On the Power of Adaptivity in Sparse Recovery

Piotr Indyk **Eric Price** David Woodruff

MIT MIT IBM

2011-08-03

Standard Sparse Recovery Framework

- Specify distribution on $m \times n$ matrices A (independent of x).

- Given linear sketch Ax, recover $\hat{x}$.
- Satisfying the recovery guarantee:

$$\|\hat{x} - x\|_2 \leq (1 + \epsilon) \min_{k\text{-sparse } x_k} \|x - x_k\|_2$$

  for constant C, with probability $2/3$.
- Solvable with $\Theta(\frac{1}{\epsilon} k \log \frac{n}{k})$ measurements [CRT06, GLPS10]

Adaptive Sparse Recovery Framework

- For $i = 1 \dots r$:
  - Choose matrix A_i based on previous observations (possibly randomized).
  - Observe $A_i x$.
  - Number of measurements m is total number of rows in all A_i.
  - Number of *rounds* is r.
- Given linear sketch Ax, recover $\hat{x}$.
- Satisfying the recovery guarantee:

$$\|\hat{x} - x\|_2 \le (1 + \epsilon) \min_{k\text{-sparse } x_k} \|x - x_k\|_2$$

  for constant C, with probability $2/3$.
- Solvable with $\Theta(k \log \frac{n}{k})$ measurements and $\epsilon = o(1)$ [HBCN09]

Adaptive Sparse Recovery Framework

- For $i = 1 \ldots r$:
  - Choose matrix A_i based on previous observations (possibly randomized).
  - Observe $A_i x$.
  - Number of measurements m is total number of rows in all A_i.
  - Number of *rounds* is r.
- Given linear sketch Ax, recover $\hat{x}$.
- Satisfying the recovery guarantee:

$$\|\hat{x} - x\|_2 \leq (1+\epsilon) \min_{k\text{-sparse } x_k} \|x - x_k\|_2$$

for constant C, with probability $2/3$.

- **Main Theorem:** Solvable with $\Theta(\frac{1}{\epsilon} k \log\log \frac{n}{k})$ measurements
  - $O(\log^* k \log\log n)$ rounds.
- **Also:** $O(\frac{1}{\epsilon} k \log k + k \log n)$ measurements, 2 rounds.

When does adaptivity make sense?

- Some compressive sensing architectures support it.
- Streaming algorithms:
  - Adaptivity corresponds to multiple passes.
  - Router finding most common connections: **No**.
  - Mapreduce finding most frequent URLs: **Yes**.
- Cameras:
  - Programmable pixels (align mirrors, LCD display): **Yes**.
  - Hardwired lens: **No**.
- Disease testing:
  - **Yes**, but multiple rounds can be costly.

Outline of approach

- **Goal:** k-sparse recovery with $O(k \log\log(n/k))$ adaptive measurements, $1+\epsilon$ approximation factor, $O(\log^* k \log\log n)$ rounds.

1. 1-sparse recovery with $O(\log\log n)$ adaptive measurements, $O(1)$ approximation factor.
2. Black box conversion to k-sparse recovery (a la [GLPS10]):
   - Subsample at rate ϵ/k and apply (1), $O(k/\epsilon)$ times.
   - Replace k by $k/2$, repeat.

1-sparse recovery

- **Goal:** 1-sparse recovery with $m = O(\log\log n)$, $C = O(1)$.
- Non-adaptive lower bound: why is this hard?
- Suppose x is random e_i plus Gaussian noise.
- $\|x - e_i\|_2 \approx 1/(3C)$.
- C-approximate recovery requires finding i (or the noise).
- Observe $\langle v, x \rangle = v_i + \|v\|_2 N(0, 1/(3C)^2)$.
  - Gaussian channel.
  - Shannon-Hartley theorem: information capacity

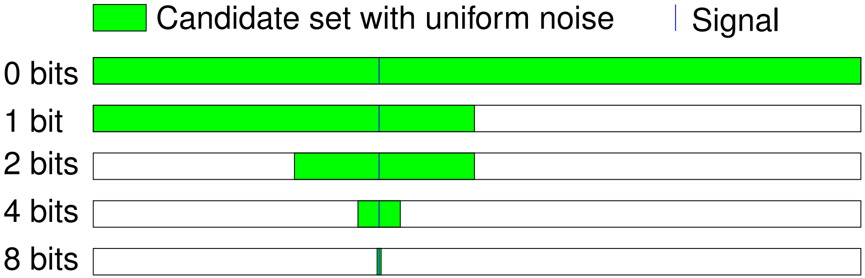
$$I(\langle v, x \rangle, i) \leq \frac{1}{2}\log(1 + \text{SNR}) = O(\log(1 + C)).$$

- Requires $\Omega(\frac{\log n}{\log 1+C})$ non-adaptive measurements.

1-sparse recovery

- What does that lower bound mean for adaptive recovery?
- Only get $\log(1 + \text{SNR})$ bits of information at each step.
- First observations only get $O(1)$ bits of information.
- Can we use those bits to improve the SNR?
- b bits can restrict i to domain of size $n/2^b$.
- Gaussian noise ℓ_2^2 smaller by 2^b factor, so SNR improves to 2^b.
- $\frac{1}{2}\log(1 + \text{SNR}) \approx b/2 + O(1)$ bits in one measurement.
- $1 \to 2 \to 4 \to 8 \to \ldots \to \log n$ takes $\log\log n$ steps.

1-sparse recovery



Formal goal

- Avoids the lower bound, but how to make an algorithm?
- Find i from $x = \alpha e_i + w$.
  - "Given $2b$ bits of information:": given $S \ni i$ with

$$\cancel{|S| < n/B^2} \qquad \alpha^2/\|w_S\|_2^2 > B^2.$$

  - "Restrict to set of size $n/2^b$": find $S' \ni i$ with

$$\cancel{|S'| < 1 + 10|S|/B} \qquad \|w_{S'}\|_2^2 < 10\|w_S\|_2^2/B.$$

- Noise isn't necessarily uniform Gaussian!
- Everything outside of S is irrelevant.

Formal goal (restated)

- Essentially: get $\log B$ bits when SNR is B^2.
- Suppose $x = \alpha e_i + w$ with $\alpha^2 / \|w\|_2^2 > B^2/\delta$.
- Using $O(1)$ measurements, find $S \ni i$ with

$$\|w_S\|_2^2 < \|w\|_2^2 / B.$$

  with probability $1 - O(\delta)$.
- Extreme case: if $\alpha^2 / \|w\|_2^2 > 100n^2$, find i exactly.
  - $O(n)$-approximate 1-sparse recovery.

$O(n)$-approximate 1-sparse recovery

Getting $\log n$ bits when SNR is n

- Suppose $x = \alpha e_i + w$, with $\|w\|_1 < \alpha/(10n)$.
- *Almost* exactly sparse.
- Two measurements: $a = \sum x_i$, $b = \sum (n+i) x_i$.
- $b/a \approx (n+i)$, so output: round$(b/a - n)$.
  - If w were zero, it would be exact.
  - Since w is really close to zero, b/a is within $1/2$ of $n+i$.
- Gives $O(n)$-approximate ℓ_1 recovery.
- For ℓ_2, use random signs s_i.
  - $a = \sum s_i x_i$, $b = \sum (n+i) s_i x_i$.
  - Gives $O(n/\sqrt{\delta})$-approximate recovery that succeeds with probability $1 - \delta$.

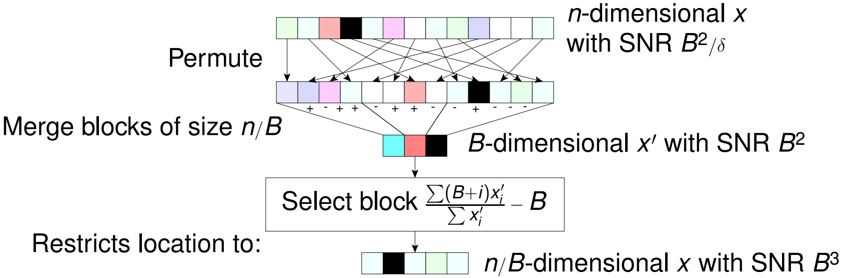
B-approximate 1-sparse recovery

Getting $\log B$ bits when SNR is B

- We've shown how to get $\log n$ bits in two measurements with SNR above n^2.
- We need to get $\log B$ bits from SNR B^2/δ for $1 \ll B < n$.
- Idea: reduce to case with SNR above n^2.
- Hash with random signs to B bins.

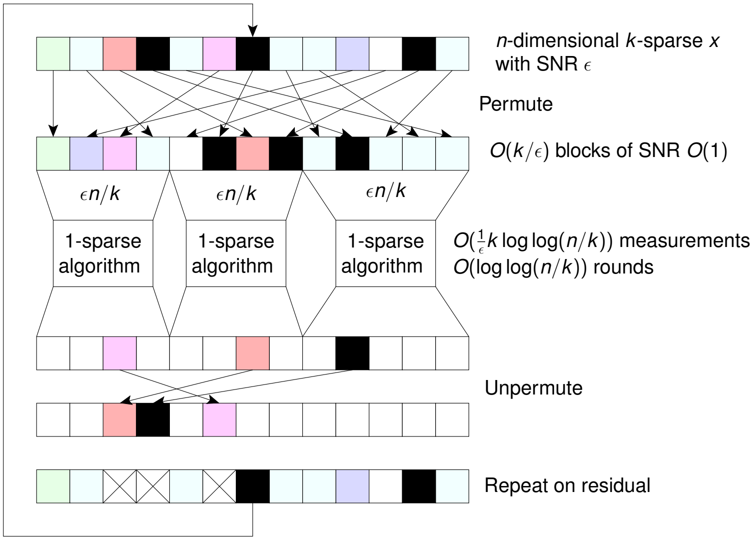
B-approximate 1-sparse recovery

Getting $\log B$ bits when SNR is B



- $1 - O(\delta)$ chance that
  - Noise doesn't increase by $1/\delta$, so the correct block is chosen.
  - The correct block has SNR within $1/\delta$ of expected B^3/δ.
- Thus SNR $B^2/\delta \to B^3$ with probability $1 - \delta$ using 2 measurements.
- In $\log \log n$ iterations, SNR grows from $O(1)$ to n^2 and we recover exactly.

1-sparse to k-sparse



Full Algorithm Outline

1. $O(n)$-approximate non-adaptive 1-sparse recovery
   - Observe $\sum s_i x_i$ and $\sum (n+i) s_i x_i$ for random signs s_i.
2. B-approximate non-adaptive 1-sparse recovery
   - Hash to $\Theta(B)$ buckets, perform (1) on those.
   - Finds a set with $O(1/\sqrt{B})$ of the noise.
   - Tangential: would give $O(\log_B n)$ non-adaptive measurements
3. $O(1)$-approximate adaptive 1-sparse recovery
   - Repeat (2) with $B_0 = O(1)$, $B_i = \Omega(B_{i-1}^{1.5})$.
   - Need error probabilities to converge: $B_i = \Omega(B_{i-1}^{1.5}/4^i)$
4. $1+\epsilon$-approximate adaptive k-sparse recovery
   - Subsample vector at rate ϵ/k, $O(\frac{1}{\epsilon} k \log(1/\delta))$ times.
   - Perform (3) on each subsampling, getting S containing $1-\delta$ of the heavy hitters with probability $1-\delta$.
   - Repeat on $[n] \setminus S$, setting $k \leftarrow \delta k$, $\delta \leftarrow 1/2^{1/2^i \delta}$.
   - Measure and output $x_{S_1 \cup S_2 \cup \ldots}$

Interlude

- We've shown an algorithm for $1+\epsilon$-approximate *adaptive* sparse recovery.
  - $O(\frac{1}{\epsilon}k \log\log(n/k))$ measurements.
  - $O(\log^* k \log\log n)$ rounds.
- We also give a two round algorithm with $O(\frac{1}{\epsilon}k \log(k/\epsilon) + k\log(n/k))$ space.
  - Separate the dependence on n and ϵ.

Separating ϵ and n

- Hash to $O(k^2/\epsilon^2)$ blocks, and probably all of:
  - A perfect hash, so heavy hitters land in different blocks.
  - Each heavy hitter dominates the noise in the same block.
  - Overall, the noise grows by at most $1 + \epsilon/2$ factor
- Solve $(1 + \epsilon)$-approximate sparse recovery in reduced space: $O(\frac{1}{\epsilon} k \log k)$
- Identifies $O(k)$ blocks to search containing enough heavy hitter mass.
- Heavy hitters are $O(1)$-heavy among their blocks, so $O(\log n)$ per block suffices.
- Result: $O(\frac{1}{\epsilon} k \log k + k \log n)$.

Summary and future work

- We give an algorithm for $1+\epsilon$-approximate *adaptive* sparse recovery.
  - $O(\frac{1}{\epsilon}k \log\log(n/k))$ measurements.
  - $O(\log^* k \log\log n)$ rounds.
- We give a two round algorithm with $O(\frac{1}{\epsilon}k \log(k/\epsilon) + k\log(n/k))$ space.
  - Separate the dependence on n and ϵ.
- Lower bound?
- Spread the measurements more evenly among rounds?
  - Given 1GB space, how many passes do we need?
- Clearer characterization of measurement/round tradeoff?
  - Given 4 iterations, how many total blood tests do we need?
- More general bounds for adaptivity?

Thoughts on the lower bound

- We still know the first measurement gets at most $O(1)$ bits.
- Hence after b bits, you should only get $O(2^b)$ bits: else you could guess the b bits and get $\omega(1)$ bits in expectation the first time.
  - Although this is information theory, not actual bits...
- Hence $\log^* n$ measurements necessary for $k = 1$.
- For general k, split vector into k chunks of size n/k, and use a direct product theorem?
  - Again, information theoretic capacity, not number of bits...