CS246 Exam 2025 — Question 10: Mining Data Streams (12 points)

Setup

Two servers S_1 and S_2 each serve **m** distinct, non-overlapping IPs. Each maintains a Bloom filter of **n** bits with **k** hash functions. We must combine them into a single structure of at most n bits on a primary server.

Given: $k = (n/2m) \cdot \ln 2$ and $k \gg 1$.

Define:

- **$c(b_1, b_2)$:** Combine method → data structure of $\leqslant$ n bits
- **q(x):** Query method → was x served by either server?

Part 1 (6 points) — Same Hash Functions

S_1 and S_2 use the same k hash functions ($H^1\_j = H^2\_j$ for all j).

Combine: $c(b_1, b_2) = b_1$ OR b_2 (bitwise OR)

Query: q(x) = check if B_combined[H_j(x)] = 1 for all j = 1, …, k

Standard Bloom filter membership test on the combined array. **No false negatives:** if x was served by either server, all its hash positions were set to 1 in that server's filter, and OR preserves all 1s.

False Positive Probability

The combined filter is equivalent to a single Bloom filter with **2m** items in n bits with k hash functions. Computing the fill fraction:

$$\frac{k \cdot 2m}{n} = \frac{n}{2m} \cdot \ln 2 \cdot \frac{2m}{n} = \ln 2$$

$$\Pr(\text{bit} = 1) = 1 - e^{-\ln 2} = \frac{1}{2}$$

Exactly half the bits are 1 — the optimal operating point. A false positive occurs when all k positions for a non-member are 1:

$$\boxed{\Pr(\text{false positive}) = \left(\frac{1}{2}\right)^k}$$

Part 2 (6 points) — Different Hash Functions

S_1 and S_2 use independently chosen hash functions ($H^1{}_1 \ldots_k$ and $H^2{}_1 \ldots_k$).

Combine: $c(b_1, b_2) = b_1$ OR b_2 (same as Part 1)

Query: q(x) = return YES if either check passes:

- B_combined[H^1_j(x)] = 1 for all j = 1, …, k (S_1's hash functions), **OR**
- B_combined[H^2_j(x)] = 1 for all j = 1, …, k (S_2's hash functions)

Since the hash functions differ, we can't use a single check — we run **two separate membership tests** and accept if either passes. **No false negatives** by the same argument as Part 1.

False Positive Probability

Fill fraction: Each server independently sets bits. A bit is 0 only if neither server set it:

$$\Pr(\text{bit} = 0) = e^{-km/n} \cdot e^{-km/n} = e^{-2km/n} = e^{-\ln 2} = \frac{1}{2}$$

So Pr(bit = 1) = 1/2, same as Part 1.

Each individual check has false positive rate (1/2)^k. Applying the **union bound** over both checks:

$$\Pr(\text{FP}) \leq \left(\frac{1}{2}\right)^k + \left(\frac{1}{2}\right)^k = 2 \cdot \left(\frac{1}{2}\right)^k$$

$$\boxed{\Pr(\text{false positive}) \leq \left(\frac{1}{2}\right)^{k-1}}$$

Comparison

	Same Hash Functions	Different Hash Functions
Combine	b_1 OR b_2	b_1 OR b_2
Query	Single check with shared H	Two checks, one per server's H
FP probability	(1/2)^k	⩽ 2 · (1/2)^k = (1/2)^(k−1)

Different hash functions cost roughly a **factor of 2** in false positive probability — two separate tests each independently produce false positives, whereas shared hash functions allow a single unified test.