THE UNIVERSITY OF TEXAS AT AUSTIN
Department of Statistics and Data Sciences
College of Natural Sciences

SDS 385: Stat Models for Big Data

Lecture 8: Locality sensitive hashing

Purnamrita Sarkar
Department of Statistics and Data Science
The University of Texas at Austin

https://psarkar.github.io/teaching

Distance measure

We call $d(x, y)$ a distance metric between points x and y in some space, if,

- $d(x, y) \geq 0$
- $d(x, y) = 0 \leftrightarrow x = y$
- *Symmetry:* $d(x, y) = d(y, x)$
- *Triangle inequality:* $d(x, y) \leq d(x, z) + d(z, y)$

Examples

- Euclidian distance $d(x, y) = \sqrt{\|x - y\|^2}$
- L_r norm, $d(x, y) = \left(\sum_i |x_i - y_i|^r\right)^{1/r}$
- $r = 1$: Manhattan distance
- $r \to \infty$: infinity norm
- $r = 2$: Euclidean distance

Examples: Jaccard distance

- Let x, y be sets
- $d(x, y) = 1 - \mathit{Jaccard}(x, y)$
- Can you prove that this is a distance metric?
- Non-negativity is satisfied trivially
- $d(x, y) = 0$ implies $|x \cup y| = |x \cap y|$
- Symmetry is true trivially
- Triangle inequality?

Examples: Jaccard distance

- Remember $J(x, y) = P(h(x) = h(y))$ where h is the min-hash?
- $d(x, y) = P(h(x) \neq h(y))$
- $1(h(x) \neq h(y)) \leq 1(h(x) \neq h(z)) + 1(h(z) \neq h(y))$
- This is because if $h(x) \neq h(y)$, we cannot have $h(x) = h(y) = h(z)$
- So $P(h(x) \neq h(y)) \leq P(h(x) \neq h(z)) + P(h(z) \neq h(y))$

The cosine distance

- Cosine distance between two unit length vectors is the angle between them, which is in $[0, 180]$
- $d(x, y) = \arccos x^T y$
    - Non-negativity: trivial
    - Symmetry: trivial
    - $d(x, y) = 0$ implies they are in the same direction
    - Triangle inequality: argue physically.

Locality sensitive hashing

Let $d_1 < d_2$ be two distances according to some distance measure d. Let $p_1 > p_2$. A family F of functions is said to be (d_1, d_2, p_1, p_2)-sensitive if for every $f \in F$,

- $d(x, y) \leq d_1 \rightarrow P(f(x) = f(y)) \geq p_1$
- $d(x, y) \geq d_2 \rightarrow P(f(x) = f(y)) \leq p_2$

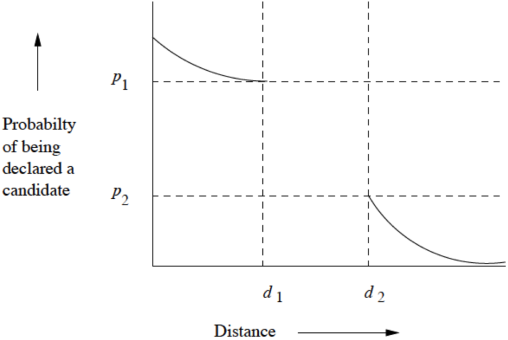
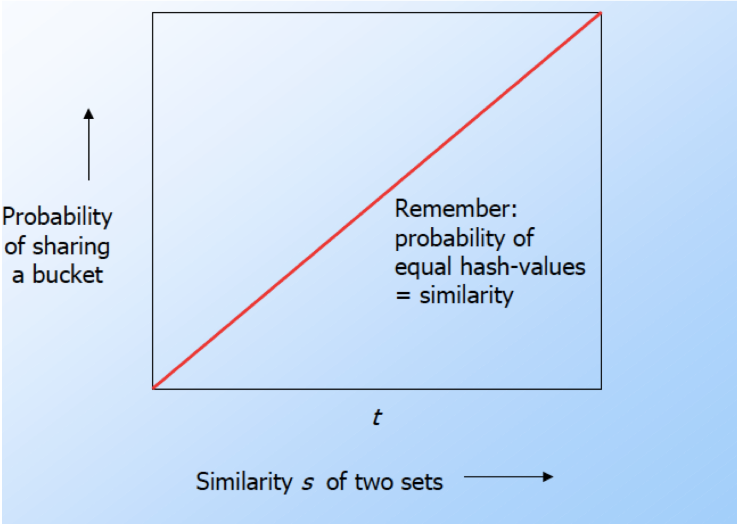


Figure 3.9: Behavior of a (d_1, d_2, p_1, p_2)-sensitive function

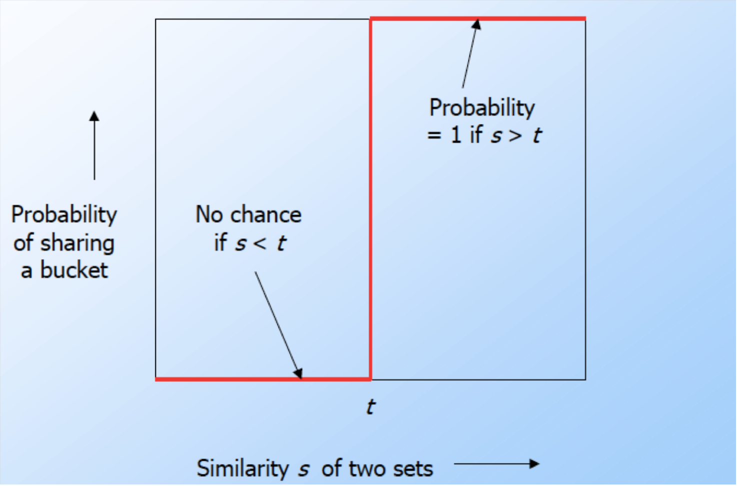
Amplifying the probabilities-AND

- Create new functions by concatenating $\{f_1, \ldots, f_r\}$
- Create a new hash function g and declare $g(x) = g(y)$ iff $f_i(x) = f_i(y) \ \forall i$
- This new family of functions is (d_1, d_2, p_1^r, p_2^r) sensitive
- Note that while each probability has decreased, the ratio (p_1/p_2) has increased exponentially.

What one hash function gives you

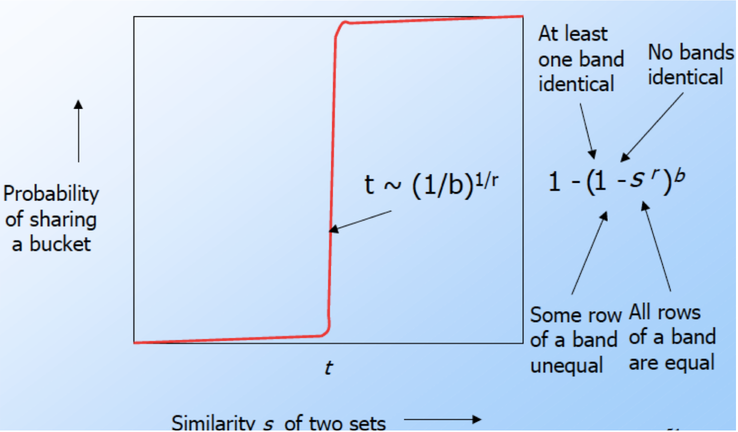


What we want



Probability
= 1 if $s > t$
Probability
of sharing
a bucket
No chance
if $s < t$
t
Similarity s of two sets

What amplification gives you

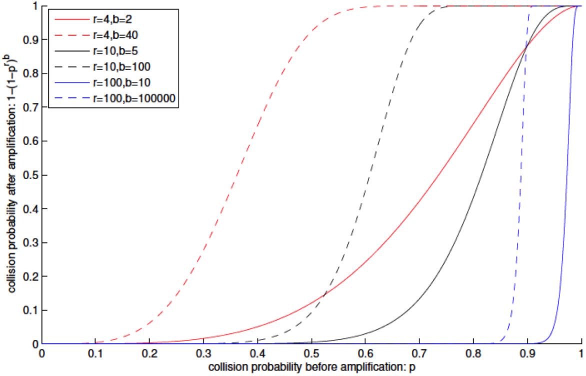


Amplifying the probabilities-OR

- Create new functions by concatenating $\{f_1, \ldots, f_r\}$
- Create a new hash function g and declare $g(x) = g(y)$ iff $f_i(x) = f_i(y)\ \exists i$
- This new family of functions is $(d_1, d_2, 1-(1-p_1)^r, 1-(1-p_2)^r)$ sensitive
- Note that while each probability has decreased, the ratio $(1 - p_1/1 - p_2)$ has decreased exponentially.

Amplifying the probabilities-AND/OR cascades

- First create AND
- Then use a band of the AND's to create OR
- $1 - (1 - p^r)^b$



Example with minhash

- Take the minhash family with the Jaccard distance
- If $d(x, y) < d_1$, then $P(h(x) = h(y)) = J(x, y) \geq 1 - d_1$
- If $d(x, y) > d_2$, then $P(h(x) = h(y)) = J(x, y) \leq 1 - d_2$
- So the minhash family is $(d_1, d_2, 1 - d_1, 1 - d_2)$ sensitive

Hamming distance

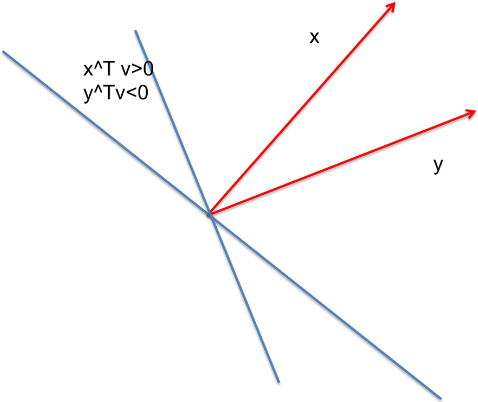
- The number of components in which two vectors (of equal length) differ.
- Easy to see that this is a distance metric.

Hamming distance: hashing scheme

- Take two length d vectors
- Pick index i at random
- $f_i(x) = f_i(y)$ iff $x_i = y_i$
- $P(f_i(x) = f_i(y)) = 1 - d_1/d$
- So this is $(d_1, d_2, 1 - d_1/d, 1 - d_2/d)$ sensitive for any $0 < d_1 < d_2$

Cosine distance

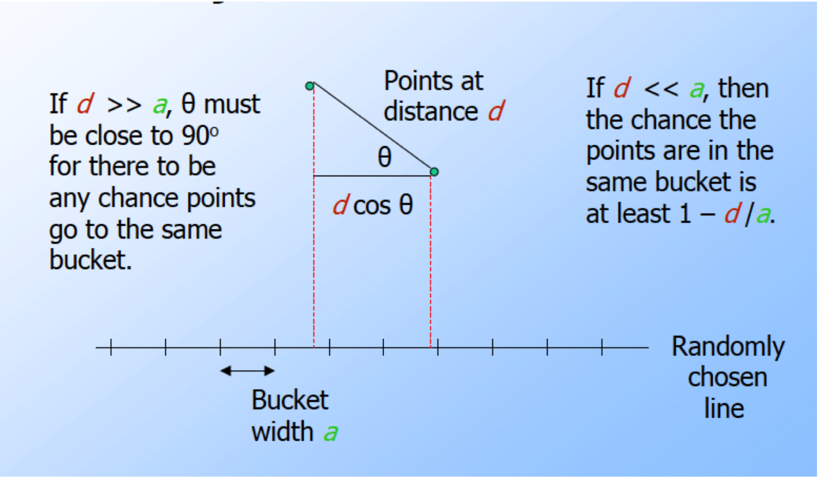
- Pick a unit vector v at random
- $f_v(x) = f_v(y)$ iff $v^T x, v^T y$ have the same sign.
- $P(f_v(x) \neq f_v(y)) = 2P(v^T x \geq 0, v^T y \leq 0) = 2\frac{\theta(x,y)}{2\pi}$



Euclidean distance

- Hash functions corresponding to random lines
- Partition the line into bins of size a
- Hash each point containing its projection onto the line
- Intuition: nearby points are always close; distant points are rarely in same bucket.

Euclidean distance



Euclidean distance

- If $d \ll a$, then $P(h(x) = h(y)) = 1 - d/a$
- If $d > 2a$,
  - We need $cos\theta < 1/2$ to have some nonzero probability of falling in the same bucket
  - So $\theta \in [\pi/3, \pi/2]$
  - So $P(h(x) = h(y)) \leq 1/3$
- So, $d_1 \leq a/2 \rightarrow p_1 \geq 1/2$
- $d_1 \geq 2a \rightarrow p_2 \leq 1/3$
- So $(a/2, a, 1/2, 1/3)$ sensitive LSH family.
- Trouble is, before we had any $d_1 < d_2$ now it seems we need $d_1 \leq d_2/4$

Euclidean distance

- But note that as long as $d_1 < d_2$ the probability of falling in the same bucket in this scheme is always larger than probability of falling in two different buckets.
- So indeed, we have a (d_1, d_2, p_1, p_2) sensitive family for any $d_1 < d_2$ for **some** $p_1 > p_2$.
- Now do the AND-OR constructions

Acknowledgment

- Ullman's lecture notes from "Mining of Massive Datasets".
- Some slides from `http://infolab.stanford.edu/~ullman/mining/2009/similarity3.pdf`
- The S curve plot was taken from Scribe notes of EE381V at UT from Fall 2012