SEEM5020 2026 Spring: Homework 5 Solution

- **Q1. λ-Snapshot vs. Exponential-Histogram (DGIM)**

  Solution:

  - **Space Complexity:** To achieve ϵN absolute error, λ-Snapshot requirs $O(\frac{1}{\epsilon}\log\frac{1}{\epsilon})$ bits. DGIM requires $O(\frac{1}{\epsilon}\log^2 N)$ bits to achieve ϵ relative error.
  - **Query Flexibility:** DGIM is more flexible. It can answer queries for any arbitrary $k \leq N$ with (ϵ) relative error. In contrast, λ-Snapshot is optimized for a fixed window size N with error absolute ϵN, thus answering for arbitrary k would lead to significant relative errors when k is small.
  - **Generalizability:** DGIM generalizes naturally to summing non-negative integers by adjusting the bucket merging logic or by treating each bit position as a separate binary stream. While λ-Snapshot is more specialized, it is fundamentally a sampling-based counter for binary streams.
  - **Simplicity:** λ-Snapshot is significantly simpler to implement, requiring only a queue of indices and a periodic sampling rule based on the arrival of 1-bits. DGIM requires managing strict bucket constraints (at most two of each size) and complex merging logic.

- **Q2. Space-Efficient Euclidean LSH**

  Solution: Strategy: Let $a_1, a_2, \ldots, a_k$ be k random integers drawn uniformly and independently from $[0, P-1]$, we define $H(\mathbf{x}) = (\sum_{i=1}^{k} a_i h_i(\mathbf{x})) \pmod P$.

  Probability Analysis: For any two vectors $\mathbf{x}_1, \mathbf{x}_2$ such that $g(\mathbf{x}_1) \neq g(\mathbf{x}_2)$, the probability of a collision $H(\mathbf{x}_1) = H(\mathbf{x}_2)$ under a universal hash family is at most $1/P$. There are $\binom{n}{2}$ such pairs. By the Union Bound, the probability that there exists any collision is:

  $$\mathbb{P}[\text{any collision}] \leq \binom{n}{2} \cdot \frac{1}{P} < \frac{n^2}{2} \cdot \frac{1}{n^3} = \frac{1}{2n}.$$

  Space Reduction: Instead of an array of size P, we use a Hash Table using chaining with $m = O(n)$ slots. We store the $H(x)$ values in this table. To handle the query, we will need $O(k)$ time to compute $g(x)$ and $H(x)$.

- **Q3. Success Probability of LSH with Candidate Pruning**

  Solution: Let E_1 be the event that an r-near neighbor x does not collide with query q in any of the L tables. Let E_2 be the event that more than $5n^\rho$ "far" points collide with q across the L tables. The algorithm succeeds if neither E_1 nor E_2 occurs.

  From the parameters: $k = \log_{1/p_2} n \implies p_2^k = 1/n$. The collision probability for a near neighbor in one table is $p = p_1^k = p_1^{\log_{1/p_2} n} = n^{-\rho}$. The expected number of far points in one table is $\mathbb{E}[Y_{\text{table}}] \leq (n-1) \cdot p_2^k < 1$.

  - **Event E_1:** $\mathbb{P}[E_1] = (1-p)^L \leq (1-1/n^\rho)^{2n^\rho} \leq e^{-2}$.
  - **Event E_2:** $\mathbb{E}[Y_{\text{total}}] = L \cdot \mathbb{E}[Y_{\text{table}}] < 2n^\rho$. By Markov's Inequality: $\mathbb{P}[Y \geq 5n^\rho] \leq \frac{\mathbb{E}[Y]}{5n^\rho} < \frac{2n^\rho}{5n^\rho} = 0.4$.

  By the Union Bound, $\mathbb{P}[\text{Failure}] \leq \mathbb{P}[E_1] + \mathbb{P}[E_2] < e^{-2} + 0.4$. The success probability is at least $1 - (e^{-2} + 0.4) = 0.6 - e^{-2}$.

- **Q4. Using LSH to answer c-approximate Nearest Neighbor query**

  Solution: Apply the binary search on radius r with the (c, r)-ANNS query to answer the c-ANNS query. At first, apply $(\sqrt{c}, r)$-ANNS for r:

$$r = \{\frac{d_{min} \cdot \sqrt{c}^{-1}}{2}, \frac{d_{min}}{2}, \frac{d_{min} \cdot \sqrt{c}^{1}}{2}, \frac{d_{min} \cdot \sqrt{c}^{2}}{2} \cdots \frac{d_{max}}{\sqrt{c}-1}\}$$

  i.e., $r = \frac{d_{min}}{2} \cdot \sqrt{c}^{j}$, where $j = -1, 0, 1 \ldots \lceil \log_{\sqrt{c}}(\frac{2 \cdot d_{max}}{(\sqrt{c}-1)d_{min}}) \rceil$. We use binary search to find the minimum r that $(\sqrt{c}, r)$-ANNS returns a point p but $(\sqrt{c}, \frac{r}{\sqrt{c}})$-ANNS fails and we have $d(q, p) \leq \sqrt{c} \cdot r$. Thus, $\forall p' \in P, d(q, p') > \frac{r}{\sqrt{c}}$ for $(\sqrt{c}, \frac{r}{\sqrt{c}})$-ANNS fails, which is equivalent to the statement $r < \sqrt{c} \cdot \min_{p' \in P} d(q, p')$. Hence, we have

$$d(q, p) \leq \sqrt{c} \cdot r < c \cdot \min_{p' \in P} d(q, p')$$

  which means point p corresponds to the answer of c-ANNS query. Next consider the two cases where $r = \frac{d_{min} \cdot \sqrt{c}^{-1}}{2}$ or $r = \frac{d_{max}}{\sqrt{c}-1}$. When $r = \frac{d_{min} \cdot \sqrt{c}^{-1}}{2}$, if $(\sqrt{c}, \frac{d_{min} \cdot \sqrt{c}^{-1}}{2})$-ANNS returns a point p which has $d(q, p) \leq \frac{d_{min}}{2}$, then there is no other point $p' \in P$ that $d(q, p') < \frac{d_{min}}{2}$, otherwise, by triangle inequality, we will have $d(p, p') \leq d(q, p) + d(q, p') < d_{min}$, which contradicts the fact. Hence, the point p returned is the nearest neighbor of q. When $r = \frac{d_{max}}{\sqrt{c}-1}$, if $(\sqrt{c}, \frac{d_{max}}{\sqrt{c}-1})$-ANNS fails to return a point, which means $D = \min_{x \in P} d(q, x) = d(q, p') > \frac{d_{max}}{\sqrt{c}-1}$. By triangle inequality, we have:

$$\forall p \in P, d(q, p) \leq d(p, p') + d(q, p') \leq d_{max} + D < (\sqrt{c} - 1) \cdot D + D < \sqrt{c} \cdot D$$

  Hence, any point $p \in P$ is the answer of $\sqrt{c}$-ANNS query, and also the answer of c-ANNS query. Now we need $k = \log_2(\lceil \log_{\sqrt{c}}(\frac{2\sqrt{c} \cdot d_{max}}{(\sqrt{c}-1)d_{min}}) \rceil)$ $(\sqrt{c}, r)$-ANNS queries in total to obtain the answer of c-ANNS query. The failure probability of c-ANNS query is at most $\frac{1}{5}$, so the failure probability of each $(\sqrt{c}, r)$-ANNS query should be at most $\frac{1}{5k}$. According to Q3, set $L = \ln(10k) \cdot n^{\rho}$ and the stopping condition to retrieve at most $10 \cdot L$ data points to make the fail probability smaller than $\frac{1}{5k}$. Now, we can achieve a c-ANNS query.