Approximate Nearest Neighbor Search via Group Testing

Saksham Rathi (22B1003)

Kshitij Vaidya (22B1829)

Ekansh Ravi Shankar (22B1032)

CS754: Advanced Image Processing
Under Prof. Ajit Rajwade

Indian Institute of Technology Bombay
Spring 2025

Contents

Nearest Neighbor Search

- Nearest neighbor search is a fundamental problem with many applications in machine learning systems.
- **Task:** Given a dataset $D = \{x_1, x_2, \ldots, x_N\}$, the goal is to build a data structure that can be queried with any point q to obtain a small set of points $x_i \in D$ that have high similarity (low distance) to the query. This structure is called an index.
- Such tasks frequently arise in genomics, web-scale data mining, machine learning, and other large-scale applications.

Locality Sensitive Hashing

- **Locality Sensitive Hashing (LSH)** algorithms use an LSH function to partition the dataset into buckets.
- The hash function is selected so that the distance between points in the same bucket is likely to be small.
- To find the near neighbors of a query, we hash the query and compute the distance to every point in the corresponding bucket.
- **Count-Based LSH** identifies neighbors by simply counting how many times two points land in the same hash bucket across multiple hash functions.

Formal Problem Statement

- **(R, c)-Approximate Near Neighbor:** Given a dataset D, if there exists a point within distance R of a query y, return some point within distance $c \cdot R$, with high probability.
  - R is the distance threshold (radius).
  - $c > 1$ is the approximation factor.
- Any algorithm that solves the randomized nearest neighbor problem also solves the approximate near neighbor problem with $c = 1$ and any $R \geq$ distance to the nearest neighbor.
- (Definition) **Randomized Nearest neighbor:** Given a dataset D and a distance metric $d(\cdot, \cdot)$ and a failure probability $\delta \in [0, 1]$, construct a data structure which, given a query point y reports the point $x \in D$ with the smallest distance $d(x, y)$ with probability greater than $1 - \delta$.

Group Testing

- We are given a set D of N items, with k positives ("hits") and $N - k$ negatives ("misses").
- **Goal:** Identify all positive items using fewer than N group tests.
- A *group test* is positive iff at least one item in the group is positive.
- **Testing Variants:** Can be *noisy* (with false positives/negatives), *adaptive* (tests depend on previous results), or *non-adaptive* (all tests run in parallel).
- The paper uses a **doubly regular design:** Each item appears in an equal number of tests; each test has an equal number of items.