COMS E6998-9: Algorithms for Massive Data (Fall'23) Oct 16, 2023

Lecture 12: Sparse Fourier Transform

Instructor: *Alex Andoni* Scribes: *Manuel Paez*

1 Overview

Given access to signal $a \in \mathbb{R}^n$, let $\hat{a} = Fa$ be a Fourier transform and assume $\hat{a} + k$ sparse
Goal: Recover $\hat{a}'$ such that, for $Err_2^k(\hat{a}) = \underset{\hat{a}\text{ (k-sparse)}}{\operatorname{argmin}} \|\hat{a} - \hat{a}''\|$:

$$\|\hat{a}' - \hat{a}\| \leq C \cdot Err_2^k(\hat{a}) \tag{1}$$

for some constant C
Note: W/O assumption on $\hat{a}$, FFT has $O(nlogn)$ runtime
Goal: With assumption on $\hat{a}$, solve in $O(k(logn)^{O(1)})$ runtime

1.1 Case 1: $K = 1$ and no noise $\left(Err_2^1(\hat{a}) = 0\right)$

1.2 Case 2: $K = 2$ and noise $\left(Err_2^1(\hat{a}) \neq 0\right)$

Reminder: Algorithm for Case 1:
$\hat{a} = \hat{a}_u e_u$
$a_1/a_0 = \omega^u$ where $u = ...(a/a_0)$
$a_u = a_0$
Facts from before: $\hat{a}_u = \frac{1}{n}\sum a_j \omega_n^{uj}$ and $\omega_n \leq e^{-2\pi i}$
$\hat{a} = Fa$ and $F_{uj} = \frac{1}{n}\omega_n^{-uj}$
$a = F^{-1}\hat{a}$ and $F_{ju}^{-1} = \frac{1}{n}\omega_n^{uj}$
$\|\hat{a}\|_2^2 = \frac{1}{n} \cdot \|a\|_2^2$
With these facts, we have
$\hat{a} = \hat{a}_u e_u + \sum_{v \neq u} \hat{a}_v e_v$
$E = Err_2^1(\hat{a}) = \sum_{v \neq u} \hat{a}_v^2$
For C large enough, might as well assume $E < \epsilon \cdot \hat{a}_u^2$ for some $\epsilon > 0$.
Otherwise, a solution is to take $a = 0$, then

$$\|0 - \hat{a}\| = \hat{a}_u^2 + E \leq \frac{E}{\epsilon} + E = \frac{2E}{\epsilon}$$

$$C = 2/\epsilon$$

$\|F^{-1}\hat{a}_u\|_2^2 = n \cdot \hat{a}_u^2 \Rightarrow$ distributed almost equally on $[n]$
$\|F^{-1}\sum_{v \neq u} \hat{a}_v\|_2^2 = n \cdot E \Rightarrow$ it can be focused against on a_0/a_1

2 Algorithm for Case 2

We will recover u (the index in Fourier) bit by bit
$u = (b_{logn}, b_{logn-1}, ..., b_1, b_0)$ in binary

2.1 b_0

$b_0 : u = 2v + b_0$
$a_0 : \hat{a}_u \cdot \omega^0 = \hat{a}_u$
$a_{n/2} = \hat{a}_u \cdot \omega_(2)^{(2v+b_0)\cdot\frac{n}{2}} = \hat{a}_u \cdot \omega_n^{vn} \cdot \omega_n^{b_)\cdot\frac{n}{2}} = (-1)^{b_0} \cdot \hat{a}_u$

$$b_0 = 0 \Leftrightarrow |a_0 - a_{n/2}| < |a_0 + a_{n/2}| \tag{2}$$

Now, Let's look at a_r vs $a_{r+n/2}$ $a_r = \hat{a}_u \cdot \omega^{ur}$
$a_{r+n/2} = \hat{a}_u \cdot \omega^{u(r+n/2)} = \hat{a}_u \cdot \omega^{ur} \cdot \omega^{(2v+b_2)\cdot\frac{n}{2}} = (-1)^{b_0} a_r$
Thus,

$$\forall r, b_0 = 0 \Leftrightarrow |a_r - a_{r+n/2}| < |a_r + a_{r+n/2}| \tag{3}$$

We denoted this previous line as T_r^0

2.1.1 b_1

For b_1 : consider shifted signal $a'_j = a_j\omega^j$
$\hat{a}'_u = \hat{a}'_{u-1}$ (time / phase shift)
if $b_0 = 1$, apply time / phase shift as it makes $b_0 = 0$
$u = 2b + 0 = 2(2\omega + b_1) + 0 = 4 \cdot \omega + 2b_1$
$a_r = \hat{a}_u \cdot \omega^{ur}$
$a_{r+n/4} = \hat{a}_u \cdot \omega^{u(r+n/4)} = \hat{a}_u \cdot \omega^{ur}\omega^{(4w+2b_1)\cdot n/4)} = a_r \cdot \omega^{b_i \cdot n/2} = (-1)^{b_1} \cdot a_r$
Thus,

$$b_1 = 0 \Leftrightarrow |a_r - a_{r+n/2}| < |a_r + a_{r+n/2}| \tag{4}$$

We denoted this previous line as T'_r
Thus for $T_r^l : |a_r - a_{r+n/2(l+1)}| < |a_r + a_{r+n/2(l+1)}|$

2.2 With noise

Suppose $\mu =$ noise with time
$a_j = \hat{a}_u\omega^{uj} + \mu_j$
$\|\mu\|_2^2 = \|a - F^{-1}\hat{a}_u e_u\|_2^2 = n \cdot \sum_{v\neq u} \hat{a}_v^2 = n \cdot E$
Intuition: On average, $\mu_r \approx \sqrt{E} < \sqrt{\epsilon} \cdot \hat{a}_u$
$\Rightarrow$: For random r, T_r should correct

Claim 1. *fix any $e \leq \{0, ..., logn\}$ then T_r^l for random r is correct with prob* $\geq 1 - O(\sqrt{\epsilon})$

Proof. T_r^l looks at two random entries r, r' and it is correct if

$$|u_r||u_{r'}| < \frac{1}{4}$$

$$Pr[|u_r| > 1/4\hat{a}_u] = Pr[|u_r|^2 > 1/16 \cdot \hat{a}_u^2]$$

By Markov's, we will have that

$$\leq \frac{E[|u_r|^2]}{\frac{1}{16} \cdot \hat{a}_u^2} = \frac{\frac{1}{n} \cdot n \cdot E}{\hat{a}_u} \cdot 16 \leq 16\epsilon$$

Thus, $Pr[|u_r|$ and $|u_{r'}| < \hat{a}_u/u] \leq 32\epsilon$ (via union bound) □

Therefore, for $\epsilon < 1/128, T^l_{r,r'}$ correct with prob $geq 3/4$. We need to recover every bit l with successful prob $\geq 1 - \frac{1}{10logn}$

To guarantee this successful prob: for fixed bit l,

To guarantee this: for fixed bit l

Take majority vote

By Chernoff bound, $Pr[$ Maj is wrong $] \leq e^{-O(z)}$

samples into $a : \log n \cdot z \cdot 2 = O(\log n \log \log n)$

Thus, we have shown Fourier transform for 1-sparse vector