Problem Set 3

Sublinear Algorithms

Due Tuesday, October 28

1. Recall that $M(X, d, \epsilon)$ denotes the packing number for space X with distance d and radius ϵ, and $N(X, d, \epsilon)$ denotes the covering number. Prove that
$$M(X, d, 2\epsilon) \leq N(X, d, \epsilon) \leq M(X, d, \epsilon)$$

2. Give an algorithm to construct the k-tree-sparse approximation of a vector. The input is an integer k and a complete n-vertex binary tree T with a nonnegative value x_v associated with each vertex v. The output is the set S of size k that includes the root, is a connected subset of the tree, and maximizes $\sum_{v \in S} x_v$.

   (a) Show a simple DP algorithm to solve this in $O(nk^2)$ time and $O(nk)$ space.

   (b) Show how to optimize the algorithm to take $O(nk)$ time.

3. In this problem we show that matrices that satisfy the RIP-2 cannot be very sparse. Let $A \in \mathbb{R}^{m \times n}$ satisfy the $(k, 1/2)$ RIP for $m < n$. Suppose that the average column sparsity of A is d, i.e. A has nd nonzero entries.

   Furthermore, suppose that $A \in \{0, \pm\alpha\}^{m \times n}$ for some parameter α.

   (a) By looking at the sparsest column, give a bound for α in terms of d.

   (b) By looking at the densest row, give a bound for α in terms of n, m, d and k.

   (c) Conclude that $d \gtrsim k$. (Recall that this means: there exists a constant C for which $d \geq k/C$.)

(d) What if each non-zero $A_{i,j}$ were drawn from $N(0,1)$?

(e) [Optional] Extend the result to general settings of the non-zero $A_{i,j}$.

4. In class we have shown various algorithms for sparse recovery that tolerate noise and use $O(k\log(n/k))$ measurements, and shown that any ℓ_1/ℓ_1 sparse recovery algorithm must use this many measurements. But what if we don't care about tolerating noise, and only want to recover x from Ax when x is exactly k-sparse?

   Consider the matrix

$$A = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{2k-1} & \alpha_2^{2k-1} & \dots & \alpha_n^{2k-1} \end{pmatrix}$$

   for distinct α_i.

   (a) Prove that any $2k \times 2k$ submatrix of A is invertible.

   (b) Give an $n^{O(k)}$ time algorithm to recover x from Ax under the assumption that x is k-sparse.

   (c) [Optional] Give an $n^{O(1)}$ time algorithm to recover x from Ax under the assumption that x is k-sparse. You may choose specific values for the α_i. Hint: look up syndrome decoding of Reed-Solomon codes.

5. In order to show that SSMP makes progress in each stage, we used a lemma that we will show in this problem.

   Let $x_1, \dots, x_k \in \mathbb{R}^d$, and suppose that

$$\sum_{i=1}^{k} \|x_i\|_1 \leq (1+\delta)\|\sum_{i=1}^{k} x_i\|_1$$

   for some small enough δ (say, $\delta = 1/10$). In some sense, this is saying that there is not much "slack" in they are lined up head-to-tail.

(a) Let $z = \sum_{i=1}^{k} x_i$. Show that there exists an i such that $\|z - x_i\|_1 \leq (1 - \frac{\Omega(1)}{k})\|z\|_1$.

(b) Now suppose $z = \sum_{i=1}^{k} x_i + w$ for some $w \in \mathbb{R}^d$ with $\|w\|_1 \leq \epsilon\|z\|_1$ for small enough constant ϵ. Again, show that there exists an i such that $\|z - x_i\|_1 \leq (1 - \frac{\Omega(1)}{k})\|z\|_1$.