L6: Locality Sensitive Hashing

Jeff M. Phillips

January 27, 2025

Documents

D_1 → S_1

D_2

$\vdots$

D_n S_n

min hash vectors

m_1

m_2

$\vdots$

m_n

$m_j \in \mathbb{R}^t$

1 : if True

0 : if False

$$JS(S_1, S_2) = E\left[\frac{1}{t}\sum_{i=1}^{t} \mathbb{1}\left(m_1[i] = m_2[i]\right)\right]$$

$t \approx 1000$

$$P_r\left[\left|JS(S_1, S_2) - \hat{JS}\right| > \epsilon\right] < \delta$$

Q1: Have n documents and query q

Find similar docs to q

time $\ll$ linear in n.

Ex: Search engine

RAG

Q2: Have n documents,

find all similar pairs

time $\ll$ quadratic in n

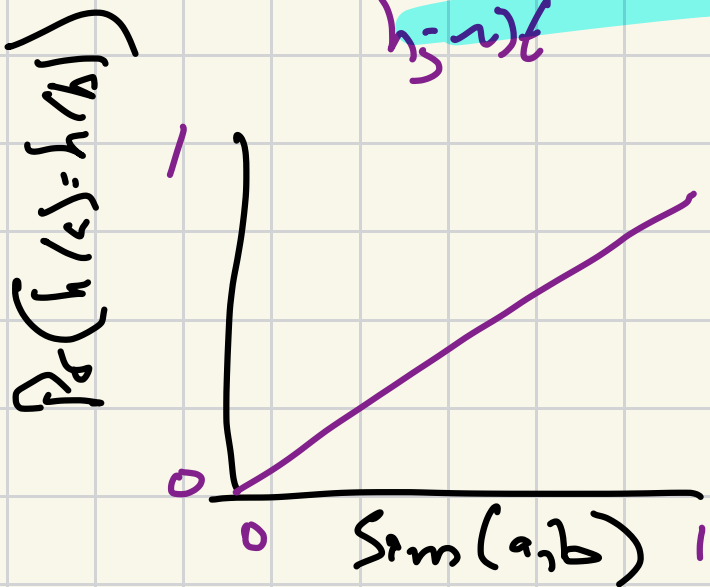
Ex: check plagarism.

Locality Sensitive Hashing

$h_1, h_2 \dots h_t \sim \mathcal{H}$ ~~before~~, ~~random hashes~~ $h_j : \Sigma^l \to [c]$

$\Pr_{h_j \sim \mathcal{H}}[h_j(a) = h_j(b)] = \frac{1}{c}$ ~~$a \neq b$~~

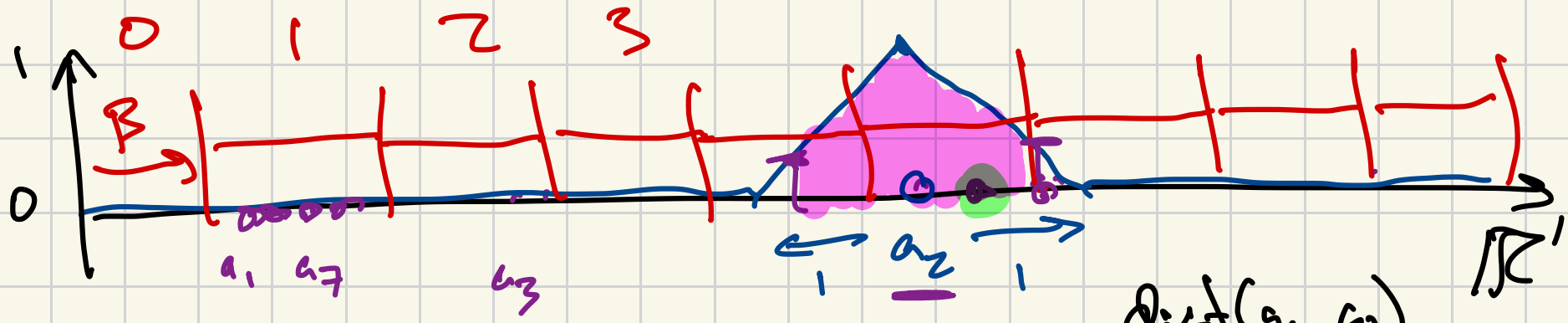
LSH: $\Pr_{h_j \sim \mathcal{H}}[h_j(a) = h_j(b)] = Sim(a,b) \in [0,1]$



min hash

$\Pr[m_a(i) = m_b(i)] = JS(a,b)$

Triangle Similarity objects $a_1, a_2 \dots a_n \in \mathbb{R}^1$



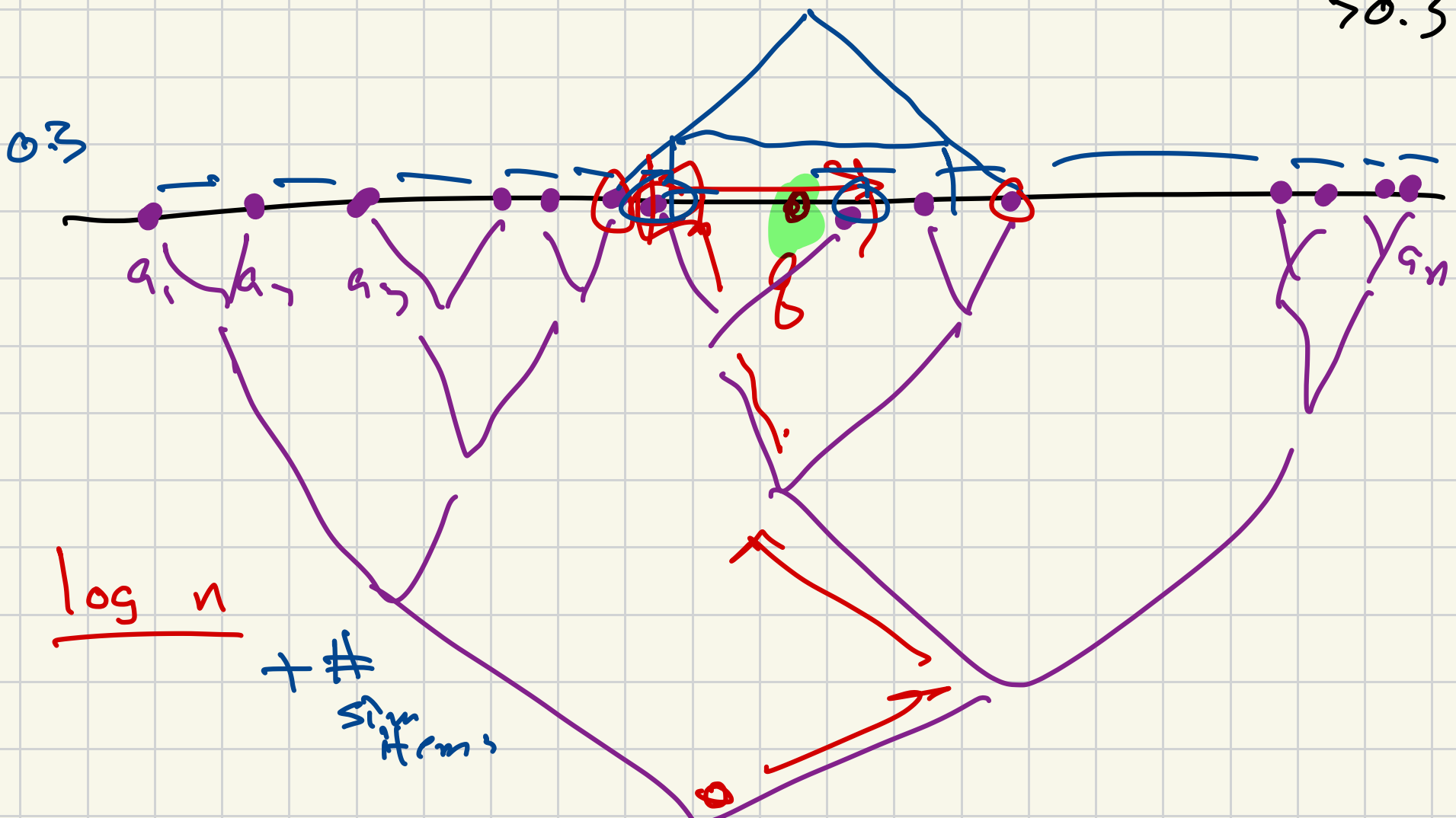
$$\text{dist}(a_1, a_2)$$

$$\text{Sim}_\Delta(a_1, a_2) = 1 - \min\{1, \|a_1 - a_2\|\}$$

$$\beta \sim \text{Unif}(0, 1) \qquad h_j = \text{bin}(\text{offset } \beta)$$

$$\Pr_{\beta \sim \text{Unif}}\left[h_\beta(a_1) = h_\beta(a_2)\right] = \text{Sim}_\Delta(a_1, a_2)$$

Under Sim_Δ, find all objects from $a_1, a_2, \ldots a_n$ with $Sim_\Delta(a_i, q) > 0.3$



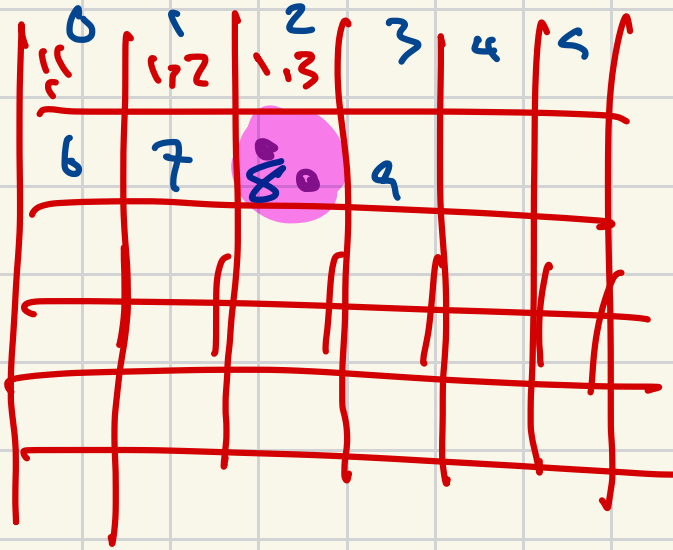
	S_1	S_2	S_3	S_4	...	S_n
h_1	1	1	3	4		1
h_2	2	0	3	1		2
h_3	2	3	2	1		1
h_t	4	4	3	2		3

AND

h_1, h_2

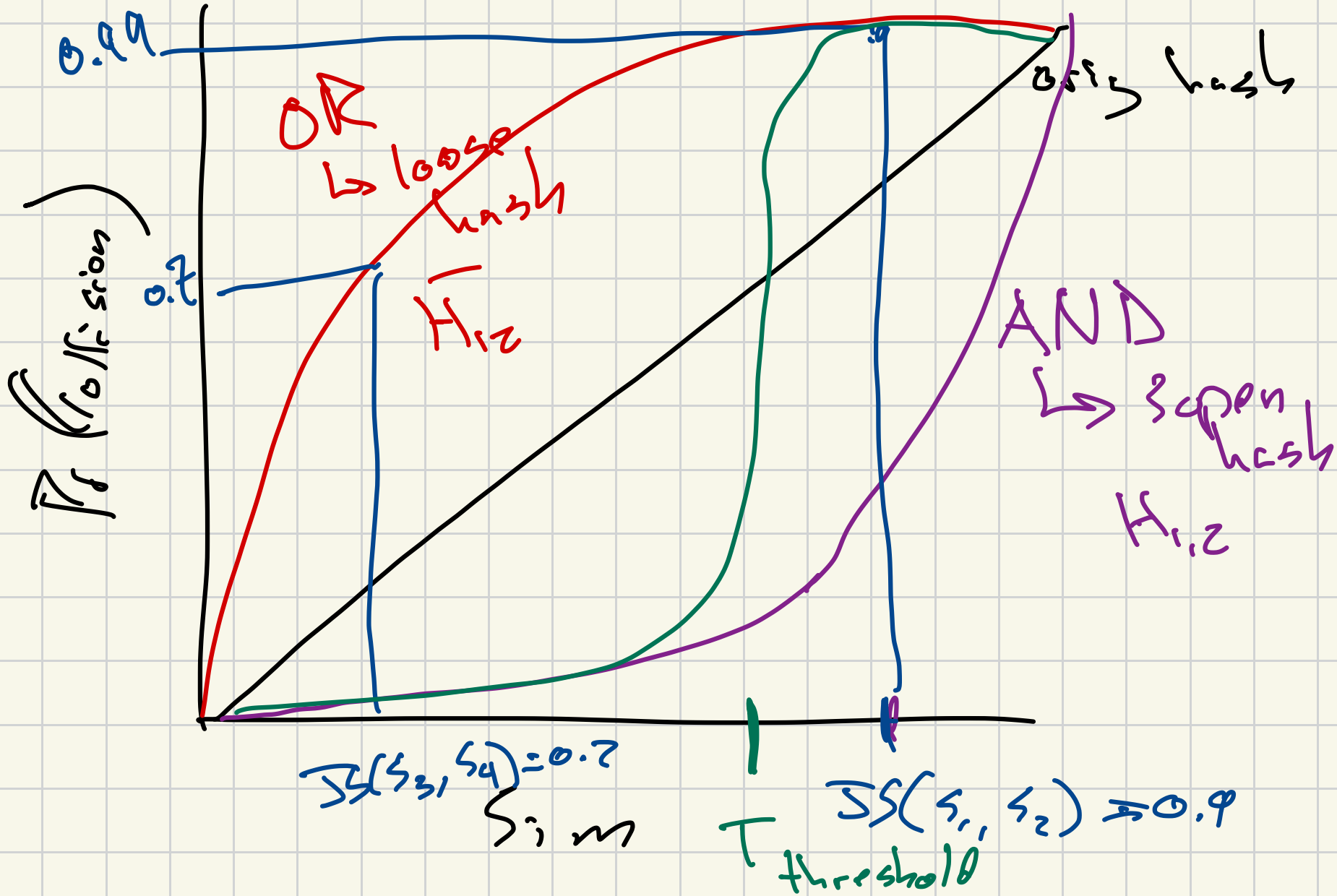
$H_{1,2}(S_1) = (1,2)$

$H_{1,2}(S_1) \stackrel{?}{=} H_{1,2}(S_4)$



$P_r[H_{1,2}(S_1) = H_{1,2}(S_2)]$

$\stackrel{?}{=} Sim(S_1, S_2)$


0.99
0.7
Pr (collision)
OR
↳ loose hash
$H_{1,2}$
big hash
AND
↳ 3 open hash
$H_{1,2}$
$JS(S_3, S_4) = 0.7$
S_3 S_4
$T_{threshold}$
$JS(S_1, S_2) = 0.9$
S_1 S_2

OR between hash cells

$\overline{H_{1,2}}(s_1) = \overline{H_{1,2}}(s_2)$ if

$h_1(s_1) = h_1(s_2)$ or $h_2(s_1) = h_2(s_2)$

Banding: combines together AND and OR hashes

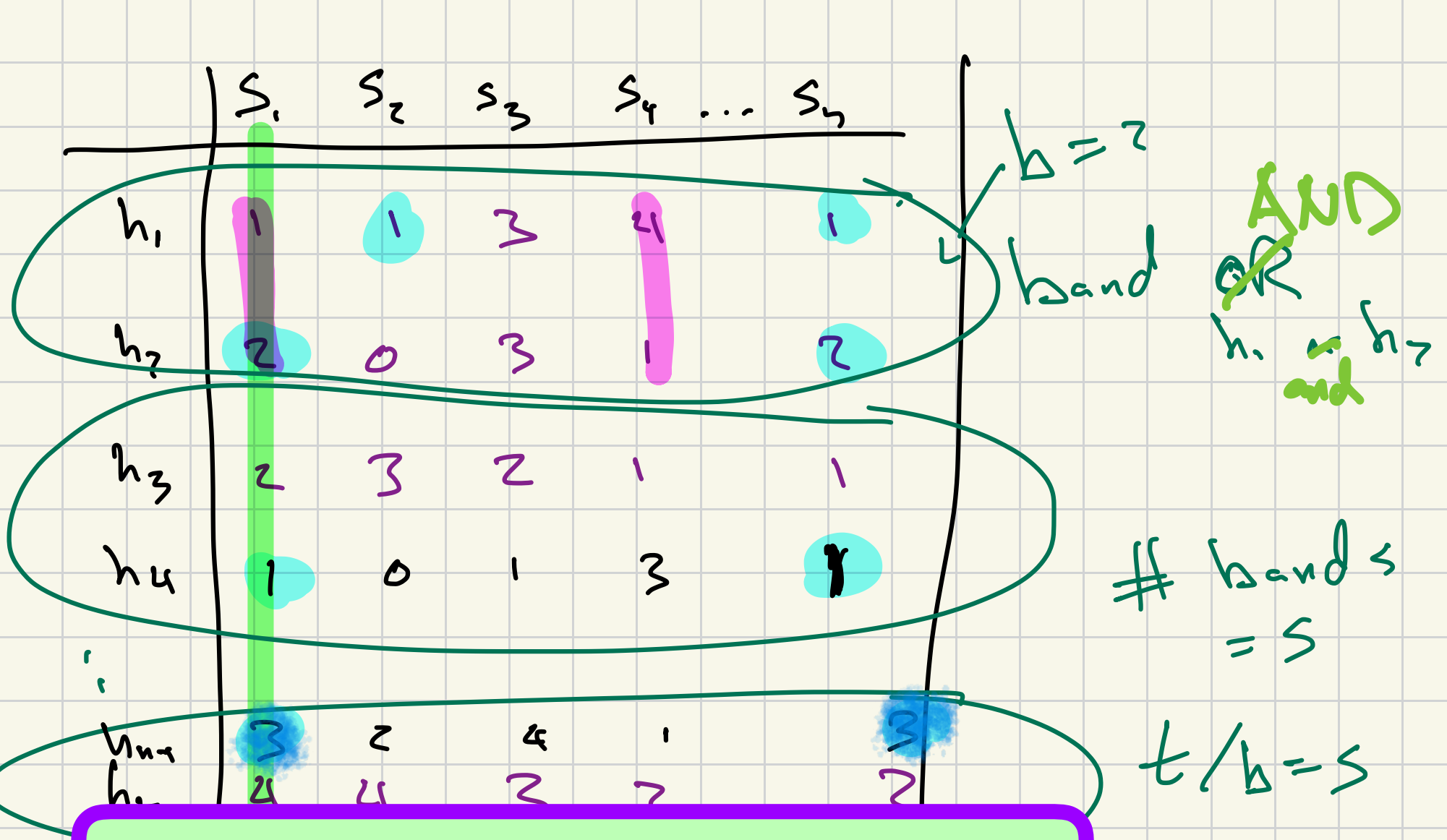
t: total # hash functions

b: # within 1 band (AND combinations)

s: # of bands (OR combo)

$t = b \cdot s$

The lecture reversed the order of AND and OR combinations of hashes — on this slide

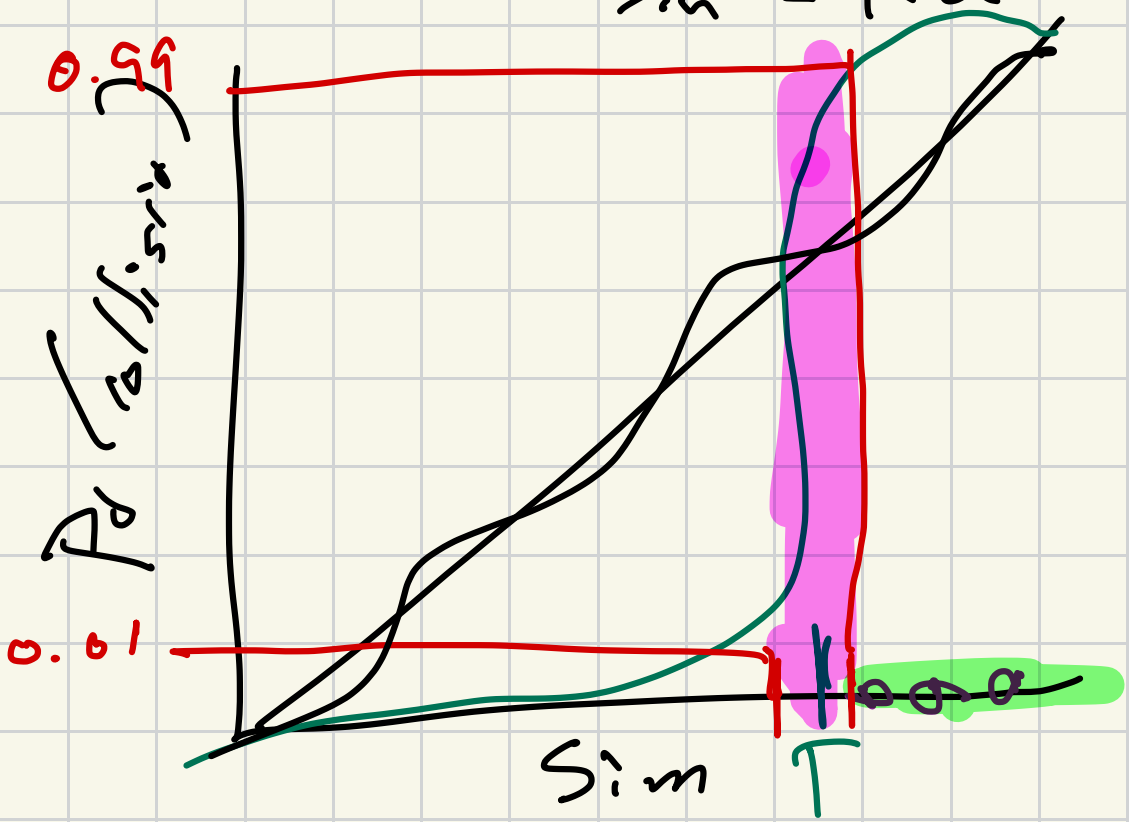


The lecture reversed the order of AND and OR combinations of hashes — on this slide

Before $\Pr[h(a)=h(b)] = Sim$

After $\Pr[h^*(a)=h^*(b)] > 0.99$ if $Sim > T+\alpha$

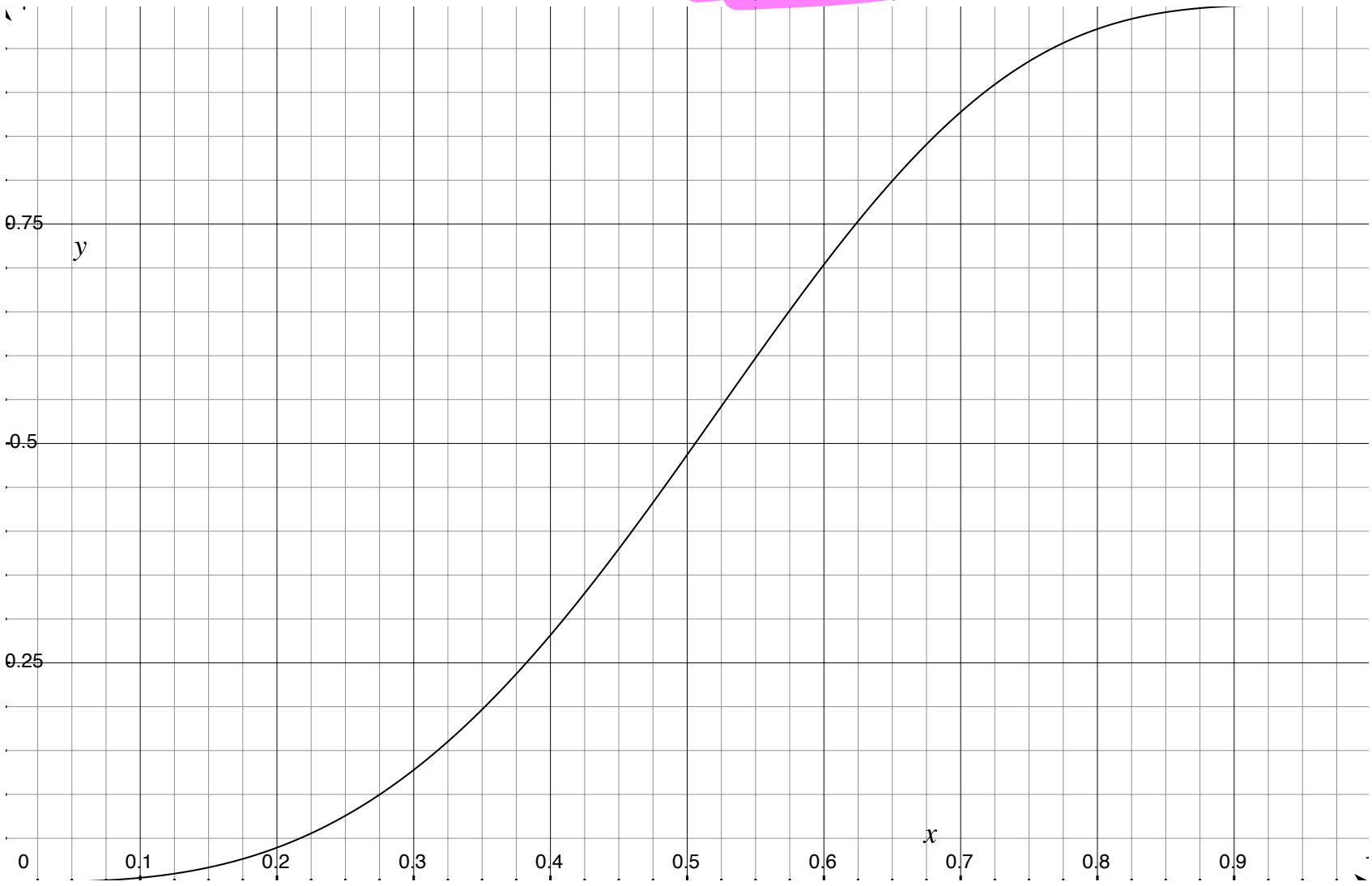
$\Pr[\dots] < 0.01$ $Sim < T-\alpha$



LSH $b = 3$ and $r = 5$

$t = 1.5$

Probability of found collision $= 1 - (1 - s^b)^r$

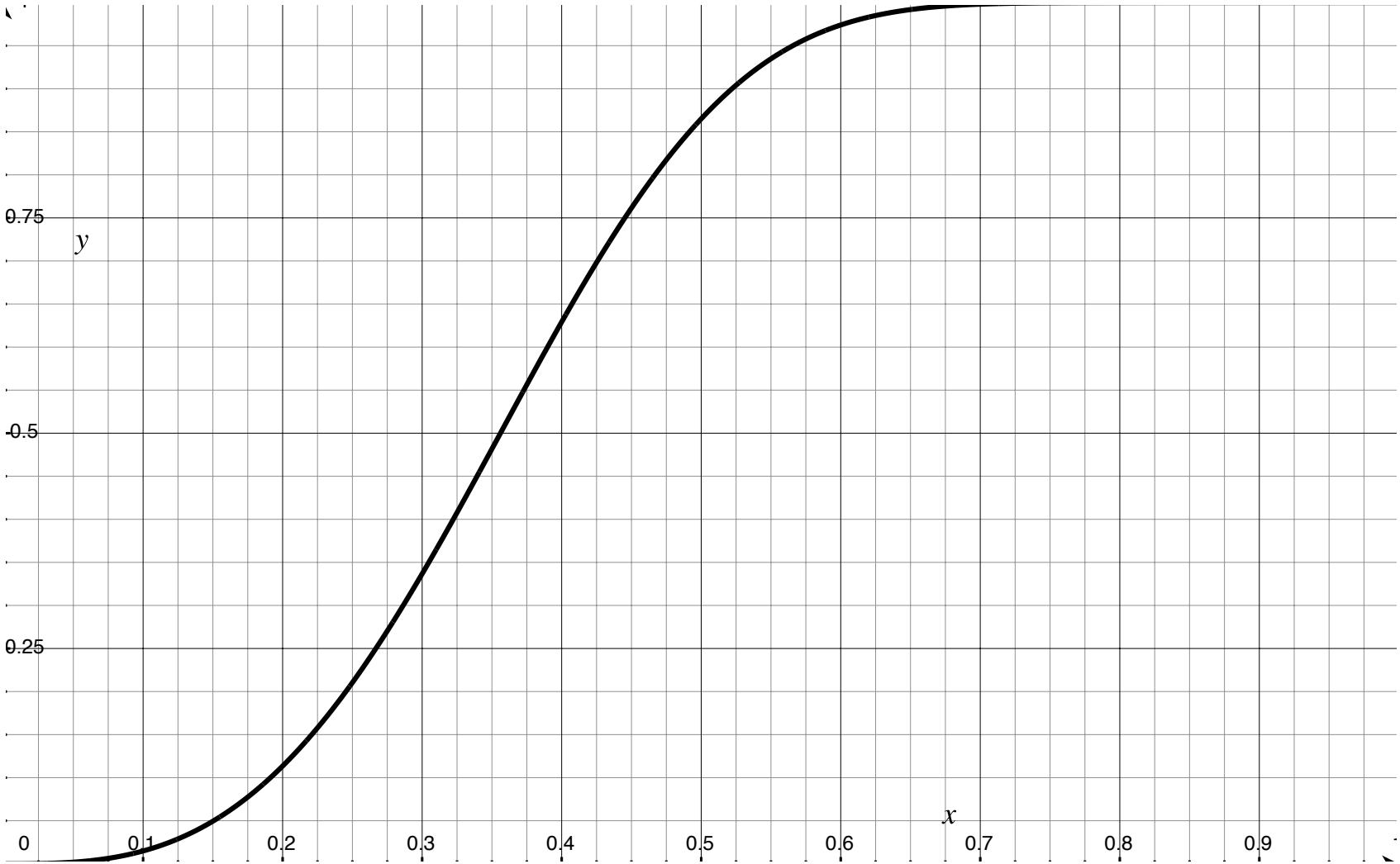


LSH $b = 3$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

LSH $b = 3$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

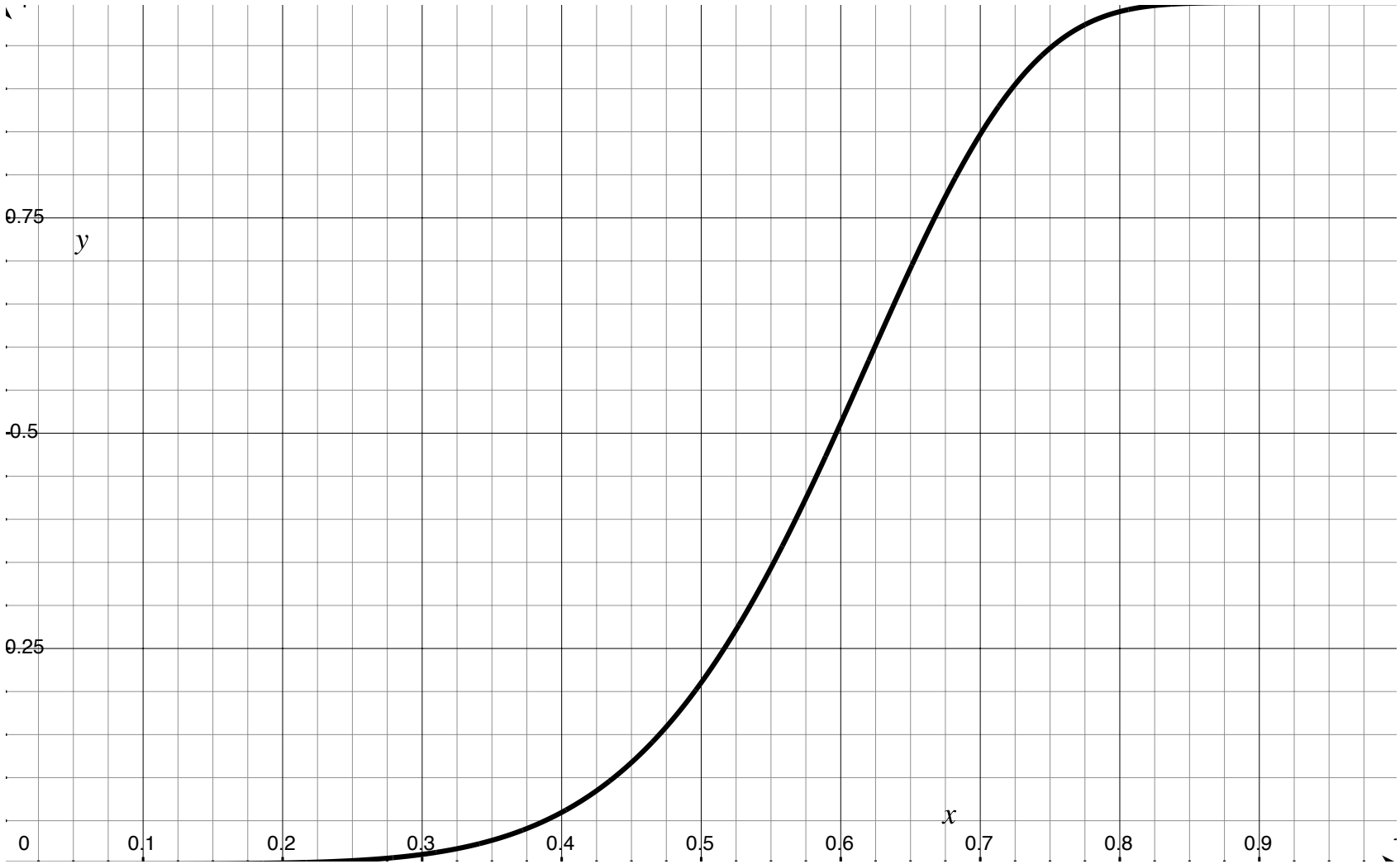


LSH $b = 6$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

LSH $b = 6$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

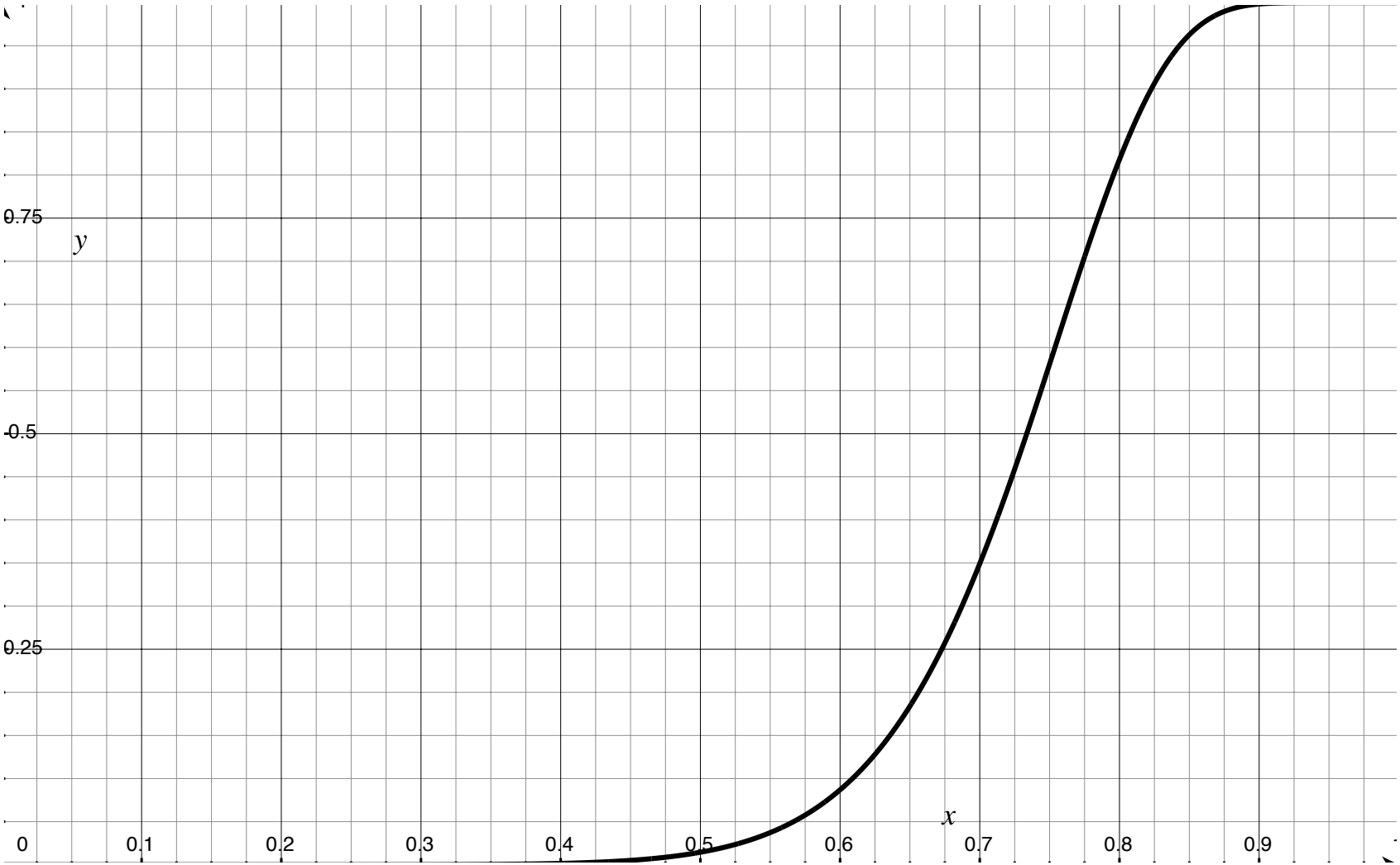


LSH $b = 10$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

LSH $b = 10$ and $r = 15$

Probability of found collision $= 1 - (1 - s^b)^r$

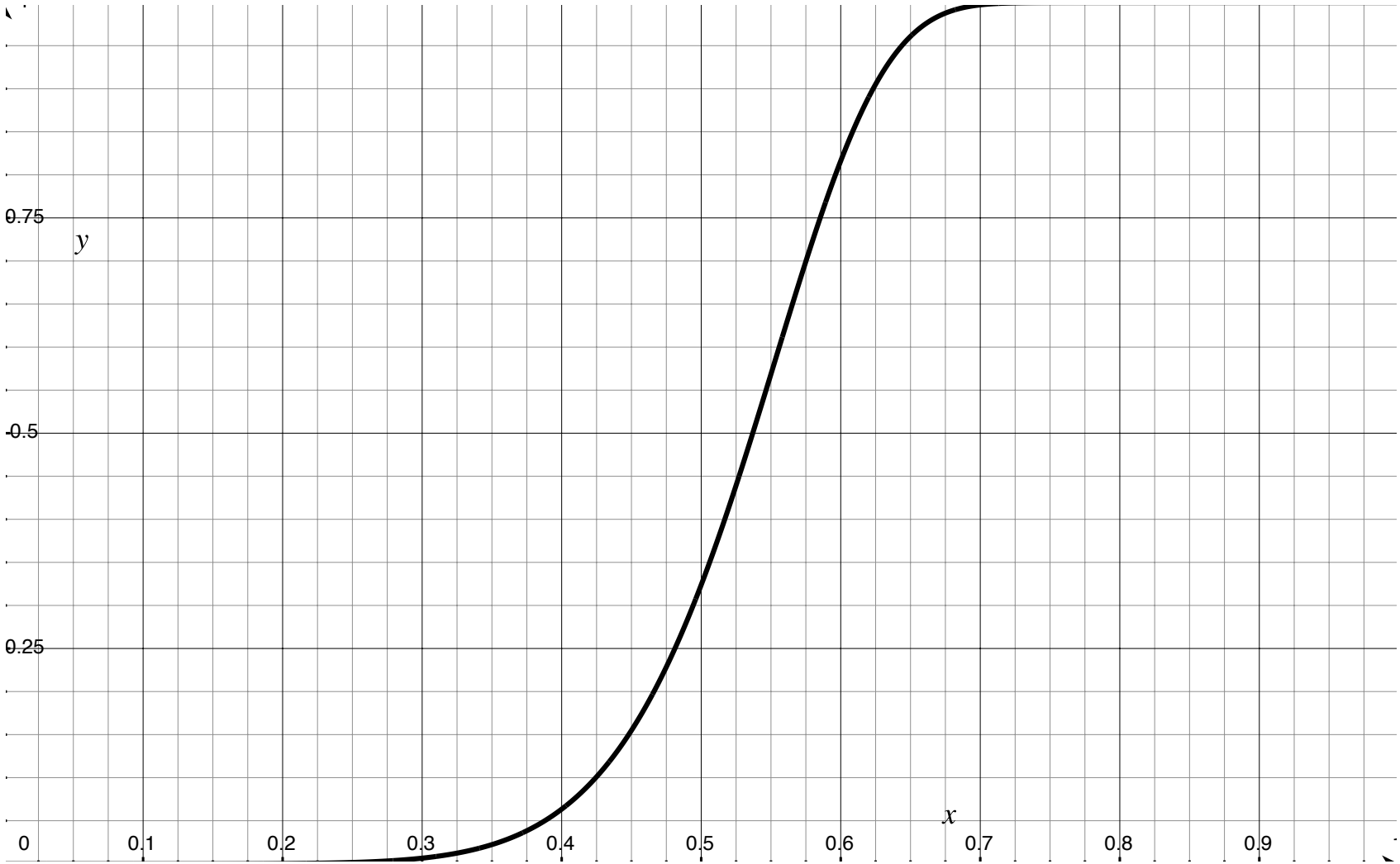


LSH $b = 8$ and $r = 100$

Probability of found collision $= 1 - (1 - s^b)^r$

LSH $b = 8$ and $r = 100$

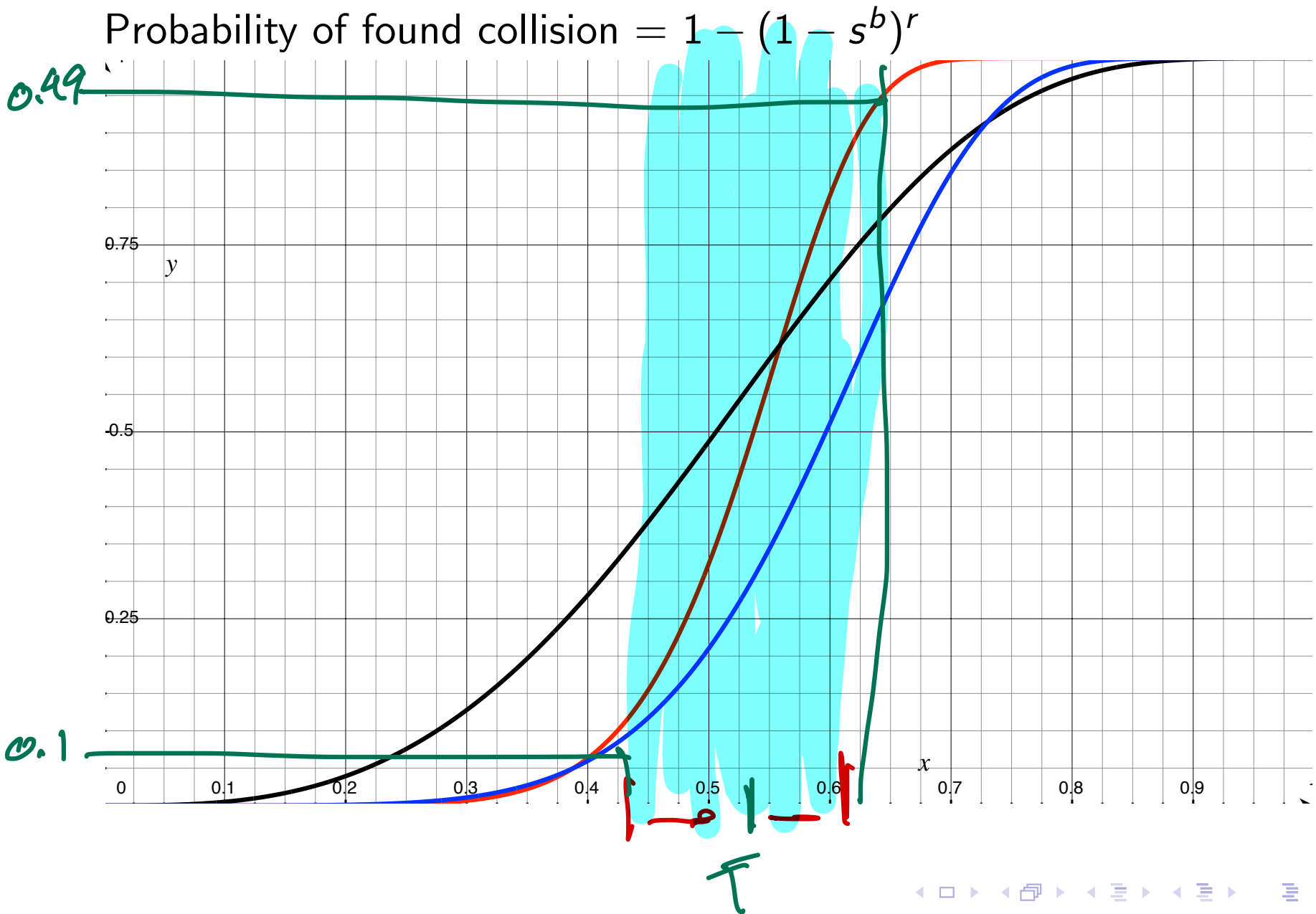
Probability of found collision $= 1 - (1 - s^b)^r$

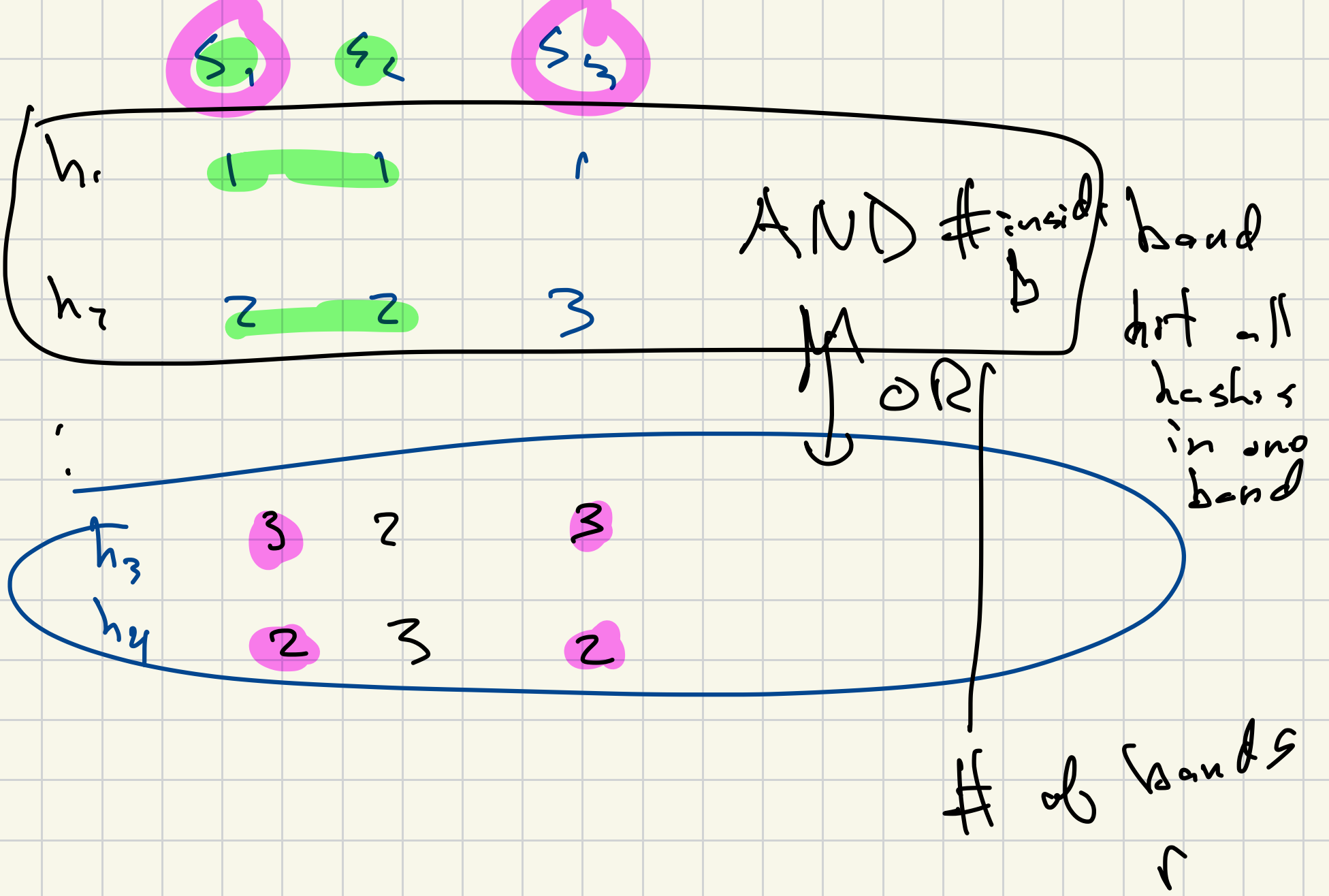


LSH $(b = 3, r = 5)$ & $(b = 6, r = 15)$ & $(b = 8, r = 100)$

Probability of found collision $= 1 - (1 - s^b)^r$

LSH $(b = 3, r = 5)$ & $(b = 6, r = 15)$ & $(b = 8, r = 100)$




S_1
S_2
S_3
h_1
1 1 r
h_2
2 2 3
AND # inside band
OR
dist. all hashes in one band
h_3
3 2 3
h_4
2 3 2
# of bands r

bands of size b | $s = Sim(a,b)$

bands r

1 hash, Prob coll $= s$

b hashes in band, Prob $\overline{\text{coll}} = (1 - s^b)$

r bands, each w/ b hashes

Prob not coll $= (1 - s^b)^r$

r bands, each w/ b hashes

Prob collision $= 1 - (1 - s^b)^r$

If have t hash functions,

how to split into bands

$t = b \cdot r$ Give threshold T

① Plot $1-(1-s^b)^r$ for diff choices

compare to T

② Rule of thumb: steepest slope $1-(1-s^b)^r$

$$b = -\log_T(t) \quad \text{s.t} \quad r = \frac{t}{b}$$

good first guess.