

Advanced Algorithm 26 Homework 1

September 2026

1 Max-Cut

Let $G = (V, E)$ be an undirected graph with nonnegative edge weights $w_e \geq 0$. Let OPT be the maximum cut weight, and define $W = \sum_{e \in E} w_e$.

Consider the randomized algorithm that assigns each vertex v an independent uniform bit $X_v \in \{0, 1\}$ and outputs the cut defined by these bits: an edge $\{u, v\}$ is cut if and only if $X_u \neq X_v$. Its expected cut weight is $W/2 \geq \text{OPT}/2$.

Derandomize this algorithm in the following two ways.

1. Use the method of conditional expectations to fix the vertex bits one at a time. Prove that the conditional expected cut weight never decreases and that the resulting cut has weight at least $W/2$.
2. Let $N = |V|$ and $k = \lceil \log_2(N + 1) \rceil$. Assign each vertex a distinct nonzero vector $a_v \in \mathbb{F}_2^k$. For a uniformly random seed $b \in \mathbb{F}_2^k$, define $X_v = \langle a_v, b \rangle \pmod{2}$. Prove that these bits are pairwise independent and uniform. Show that enumerating all seeds b yields a cut of weight at least $W/2$, and analyze the running time.

2 Multiset Fingerprinting

Let $N \geq 1$. Two insertion-only streams define multisets S and T over $[U] = \{1, \dots, U\}$: each item in a stream inserts one occurrence of that element into the corresponding multiset. Each multiset contains at most N elements, counting multiplicities.

Let $p > \max\{U, N\}$ be a prime. Assume that the contents of both streams are fixed before $z \in \mathbb{F}_p$ is chosen uniformly at random. Define the following polynomials over $\mathbb{F}_p$:

$$F_S(Z) = \prod_{a \in S} (Z - a), \quad F_T(Z) = \prod_{a \in T} (Z - a).$$

(Hint: You may use the fact that a nonzero univariate polynomial of degree d over a field has at most d roots.)

1. Design a one-pass streaming algorithm that uses $F_S(z)$ and $F_T(z)$ to decide whether $S = T$. Prove that the algorithm always answers correctly when $S = T$ and has error probability at most N/p when $S \neq T$. Analyze its space complexity. (*Hint: S and T are multisets, so elements may appear more than once.*)
2. Suppose each stream contains exactly N elements. For each $j = 1, \dots, N$, use the algorithm from the previous part to test whether the first j elements of the two streams form the same multiset. Use the same value of z for all N tests. Prove that the probability that at least one test gives an incorrect answer is at most $\frac{1}{p} \sum_{j=1}^N j = \frac{N(N+1)}{2p}$. For $\delta \in (0, 1)$, how large should p be to ensure that all N answers are correct with probability at least $1 - \delta$? Is it necessary to choose a new random value of z for each test?
3. For this part, replace the two insertion-only streams by a sequence of operations on S and T . Each operation either inserts one occurrence of an element into one of the multisets, deletes one occurrence of an element currently present in that multiset, or asks whether the current multisets are equal. Assume that each multiset always contains at most N elements. The sequence contains at most Q equality queries and is fixed before z is chosen.
Extend your algorithm to process this sequence using the same z and $O(\log p + \log N)$ space. Describe how each update changes only a constant number of stored values. Prove that the probability that at least one equality query is answered incorrectly is at most QN/p . (*Hint: When deleting an element a , consider the case $z = a$.*)

3 FKS Perfect Hashing

Let S be a fixed set of $n \geq 2$ distinct keys from a universe $\mathcal{U}$. For every table size m , assume that we can choose a random hash function $h : \mathcal{U} \rightarrow [m]$ such that, for any distinct keys $x, y \in \mathcal{U}$, $\mathbb{P}[h(x) = h(y)] \leq \frac{1}{m}$. Assume that each key and each hash function can be stored in $O(1)$ space. Choosing a hash function, evaluating it, and comparing two keys each take $O(1)$ time. All choices of hash functions are independent.

The FKS dictionary uses two levels of hashing. At the first level, choose a hash function $h_0 : \mathcal{U} \rightarrow [n]$ and partition S into buckets

$$S_i = \{x \in S : h_0(x) = i\}, \quad i \in [n].$$

Let $b_i = |S_i|$. For each nonempty bucket S_i , allocate a second-level table of size b_i^2 and choose a hash function $h_i : \mathcal{U} \rightarrow [b_i^2]$. If h_i has a collision on S_i , choose a new h_i until it is collision-free on S_i .

To answer a membership query for a key x , first compute $i = h_0(x)$ and then inspect position $h_i(x)$ in the second-level table for S_i .

1. Prove that $\mathbb{E} \left[\sum_{i=1}^n b_i^2 \right] \leq 2n - 1$. Suppose we repeatedly choose a new first-level hash function until $\sum_{i=1}^n b_i^2 \leq 4n$. Prove that the expected number of first-level hash functions chosen before this condition is satisfied is at most 2.

2. Fix a first-level hash function for which $\sum_{i=1}^n b_i^2 \leq 4n$. Prove that, for each nonempty bucket S_i , a random choice of h_i is collision-free on S_i with probability at least $1/2$.

If a collision occurs in a bucket, choose a new hash function for that bucket and try again. Prove that the complete data structure can be constructed in expected $O(n)$ time and uses $O(n)$ space. Describe what is stored in each table entry and how membership queries can be answered in worst-case $O(1)$ time.

3. Consider a global rebuilding strategy: if any second-level table has a collision, discard all second-level hash functions and choose new ones for every bucket. Suppose n is even and the first level has exactly $n/2$ nonempty buckets, each containing two keys. Each bucket uses a second-level table of size 4. In every rebuilding attempt, choose the positions of all keys independently and uniformly in their respective tables.

Determine the probability that one rebuilding attempt is collision-free in every bucket. Deduce that the expected number of rebuilding attempts is exponential in n .

4 Concentration of Measure

Let $G \sim G(n, p)$ be a random graph on $n \geq 3$ vertices, where each pair of distinct vertices is joined by an edge independently with probability p . Let $d = (n - 1)p$ be the expected vertex degree.

1. Prove that there is an absolute constant $C_1 > 0$ such that, if $d \geq C_1 \log n$, then with probability at least 0.99, all vertices v simultaneously satisfy $0.9d \leq \deg_G(v) \leq 1.1d$.
2. For $0 < p_1 < p_2 < 1$, let $G_1 \sim G(n, p_1)$ and $G_2 \sim G(n, p_2)$. Let $\kappa(G)$ be the number of connected components of G . Compare $\mathbb{E}[\kappa(G_1)]$ and $\mathbb{E}[\kappa(G_2)]$ and prove it.
3. For $S \subseteq V$, let ∂S be the set of edges with exactly one endpoint in S . Prove that there is an absolute constant $C_2 > 0$ such that, if $d \geq C_2 \log n$, then with probability at least 0.99, every set $S \subseteq V$ with $1 \leq |S| \leq n/2$ satisfies $|\partial S| \geq d|S|/4$. Deduce that under the same assumption, G is connected with probability at least 0.99.

5 Dimension Reduction

Inner Product. Fix an integer $d > 0$ and parameters $\delta, \varepsilon \in (0, 1/2)$. Let $A \in \mathbb{R}^{k \times d}$ be a random matrix with $k = O(\log(1/\delta)/\varepsilon^2)$ rows, and the entries of A are chosen i.i.d. from a Gaussian distribution with mean 0 and variance $1/k$. Prove that for any fixed $x, y \in \mathbb{R}^d$, with probability at least $1 - \delta$ over the choice of A , it holds that $|x^\top y - (Ax)^\top (Ay)| \leq \varepsilon(\|x\|_2^2 + \|y\|_2^2)$.

Linear Separability In machine learning, the goal of many classification methods is to separate data into classes using a hyperplane. A hyperplane in $\mathbb{R}^d$ is characterized by a unit vector $a \in \mathbb{R}^d$ ($\|a\|_2 = 1$) and $c \in \mathbb{R}$. It contains all $z \in \mathbb{R}^d$ such that $a^\top z = c$. Suppose our dataset consists of

n unit vectors in $\mathbb{R}^d$. These points can be separated into two linearly separable sets X, Y , where $|X| + |Y| = n$. That is, for all $x \in X, a^\top x > c$ and for all $y \in Y, a^\top y < c$ (or vice versa). Furthermore, suppose that the ℓ_2 distance of each point in X and Y to this separating hyperplane is at least ε . When this is the case, the hyperplane is said to have margin ε .

1. Show that X, Y can be separated by the hyperplane characterized by $a \in \mathbb{R}^d$ ($\|a\|_2 = 1$) and $c \in \mathbb{R}$ with margin ε is equivalent to the following condition: for all $x \in X, a^\top x \geq c + \varepsilon$ and for all $y \in Y, a^\top y \leq c - \varepsilon$ (or vice versa).
2. Show that if we use a Johnson-Lindenstrauss map $A \in \mathbb{R}^{k \times d}$ (the scaled Gaussian matrix given in the lecture) to reduce our data points to $O((\log n)/\varepsilon^2)$ dimensions, then with probability at least $9/10$, the dimension-reduced data can still be separated by a hyperplane with margin $\varepsilon/4$. (*Hint*: use the fact that the Johnson-Lindenstrauss Lemma preserves inner product.)

6 Lovász Local Lemma

Independent Representatives Let $G = (V, E)$ be a simple undirected graph with $|V| = n$ and maximum degree Δ . Suppose the vertex set V is partitioned into r disjoint sets $S_1, \dots, S_r$, each of size at least $2e\Delta$.

1. Prove that we can choose one vertex from each S_i so that no two chosen vertices are adjacent.
2. Design a randomized algorithm that finds such a choice in expected time polynomial in n .

Directed Cycle Let $D = (V, E)$ be a simple directed graph with minimum outdegree $\delta \geq 1$ and maximum indegree Δ . Prove that for any positive integer k , if $e(\Delta\delta + 1)(1 - 1/k)^\delta < 1$, then D contains a simple directed cycle whose length is a positive multiple of k . (*Hint*: Consider a random coloring of the vertices with k colors.)