CS 225 - Pseudorandomness Prof. Salil Vadhan

Problem Set 2

Harvard SEAS - Spring 2015 *Due: Fri. Feb. 27, 2015*

Your problem set solutions must be typed (in e.g. LaTeX) and submitted electronically to `cs225-hw@seas.harvard.edu`. You are allowed 12 late days for the semester, of which at most 5 can be used on any individual problem set. (1 late day = 24 hours exactly). Please name your file `PS2-lastname.*`.

The problem sets may require a lot of thought, so be sure to start them early. You are encouraged to discuss the course material and the homework problems with each other in small groups (2-3 people). Identify your collaborators on your submission. Discussion of homework problems may include brainstorming and verbally walking through possible solutions, but should not include one person telling the others how to solve the problem. In addition, each person must write up their solutions independently, and these write-ups should not be checked against each other or passed around.

Strive for clarity and conciseness in your solutions, emphasizing the main ideas over low-level details. Do not despair if you cannot solve all the problems! Difficult problems are included to stimulate your thinking and for your enjoyment, not to overwork you. *ed problems are extra credit.

Problem 2.9.(Spectral Graph Theory) Let M be the random-walk matrix for a d-regular *undirected* graph $G = (V, E)$ on n vertices. We allow G to have self-loops and multiple edges. Recall that the uniform distribution (or all-ones vector) is an eigenvector of M of eigenvalue $\lambda_1 = 1$. Prove the following statements. (Hint: for intuition, it may help to think about what the statements mean for the behavior of the random walk on G.)

1. All eigenvalues of M have absolute value at most 1.

2. G is disconnected $\iff$ 1 is an eigenvalue of multiplicity at least 2.

3. Suppose G is connected. Then G is bipartite $\iff$ -1 is an eigenvalue of M.

4. G connected $\Rightarrow$ all eigenvalues of M other than λ_1 are at most $1 - 1/\text{poly}(n, d)$. To do this, it may help to first show that the second largest eigenvalue of M (not necessarily in absolute value) equals
$$\max_x \langle xM, x\rangle = 1 - \frac{1}{d} \cdot \min_x \sum_{(i,j) \in E} (x_i - x_j)^2,$$
where the maximum/minimum is taken over all vectors x of length 1 such that $\sum_i x_i = 0$, and $\langle x, y\rangle = \sum_i x_i y_i$ is the standard inner product. For intuition, consider restricting the above maximum/minimum to $x \in \{+\alpha, -\beta\}^n$ for $\alpha, \beta > 0$.

5. G connected and nonbipartite $\Rightarrow$ all eigenvalues of M (other than 1) have absolute value at most $1 - 1/\text{poly}(n, d)$ and thus $\lambda(G) \leq 1 - 1/poly(n, d)$.

6* Establish the (tight) bound $1 - \Omega(1/d \cdot D \cdot n)$ in Part 4, where D is the diameter of the graph. Conclude that $\gamma(G) = \Omega(1/d^2n^2)$ if G is connected and nonbipartite.

Problem 3.1.(Derandomizing RP versus BPP) Show that **prRP** = **prP** implies that **prBPP** = **prP**, and thus also that **BPP** = **P**. (Hint: Look at the proof that **NP** = **P** ⇒ **BPP** = **P**.)

Problem 3.2.(Designs) Designs (also known as packings) are collections of sets that are nearly disjoint. In Chapter 7, we will see how they are useful in the construction of pseudorandom generators. Formally, a collection of sets $S_1, S_2, \ldots, S_m \subset [d]$ is called an (ℓ, a)-*design* (for integers $a \le \ell \le d$) if

- For all i, $|S_i| = \ell$.
- For all $i \ne j$, $|S_i \cap S_j| < a$.

For given ℓ, we'd like m to be large, a to be small, and d to be small. That is, we'd like to pack many sets into a small universe with small intersections.

1. Prove that if $m \le \binom{d}{a}/\binom{\ell}{a}^2$, then there exists an (ℓ, a)-design $S_1, \ldots, S_m \subset [d]$.
   Hint: Use the Probabilistic Method. Specifically, show that if the sets are chosen randomly, then for every $S_1, \ldots, S_{i-1}$,
   $$\mathop{\mathrm{E}}_{S_i}\left[\#\{j < i : |S_i \cap S_j| \ge a\}\right] < 1.$$
2. Conclude that for every constant $\gamma > 0$ and every $\ell, m \in \mathbb{N}$, there exists an (ℓ, a)-design $S_1, \cdots, S_m \subseteq [d]$ with $d = O\left(\frac{\ell^2}{a}\right)$ and $a = \gamma \cdot \log m$. In particular, setting $m = 2^\ell$, we fit exponentially many sets of size ℓ in a universe of size $d = O(\ell)$ while keeping the intersections an arbitrarily small fraction of the set size.
3. Using the Method of Conditional Expectations, show how to construct designs as in Parts 1 and 2 *deterministically* in time poly(m, d).

Problem 3.4.(Almost Pairwise Independence) A family of functions $\mathcal{H}$ mapping domain $[N]$ to range $[M]$ is ε-*almost pairwise independent*[1] if for every $x_1 \ne x_2 \in [N]$, $y_1, y_2 \in [M]$, we have
$$\Pr_{H \overset{\mathrm{R}}{\leftarrow} \mathcal{H}}[H(x_1) = y_1 \text{ and } H(x_2) = y_2] \le \frac{1+\varepsilon}{M^2}.$$

1. Show that there exists a family $\mathcal{H}$ of ε-almost pairwise independent functions from $\{0,1\}^n$ to $\{0,1\}^m$ such that choosing a random function from $\mathcal{H}$ requires only $O(m + \log n + \log(1/\varepsilon))$ random bits (as opposed to $O(m+n)$ for exact pairwise independence). (Hint: First consider domain $\mathbb{F}^{d+1}$ for an appropriately chosen finite field $\mathbb{F}$ and $d \in \mathbb{N}$, and look at maps of the form $h = g \circ f_a$, where g comes from some pairwiase independent family and $f_a : \mathbb{F}^{d+1} \to \mathbb{F}$ is defined by $f_a(x_0, \ldots, x_d) = x_0 + x_1 a + x_2 a^2 + \cdots + x_d a^d$.)

[1] Another common definition of ε-almost pairwise independence requires instead that for every $x_1 \ne x_2 \in [N]$, if we choose a random hash function $H \overset{\mathrm{R}}{\leftarrow} \mathcal{H}$, the random variable $(H(x_1), H(x_2))$ is ε-close to two uniform and independent elements of $[M]$ in statistical difference (as defined in Section 6). The two definitions are quivalent up to a factor of M^2 in the error parameter ε.

2. Give a deterministic algorithm that on input an N-vertex, M-edge graph G (with no self-loops), finds a cut of size at elast $(1/2 - o(1)) \cdot M$ in time $M \cdot \mathrm{polylog}(N)$ and space $O(\log M)$ (thereby improving the $M \cdot poly(N)$ running time of Algorithm 3.20).