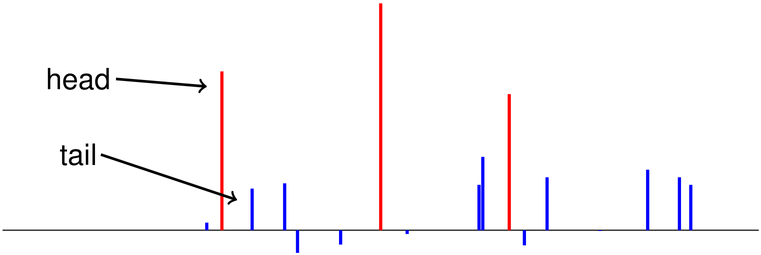
There cannot be too many of them: $|S| = O(k)$



Let S denote the set of heavy hitters:

$$S = \{i \in [n] : |\widehat{x}_i| > \mu\}.$$

There cannot be too many of them: $|S| = O(k)$

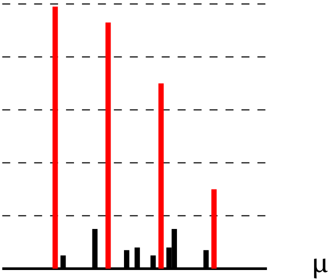


Main invariant: never modify $\widehat{x}$ outside of S

Intuition: we only modify large frequencies (say those larger than 4μ), and only those that we have reliable estimates for

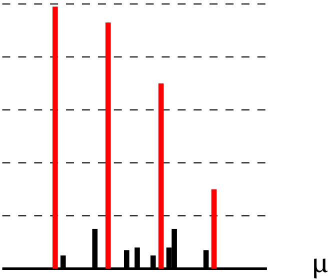
At time t:

- get $1 \pm 1/3$ approximation to near-maximum coordinates
- $||\widehat{x}||_\infty$ decreases at least by factor of 2
- only update elements in S



At time t:

- get $1 \pm 1/3$ approximation to near-maximum coordinates
- $||\widehat{x}||_\infty$ decreases at least by factor of 2
- only update elements in S

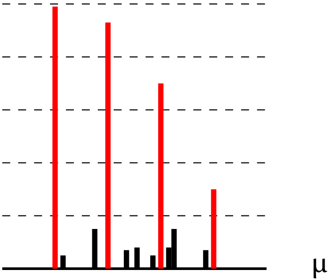


Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widehat{y}_f^M\right\}$$

At time t:

- get $1 \pm 1/3$ approximation to near-maximum coordinates
- $||\widehat{x}||_\infty$ decreases at least by factor of 2
- only update elements in S



Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$

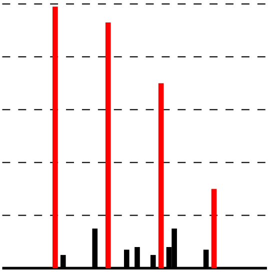
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widehat{y}_f^M\right\}$$

Need to show that estimation error is small:

$$\left|\widetilde{y}_f - \widehat{x}_f\right| = \text{noise from head elements} + \text{tail noise}$$

At time t:

- get $1 \pm 1/3$ approximation to near-maximum coordinates
- $||\widehat{x}||_\infty$ decreases at least by factor of 2
- only update elements in S



μ

Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$

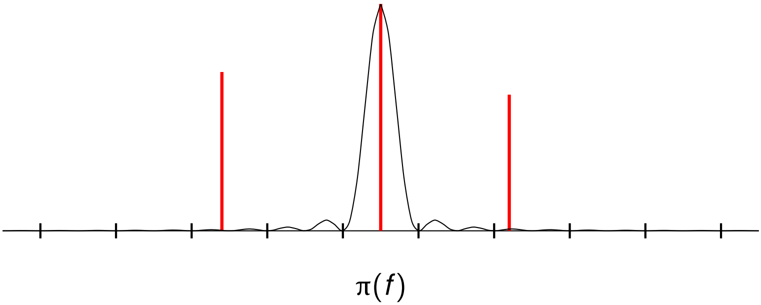
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widehat{y}_f^M\right\}$$

Need to show that estimation error is small:

$$\left|\widetilde{y}_f - \widehat{x}_f\right| = \text{noise from head elements} + O(\mu)$$

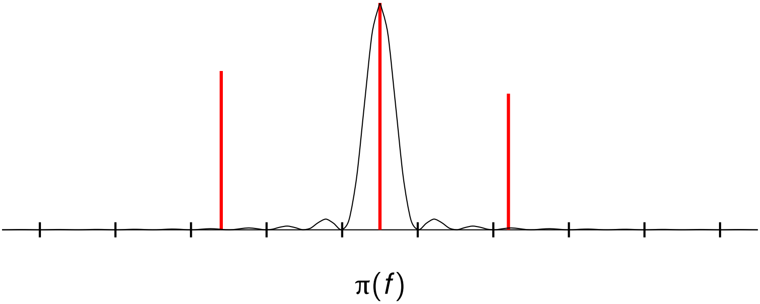
Need to show that estimation error is small:

$$\left|\tilde{y}_f - \hat{x}_f\right| \leq \sum_{f' \in S \setminus \{f\}} \omega^{a\sigma(f'-f)} \hat{x}_{f'} \hat{G}_{\pi(f)-\pi(f')} + O(\mu)$$



Need to show that estimation error is small:

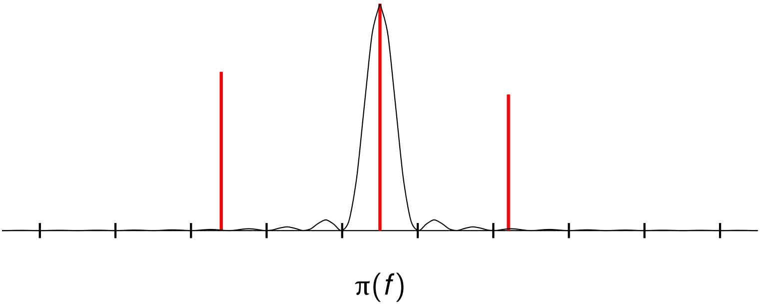
$$\left|\widetilde{y}_f - \widehat{x}_f\right| \leq \sum_{f' \in S \setminus \{f\}} \omega^{a\sigma(f'-f)} \widehat{x}_{f'} \widehat{G}_{\pi(f)-\pi(f')} + O(\mu)$$



- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)

Need to show that estimation error is small:

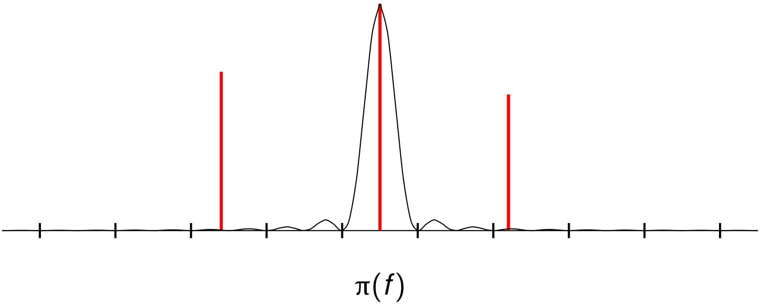
$$\left|\widetilde{y}_f - \widehat{x}_f\right| \le \sum_{f' \in S \setminus \{f\}} |\widehat{x}_{f'}||\widehat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)

Need to show that estimation error is small:

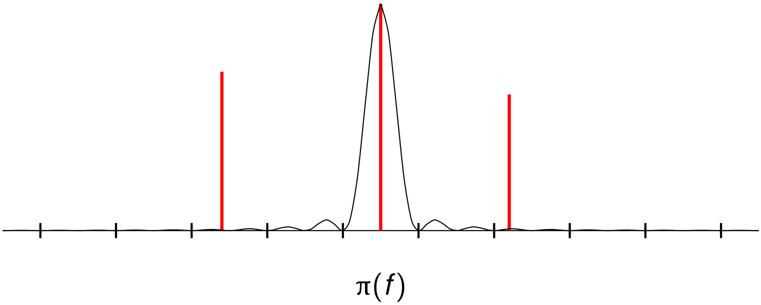
$$\left|\widetilde{y}_f - \widehat{x}_f\right| \leq \|\widehat{x}_S\|_\infty \cdot \sum_{f' \in S\setminus\{f\}} |\widehat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)
- Peel off largest elements only, so ok with error bounds like $\|\widehat{x}\|_\infty/100$

Need to show that estimation error is small:

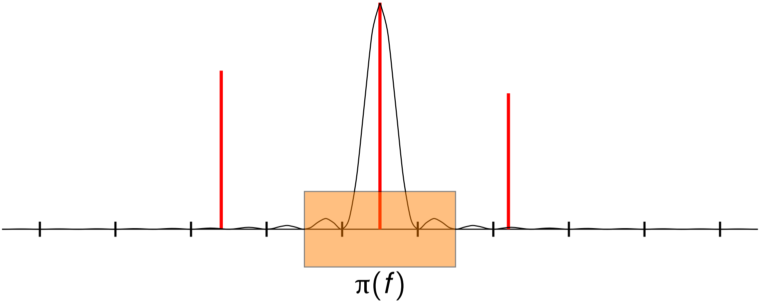
$$\left|\tilde{y}_f - \hat{x}_f\right| \leq ||\hat{x}_S||_\infty \cdot \sum_{f' \in S \setminus \{f\}} |\hat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)
- Peel off largest elements only, so ok with error bounds like $||\hat{x}||_\infty/100$
- If other head elements are far from $\pi(f)$, estimation error is small!

Need to show that estimation error is small:

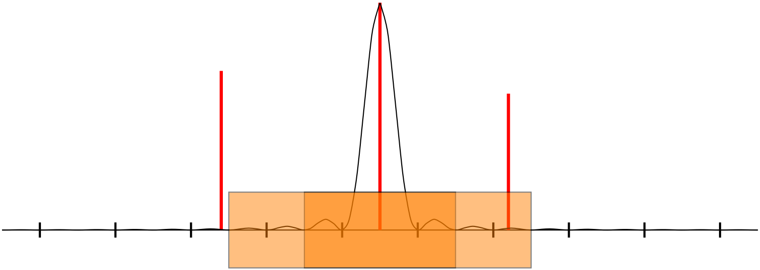
$$\left|\widetilde{y}_f - \widehat{x}_f\right| \leq ||\widehat{x}_S||_\infty \cdot \sum_{f' \in S \setminus \{f\}} |\widehat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)
- Peel off largest elements only, so ok with error bounds like $||\widehat{x}||_\infty/100$
- If other head elements are far from $\pi(f)$, estimation error is small!

Need to show that estimation error is small:

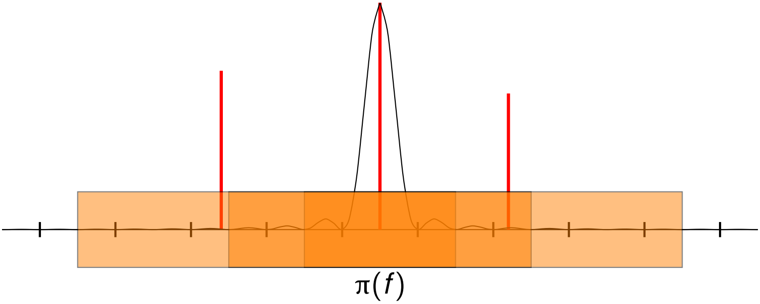
$$\left|\widetilde{y}_f - \widehat{x}_f\right| \le ||\widehat{x}_S||_\infty \cdot \sum_{f' \in S \setminus \{f\}} |\widehat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



- Cannot assume that a is random, but that is ok here! (use ℓ_1 bounds)
- Peel off largest elements only, so ok with error bounds like $||\widehat{x}||_\infty/100$
- If other head elements are far from $\pi(f)$, estimation error is small!

Need to show that estimation error is small:

$$\left|\widetilde{y}_f - \widehat{x}_f\right| \le ||\widehat{x}_S||_\infty \cdot \sum_{f' \in S \setminus \{f\}} |\widehat{G}_{\pi(f)-\pi(f')}| + O(\mu)$$



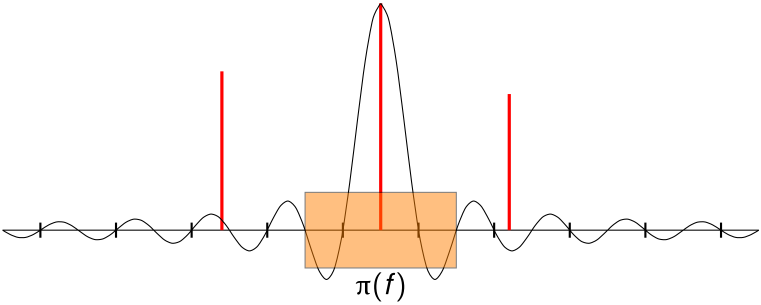
- Cannot assume that *a* is random, but that is ok here! (use ℓ_1 bounds)
- Peel off largest elements only, so ok with error bounds like $||\widehat{x}||_\infty/100$
- If other head elements are far from $\pi(f)$, estimation error is small!

Definition (Isolated at scale t)

Suppose that filter support is k/α for some constant $\alpha < 1$. A frequency $f \in [n]$ is isolated under π *at scale t* if

$$\pi(f) + [-(n/b) \cdot 2^t, (n/b) \cdot 2^t]$$

contains at most $O(\sqrt{\alpha}) \cdot 2^{(3/2)t}$ elements of $\pi(S)$.

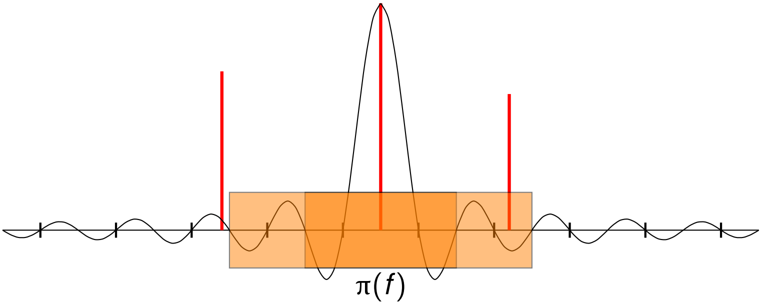


Definition (Isolated at scale t)

Suppose that filter support is k/α for some constant $\alpha < 1$. A frequency $f \in [n]$ is isolated under π *at scale t* if

$$\pi(f) + [-(n/b)\cdot 2^t, (n/b)\cdot 2^t]$$

contains at most $O(\sqrt{\alpha})\cdot 2^{(3/2)t}$ elements of $\pi(S)$.

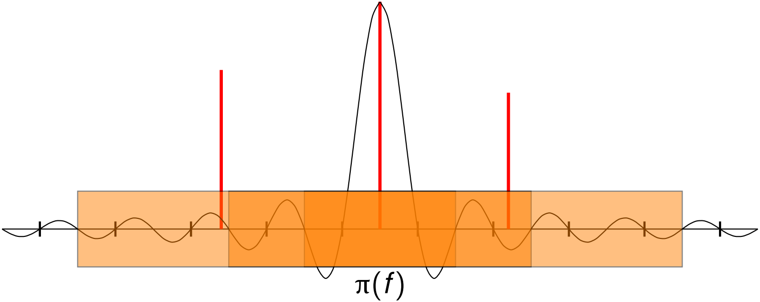


Definition (Isolated at scale t)

Suppose that filter support is k/α for some constant $\alpha < 1$. A frequency $f \in [n]$ is isolated under π *at scale t* if

$$\pi(f) + [-(n/b)\cdot 2^t, (n/b)\cdot 2^t]$$

contains at most $O(\sqrt{\alpha})\cdot 2^{(3/2)t}$ elements of $\pi(S)$.

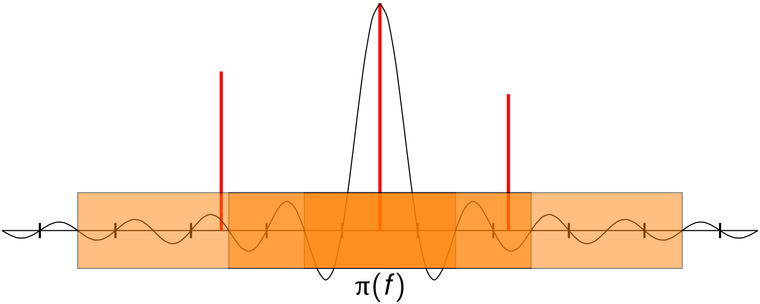


Definition (Isolated ~~at scale t~~)

Suppose that filter support is k/α for some constant $\alpha < 1$. A frequency $f \in [n]$ is isolated under π ~~*at scale t*~~ if

$$\pi(f) + [-(n/b)\cdot 2^t, (n/b)\cdot 2^t]$$

contains at most $O(\sqrt{\alpha})\cdot 2^{(3/2)t}$ elements of $\pi(S)$ for all $t \geq 0$.



Lemma

Any $i \in [n]$ *is isolated in* $2/3$ *fraction of measurements whp.*

Lemma

Any $i \in [n]$ is isolated in $2/3$ fraction of measurements whp.

Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$$

If f is isolated, then

$$||\widehat{x}||_\infty/100 + O(\mu)$$

so we have $1 \pm 1/3$ estimates for near-maximum elements, e.g.

$$|\widehat{x}_i| \geq ||\widehat{x}||_\infty/3$$

Lemma

Any $i \in [n]$ is isolated in $2/3$ fraction of measurements whp.

Main estimation step:

$$y^m \leftarrow (P_m x) \cdot G, m = 0, \ldots, M = C \log n$$
$$\widehat{w}_f \leftarrow \text{median}\left\{\widetilde{y}_f^1, \ldots, \widetilde{y}_f^M\right\}$$

If f is isolated, then

$$||\widehat{x}||_\infty / 100 + O(\mu)$$

so we have $1 \pm 1/3$ estimates for near-maximum elements, e.g.

$$|\widehat{x}_i| \geq ||\widehat{x}||_\infty / 3$$

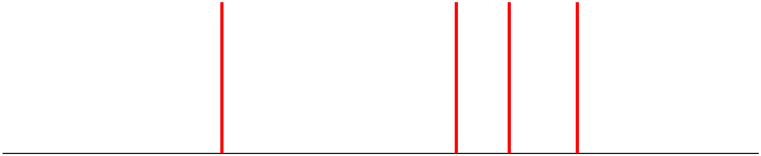
Proved that this works just like with fresh randomness!
(as long as we recover starting from largest frequencies)

Lecture so far

- Optimal sample complexity by reusing randomness
- Very simple algorithm, can be implemented
- Extension to higher dimensions: algorithm is the same, permutations are different.
  - Choose random invertible linear transformation over $\mathbb{Z}_n^d$

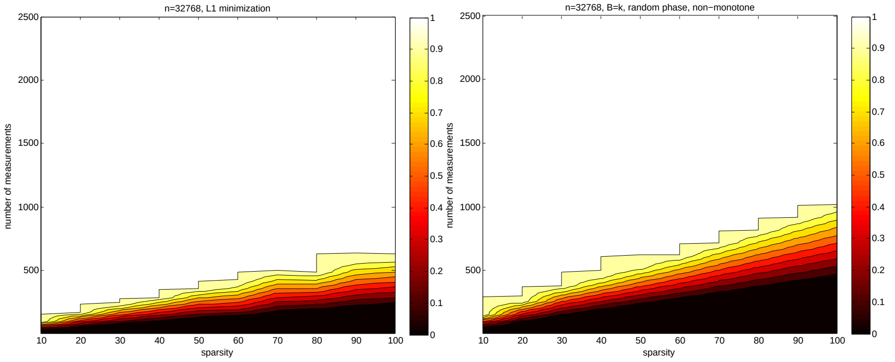
Experimental evaluation

Problem: recover support of a random k-sparse signal from Fourier measurements.
Parameters: $n = 2^{15}$, $k = 10, 20, \ldots, 100$
Filter: boxcar filter with support $k+1$



Comparison to ℓ_1-minimization (SPGL1)

$O(k \log^3 k \log n)$ sample complexity, requires LP solve



Within a factor of 2 of ℓ_1 minimization

Open questions:

- $O(k \log n)$ in $O(k \log^2 n)$ time?
- $O(k \log n)$ runtime?
- remove dependence on dimension? Current approaches lose C^d in sample complexity, $(\log n)^d$ in runtime

Open questions:

- $O(k \log n)$ in $O(k \log^2 n)$ time?
- $O(k \log n)$ runtime?
- remove dependence on dimension? Current approaches lose C^d in sample complexity, $(\log n)^d$ in runtime

More on sparse FFT:
`http://groups.csail.mit.edu/netmit/sFFT/index.html`