

Locality Sensitive Hashing and Approximate Nearest Neighbors

Lecture 1

Probability

Events, probabilities,
conditional probabilities,
union bound

Random variables,
expected value,
linearity of expectation
Markov's inequality

Algorithms

Approx. 3-SAT,

Big idea: we only want the average

$$\begin{array}{l} \text{Linearity of Expectation} \\ E[X+Y] = E[X] + E[Y] \quad (\text{stochastic variables}) \\ \text{"average sum = sum of averages"} \end{array}$$

let $Y_i = \begin{cases} 1 & \text{if } i\text{th clause satisfied} \\ 0 & \text{otherwise} \end{cases} \quad (i=1, \dots, m)$

$$E[Y_i] = 7/8 \quad (1 \text{ clause case})$$

$$E[\text{\# clauses satisfied}] = E\left[\sum_{i=1}^m Y_i\right] = \sum_{i=1}^m E[Y_i] = 7/8 m$$

let $Y_i = \begin{cases} 1 & \text{if } i\text{th clause satisfied} \\ 0 & \text{otherwise} \end{cases}$

$$Z = \sum_{i=1}^m Y_i = \text{\# clauses satisfied}$$

$$E[Z] = \frac{1}{2} E[Z | x_1 = \text{true}] + \frac{1}{2} E[Z | x_1 = \text{false}]$$

$\Rightarrow$ either $E[Z | x_1 = \text{true}]$ or $E[Z | x_1 = \text{false}]$ is $\geq E[Z]$!

QuickSort,

Random divide and conquer

① pick random 'pivot'

② Divide A into two parts



Expected running time of QuickSort

For each $i, j \in [n]$ w/ $i < j$,

$X_{ij} = \begin{cases} 1 & \text{if } i\text{th smallest element is compared to } j\text{th smallest} \\ 0 & \text{otherwise} \end{cases}$

$\sum_{i,j} X_{ij}$ = total # comparisons made by the algorithm
 $\approx$ total running time

$E[\sum_{i,j} X_{ij}]$ = average running time

$$E[\sum_{i,j} X_{ij}] = \sum_{i,j} E[X_{ij}]$$

now analyze $E[X_{ij}]$ in isolation

$$\begin{aligned} E[X_{ij}] &= \Pr[X_{ij} = 1] \\ &= \Pr[i\text{th element is compared to } j\text{th}] \end{aligned}$$

i th elem is compared w/ j th

$\Leftrightarrow$ i th or j th elem selected before any element in between happens w/ probability $\frac{2}{j-i+1}$

$$E[X_{ij}] = \frac{2}{j-i+1}$$

$$E[\sum_{i,j} X_{ij}] = \sum_{i,j} E[X_{ij}]$$

$$= \sum_{i,j} \frac{2}{j-i+1} = 2 \sum_{i,j} \frac{1}{j-i+1}$$

$$\leq 2n \left(\frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} \right)$$

$$\leq O(n \log n) \quad \text{with harmonic series}$$

Quickselect (HW)

Lecture 2

Probability

Hash Functions,

Universal Hash Functions

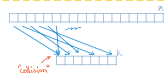
$h: [n] \rightarrow k$ s.t. for all pairs $i, j \in [n]$,
 $\Pr[h(i) = h(j)] \leq \frac{1}{k}$ (much weaker than ideal hash)

e.g. $h(x) = (ax + b) \bmod p \bmod k$

where • p prime, $> k$

• a in $\{1, \dots, p-1\}$ unif. random

• b in $\{0, \dots, p-1\}$ unif. random



Markov's Inequality

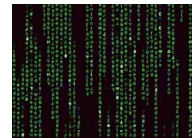
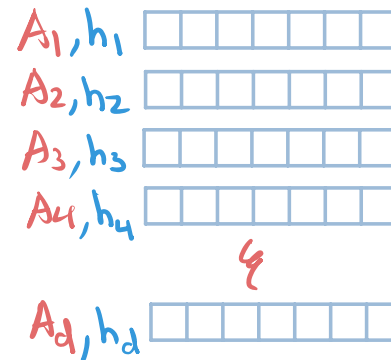
Markov's Inequality (special case)

Let $X \geq 0$ be nonnegative RV

$$\Pr[X \geq 2E[X]] \leq \frac{1}{2}$$

Algorithms

Frequency Estimation, Heavy Hitters (in streams)



estimate f_i w/
 $\min_{j=1, \dots, d} A_j[h_j(i)]$

Lecture 3 (last Tuesday)

Probability

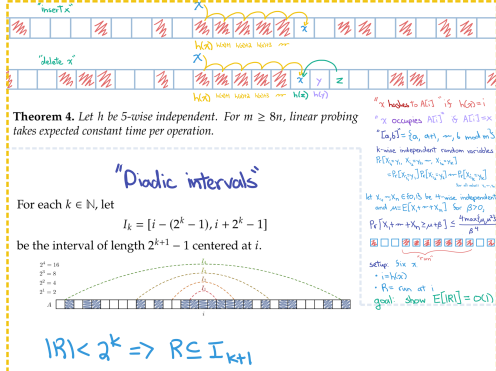
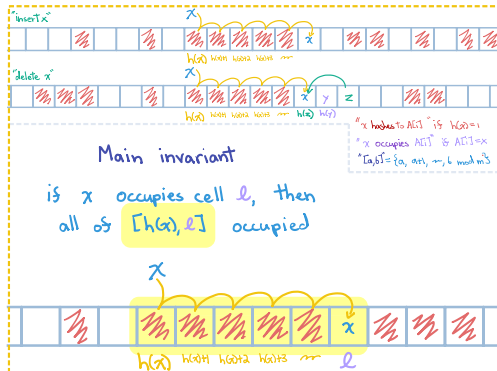
k-wise independence,
higher order moments
(generalizing universal hash
functions and Markov's inequality)

Algorithms

Hash tables w/
chaining +
linear probing

k-wise independent hash functions
 $h: [U] \rightarrow [m]$
s.t. for any k keys $x_1, x_2, \dots, x_k \in [U]$,
 k values $v_1, v_2, \dots, v_k \in [m]$,
 $P[h(x_1)=v_1, h(x_2)=v_2, \dots, h(x_k)=v_k] = \frac{1}{m^k}$
eg. $h(x) = a_0 + a_1x + a_2x^2 + \dots + a_{k-1}x^{k-1} \pmod p$
where $p \geq U$ prime,
 $a_0, a_1, \dots, a_{k-1} \in [p]$ indep, uniformly random

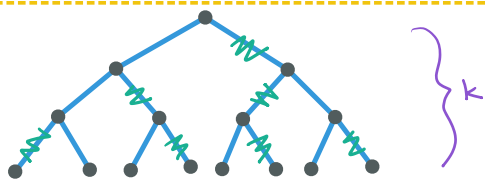
Important lemma
k-wise independent random variables
 $P[X_1=y_1, X_2=y_2, \dots, X_k=y_k]$
 $= P[X_1=y_1] P[X_2=y_2] \dots P[X_k=y_k]$
for all values $y_1, \dots, y_k$
let $X_1, \dots, X_n \in \{0,1\}$ be 4-wise independent
and $\mu = E[X_1 + \dots + X_n]$. then for $\beta > 0$,
 $P[X_1 + \dots + X_n \geq \mu + \beta] \leq \frac{4 \max\{\mu, \mu^3\}}{\beta^4}$



Lecture 4

Probability

Galton-Watson processes



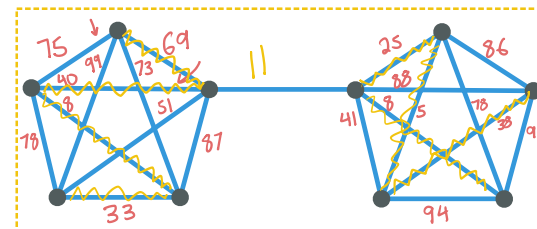
let T = complete binary tree of height k ,
w/ every edge deleted w/ prob $\leq 1/2$

Fact: $P[\text{some leaf connected to root}] \geq \frac{1}{k}$

"Galton-Watson process"

Algorithms

Random Contractions for min cut



Randomized contractions

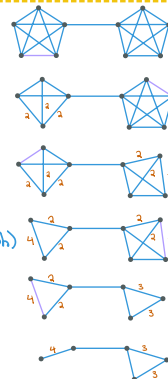
Input: $G=(V,E)$, weights $c \in \mathbb{R}_{>0}^E$
(unweighted)

① while $|E| > 1$

 Ⓐ sample $e \in E$ in proportion to c

 Ⓑ "contract" e

② return set of edges (in original graph)
 contracted to remaining edge



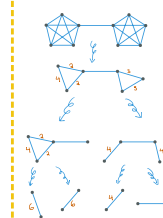
Amplification by branching

Instead of restarting at the beginning, restart partway through

High-level algorithm

① randomly contract until prob.
of ruining the min-cut is $\sim 1/2$

② Branch:
make two copies of current graph
recurse on both copies



Lecture 5

Probability

Multiplicative Chernoff Bound

Multiplicative Chernoff bound

let $X_1, \dots, X_n \in [0, 1]$ be independent

$$\mu = E[X_1 + \dots + X_n]$$

$$\epsilon \in (0, 1)$$

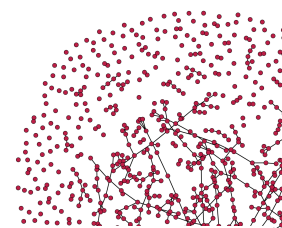
- $P[X_1 + \dots + X_n \geq (1 + \epsilon)\mu] \leq e^{-\epsilon^2 \mu / 3}$
- $P[X_1 + \dots + X_n \leq (1 - \epsilon)\mu] \leq e^{-\epsilon^2 \mu / 2}$

Algorithms (/ results)

Random graphs & the giant component

Erdős-Rényi graphs $G(n, p)$

- n vertices
- each edge sampled independently w/ probability p



Theorem 5.3. Consider a random graph $G \sim \mathcal{G}(n, p)$ for $p = c/n$, where c is a constant.

- If $c > 1$, then with high probability, there is exactly one connected component of G with $\Omega(n)$ vertices, and all other components have size $\leq O(\log n)$.
- For $c < 1$, then with high probability, all connected components of G has size $< O(\log n)$.

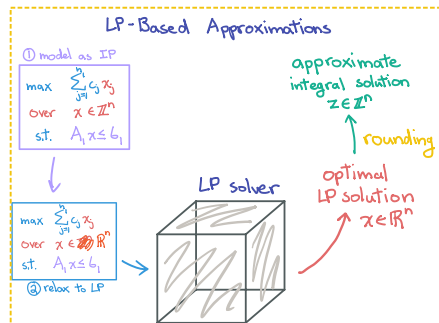
Lecture 6

Probability (& techniques)

Approximations via linear programming & randomized rounding

Linear programs (LPs)
Similar to IPs, except no integrality constraints

min/max $\sum_{j=1}^n c_j x_j$ linear objective
over $x \in \mathbb{R}^n$ linear integral variables
st. $A_1 x \leq b_1$ linear constraints
 $A_2 x = b_2$
 $A_3 x \geq b_3$



Continuous optimization vs Discrete optimization

randomization

Algorithms

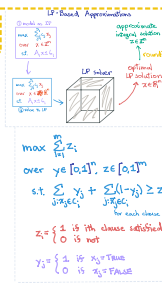
Max Sat Set Cover Covering Integer Programs

For each variable x_j , independently, randomly set $x_j = 1$ with probability p_j , and $x_j = 0$ otherwise.

$$E[\# \text{ sat. clauses}] \geq (1 - 1/e) \text{OPT}$$

k	oblivious	LP	average
1	1/2	$z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
2	3/4	$(3/4)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
3	7/8	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
4	15/16	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
5	31/32	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$
...	...	...	...
k	$1 - 1/2^k$	$(1 - 1/e)z_i \geq (3/4)z_i$	$\geq (3/4)z_i$

Can we get $\frac{3}{4}$ -APX?



Rounding LP solution
 $x \in [0, 1]^n \rightsquigarrow F \subseteq \{S_1, \dots, S_n\}$

For each set S_j , independently, take S_j with probability x_j .

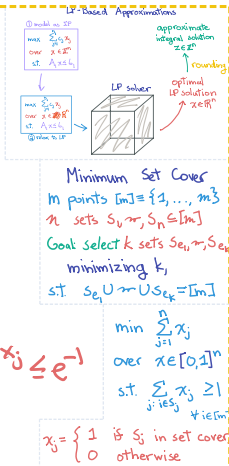
$$E[|F|] = \sum x_j \leq \text{OPT}$$

Fix a point i .

$$E[\# \text{ sets in } F \text{ covering } i] = \sum_{j \in S_i} x_j \geq z_i$$

$$P[F \text{ not cover } i] = \prod_{j \in S_i} (1 - x_j) \leq e^{-\sum_{j \in S_i} x_j} \leq e^{-z_i}$$

$$P[F \text{ cover } i] \geq 1 - e^{-z_i}$$



Lecture 7

Probability
(and techniques)

Algorithms
(models and applications)

Caching

Buy-or-rent

Lecture 8

Probability (& techniques)

Variance

Variance Reduction

Median Trick

Algorithms

Distinct Elements

F_2 estimation

Lecture 9

Probability

Gaussian random variables

Gaussian random variables

$$x \sim \mathcal{N}(\mu, \sigma^2)$$

mean variance

Def: the nicest, most pleasant, convenient random variable ever

Fact let $x \sim \mathcal{N}(\mu, \sigma^2)$, $d \in \mathbb{R}$.

$$\text{then } x+d \sim \mathcal{N}(\mu+d, \sigma^2)$$

$$dx \sim \mathcal{N}(d\mu, d^2\sigma^2)$$

Fact let $x_1 \sim \mathcal{N}(\mu_1, \sigma_1^2)$, $x_2 \sim \mathcal{N}(\mu_2, \sigma_2^2)$

$$\text{then } x_1 + x_2 \sim \mathcal{N}(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$$

Fact let $x \sim \mathcal{N}(0, 1)$, $t < \frac{1}{2}$

$$\text{then } E[e^{tx^2}] = \frac{1}{\sqrt{1-2t}}$$

Algorithms

Dimensionality reduction

Bag-of-words

literally any sentence is fine

any ↑ is literally sentence English
five encodes text in high dimensions

Lemma 8.8. Let $A \sim \mathcal{N}^{k \times d}$ be a random matrix where each coordinate A_{ij} is an independently drawn sample from $\mathcal{N}$. Let $\epsilon > 0$ be sufficiently small. For any vector x ,

$$\mathbb{P}\left[(1-\epsilon)\|x\|^2 \leq \frac{1}{k}\|Ax\|^2 \leq (1+\epsilon)\|x\|^2\right] \geq 1 - 2e^{-k/8}$$

$$k \left\{ \left[\frac{1}{k} \|Ax\|^2 \right] \leftarrow \alpha_i \right.$$

$$E[\|Ax\|^2] = \sum_{i=1}^k E[(Ax)_i^2] = k\|x\|^2$$

$$E[(\sum_{i=1}^k A_{ij} x_j)^2] = \|x\|^2$$

$$A_{ij} x_j \sim \mathcal{N}(0, x_j^2)$$

$$\sum_{j=1}^n A_{ij} x_j \sim \mathcal{N}(0, \|x\|^2)$$

$$(Ax) \sim \mathcal{N}^k(0, \|x\|^2)$$

$$\Pr[\|Ax\|^2 \geq k\|x\|^2 \cdot (1+\epsilon)] \leq e^{-\epsilon^2 k/8}$$

Dimensionality Reduction - high level

$O(\log n)$ dimensions suffice

where $n = \# \text{ points}$

$$\mathbb{R}^{\text{very large}} \xrightarrow{f(x)} \mathbb{R}^{O(\log n)}$$

One can map n points into $\mathbb{R}^{O(\log n/\epsilon^2)}$ while preserving all distances up to $(1 \pm \epsilon)$ -APX

Lecture 10

Probability

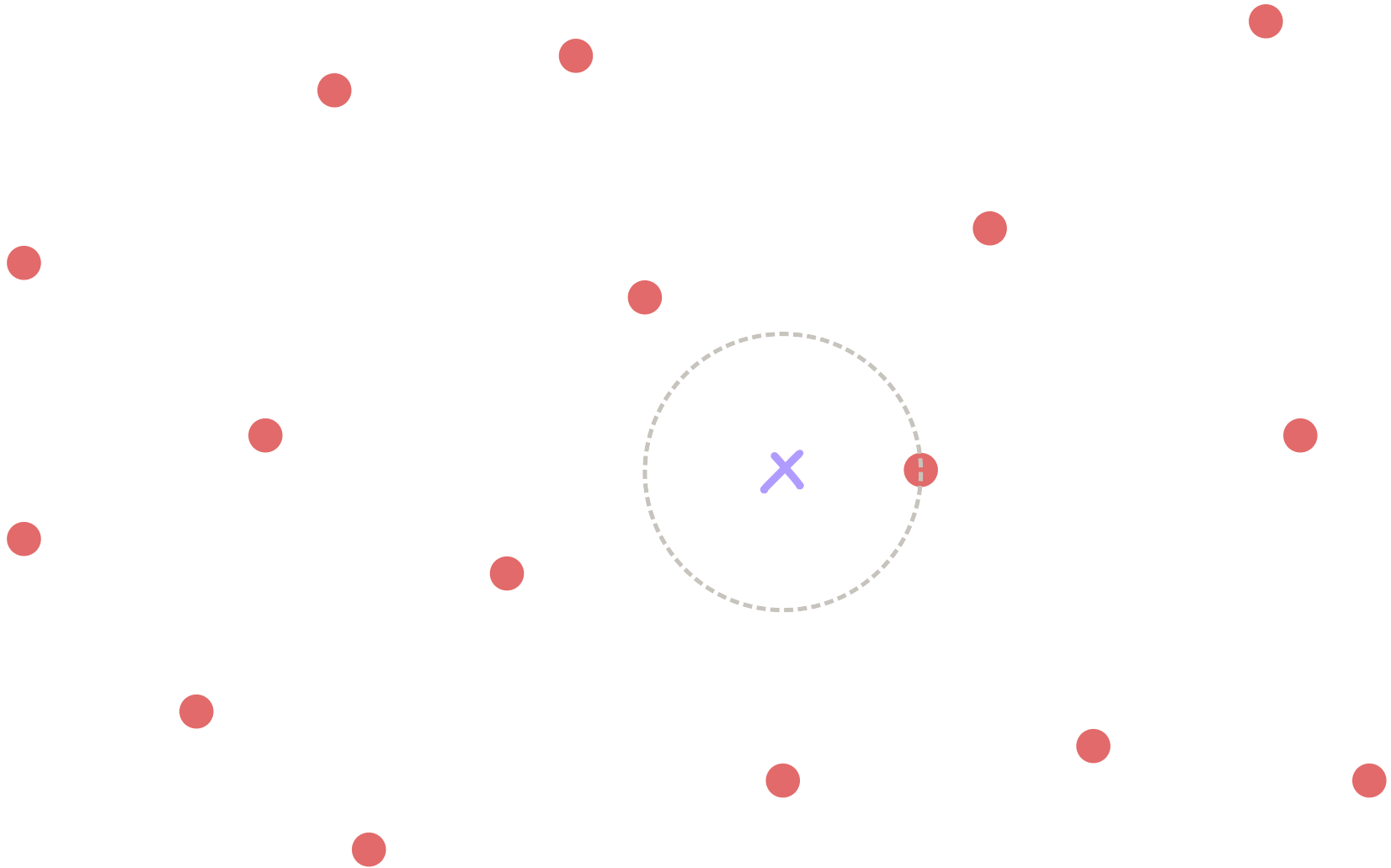
Locality-sensitive
hashing

Algorithms

Approximate
nearest neighbors

Nearest neighbor search

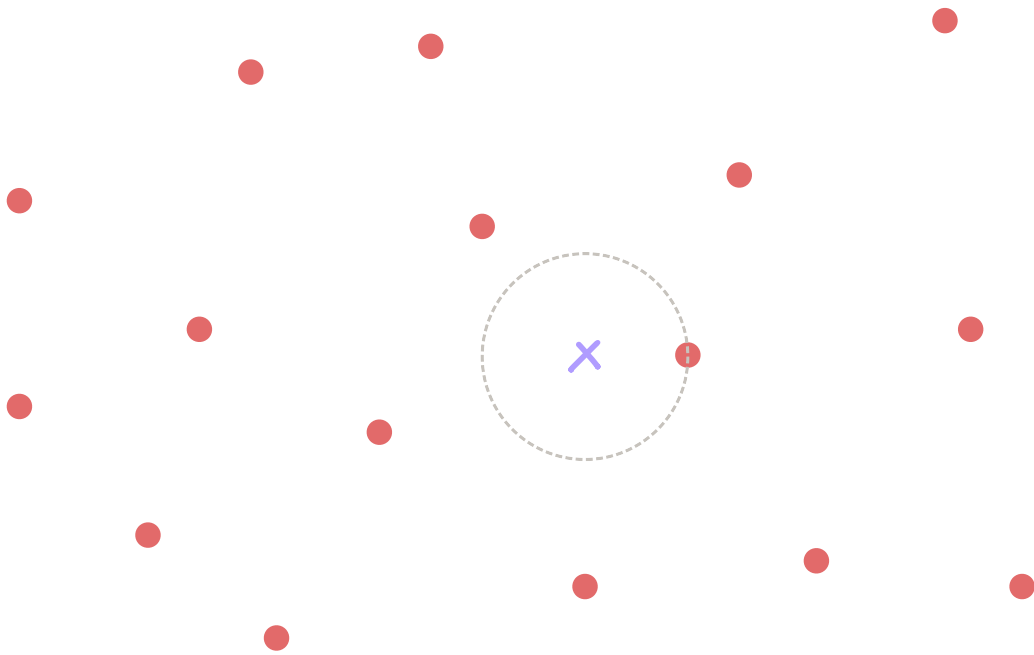
Input: n points $P \subset \mathbb{R}^d$



Query: given $q \in \mathbb{R}^d$, find closest point in P to q

Nearest neighbor search

Input: n points $P \subset \mathbb{R}^d$

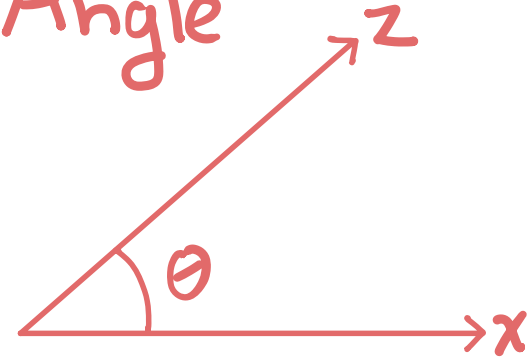


Types of distances

- Euclidean distance

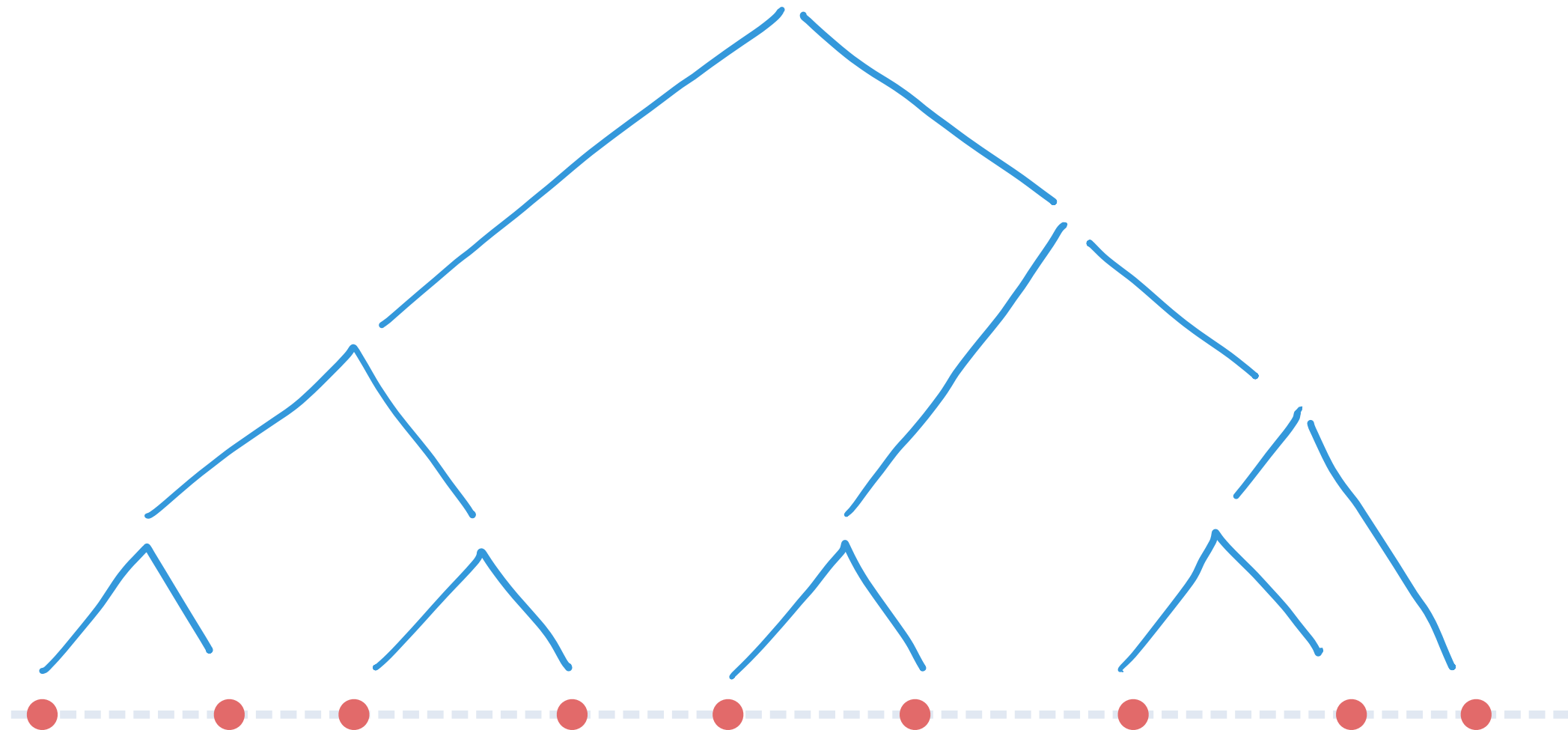
$$\|x - z\|_2 = \sqrt{\sum_i (x_i - z_i)^2}$$

- Angle

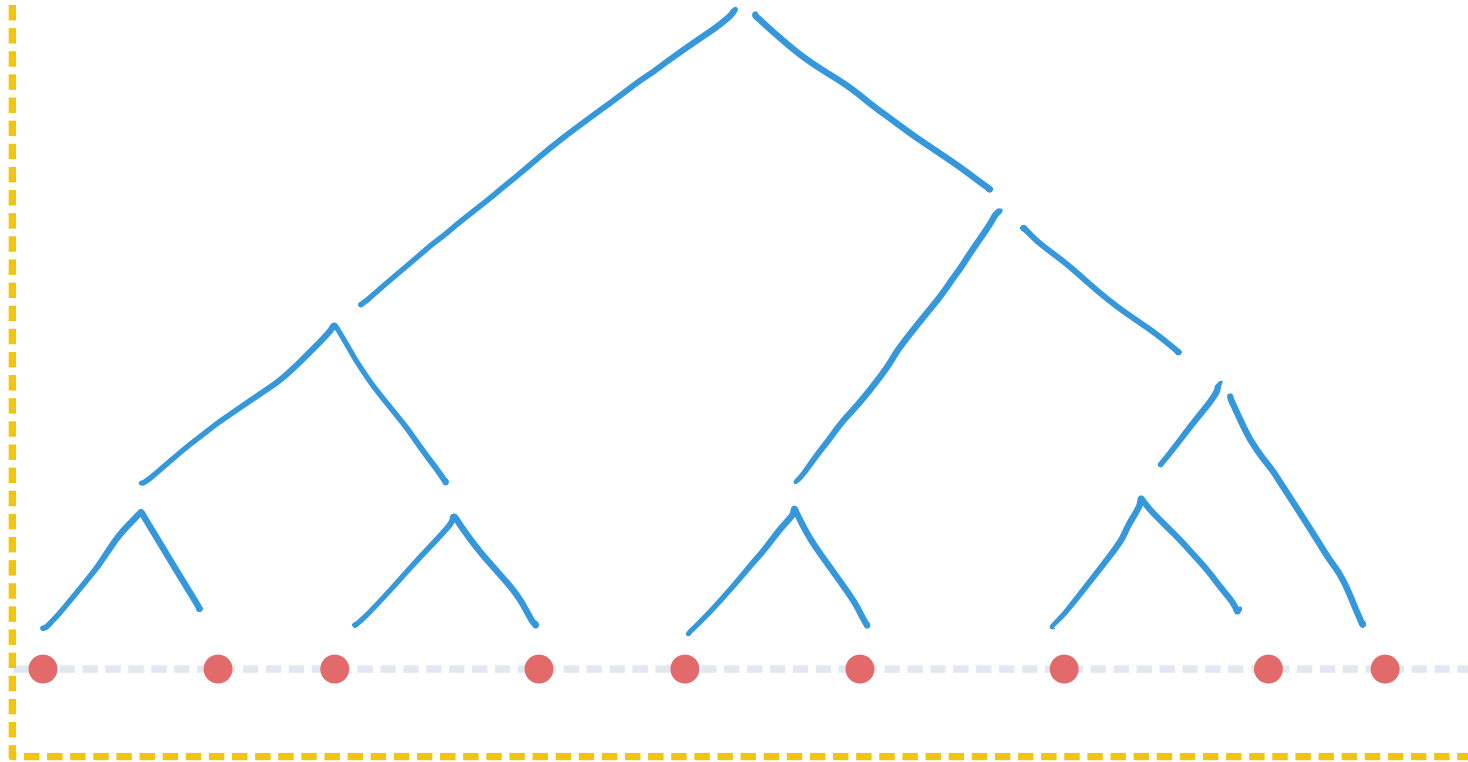


Query: given $q \in \mathbb{R}^d$, find closest point in P to q

Nearest neighbor search in 1D



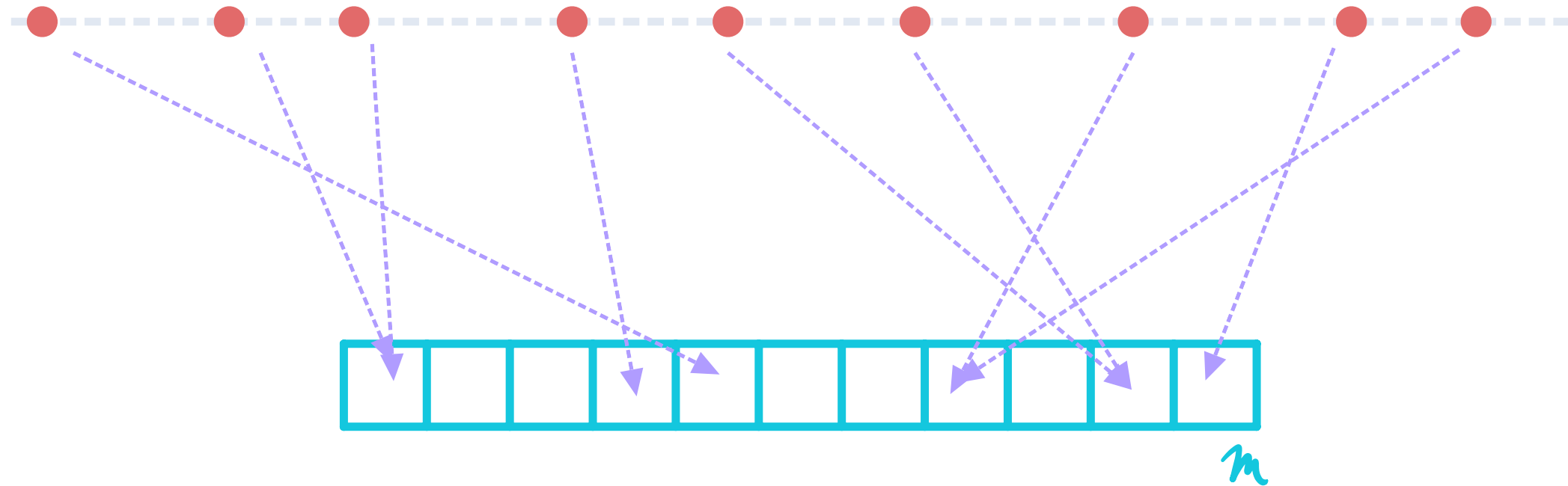
Nearest neighbor search in 1D



generalizes to k dimensions w/
kd-trees, quadtrees; $O(\log^k n)$ time

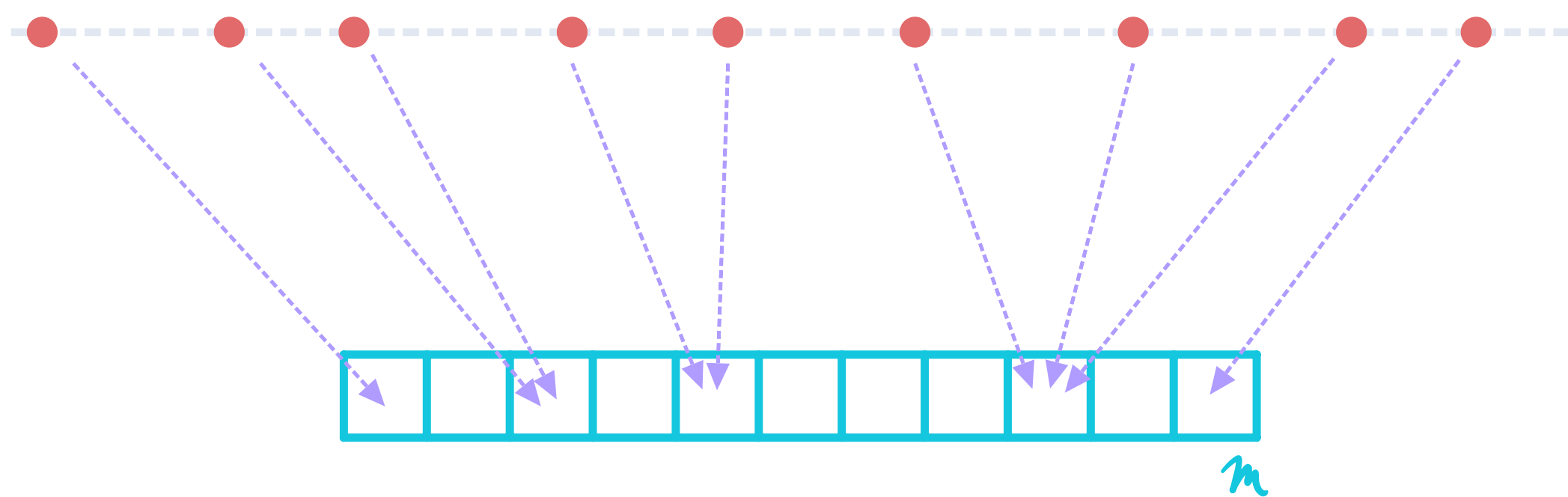
"Locality sensitive hashing"

idea: hash functions that m

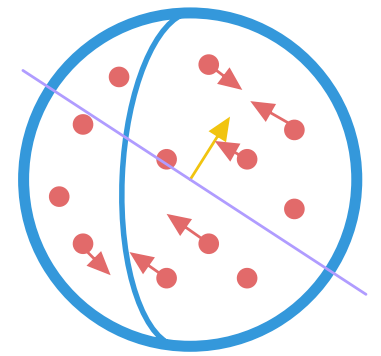
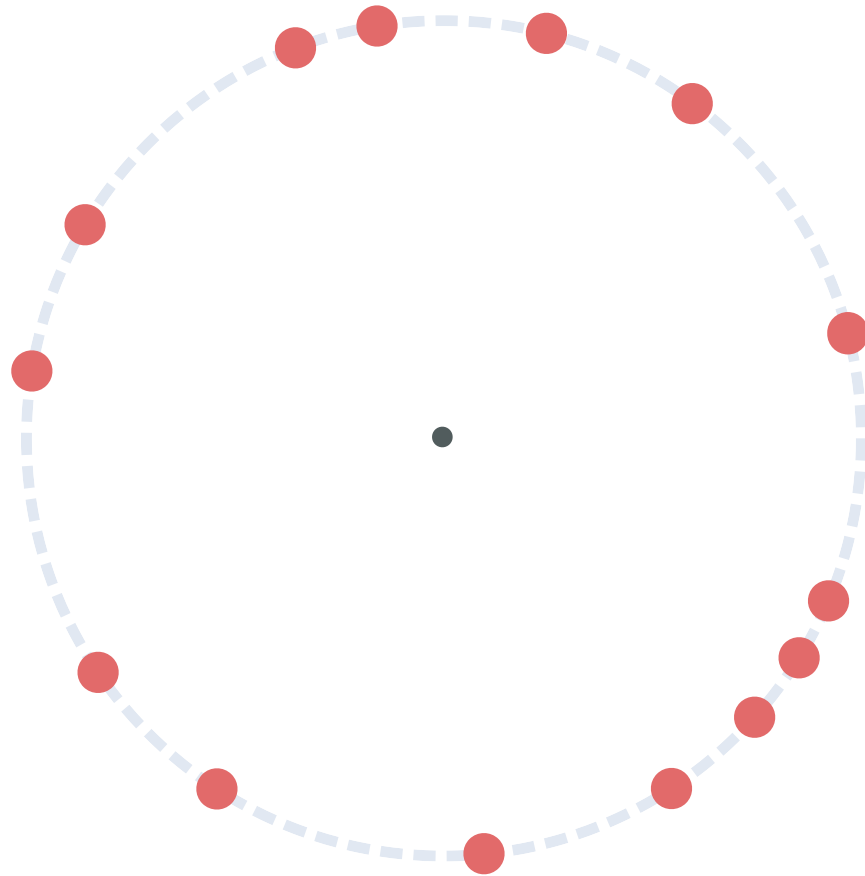


"Locality sensitive hashing"

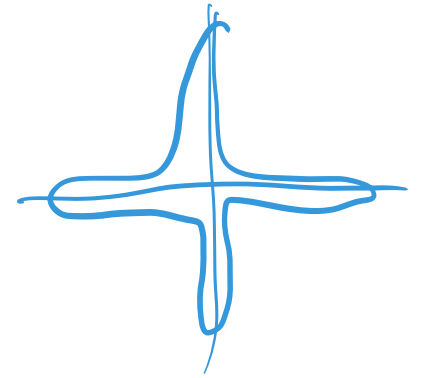
idea: hash functions that increase collisions for nearby points



Angular LSH

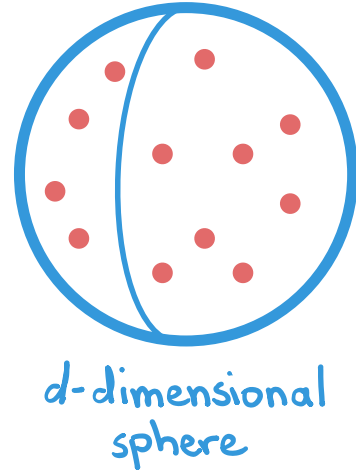
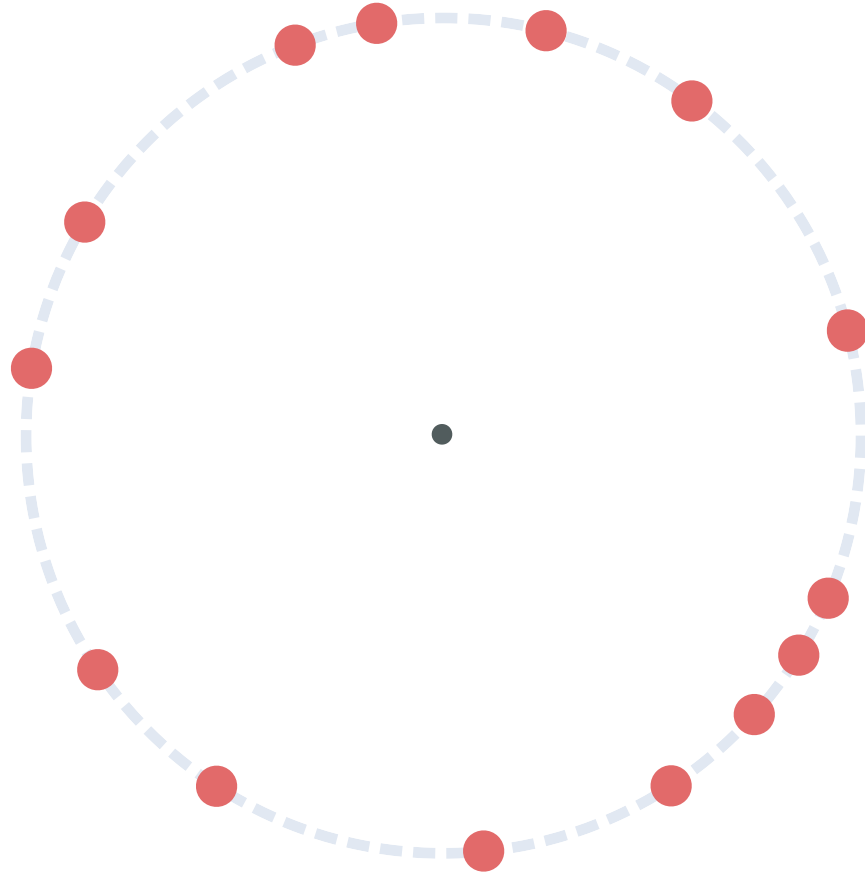


d -dimensional
sphere



Query for point w/ smallest angle

Angular LSH - parametrized and approximate



Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

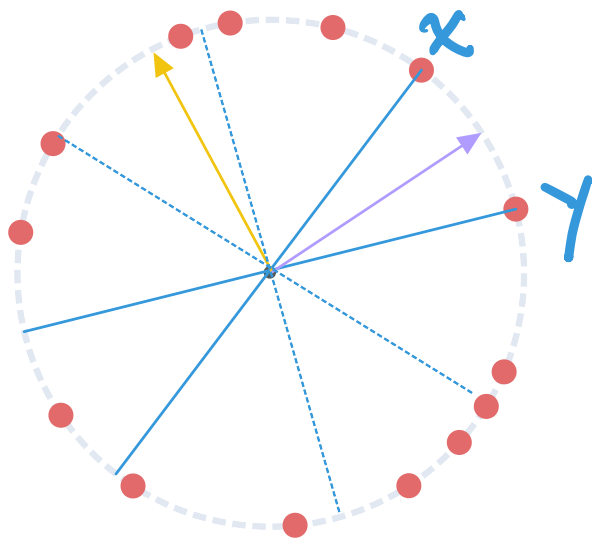
Lemma let $\bullet x, y \in S^{d-1}$

$\bullet g \sim \mathcal{N}^d$

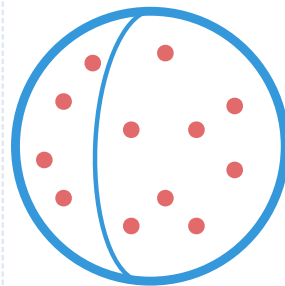
then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

proof



Angular LSH
parametrized and approximate
Fix $\theta \in [0, \pi]$. given q , either:
(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$
(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$



Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

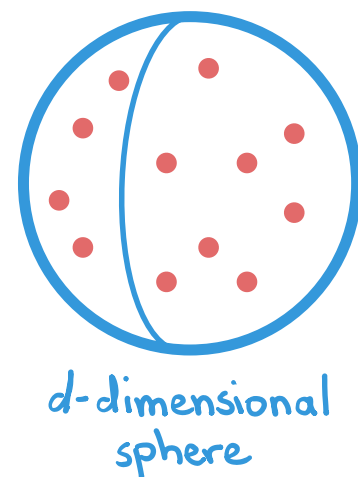
Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim \mathcal{N}^d$

then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \quad \text{w/ } g_i \sim \mathcal{N}^d$$



Algo: group P by hash code $h(x)$

to query y , scan all points $x \in P$ w/ $h(x) = h(y)$

Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\Delta(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\Delta(p, q) \leq \theta$

Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim \mathcal{N}^d$

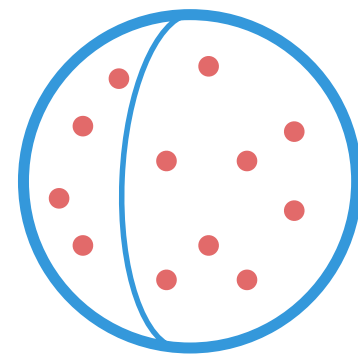
then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\Delta(x, y)}{\pi}$$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \quad \text{w/ } g_i \sim \mathcal{N}^d$$

$$\Delta(x, y) \leq \theta:$$



d-dimensional
sphere

Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

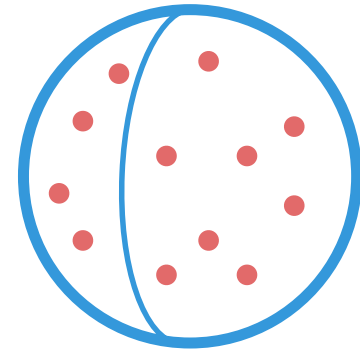
Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim \mathcal{N}^d$

then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \quad \text{w/ } g_i \sim \mathcal{N}^d$$



d-dimensional
sphere

$$\angle(x, y) > 2\theta$$

Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

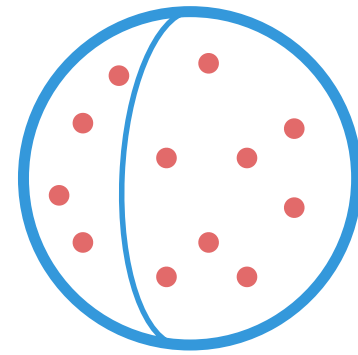
Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim \mathcal{N}^d$

then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \quad \text{w/ } g_i \sim \mathcal{N}^d$$



d-dimensional sphere

$$\angle(x, y) > 2\theta$$

$$P[h(x) = h(y)] = \left(1 - \frac{\angle(x, y)}{\pi}\right)^k \leq e^{-\frac{k \angle(x, y)}{\pi}} \leq \frac{1}{n}$$

$$E[\# \text{ false positives}] =$$

Fix $\theta \in [0, \pi]$. given q , either:

(a) output $p \in P$ w/ $\angle(p, q) \leq 4\theta$

(b) declare no $p \in P$ w/ $\angle(p, q) \leq \theta$

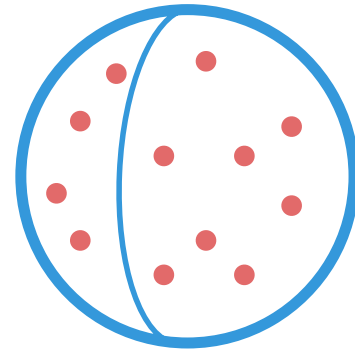
Lemma let $\bullet x, y \in S^{d-1}$
 $\bullet g \sim \mathcal{N}^d$

then

$$P[\text{sign}(\langle g, x \rangle) \neq \text{sign}(\langle g, y \rangle)] = \frac{\angle(x, y)}{\pi}$$

define $h: S^{d-1} \rightarrow \{-1, 1\}^k$ by

$$h_i(x) = \text{sign}(\langle g_i, x \rangle) \quad \text{w/ } g_i \sim \mathcal{N}^d$$



$$k = \pi \log(n) / 2\theta$$

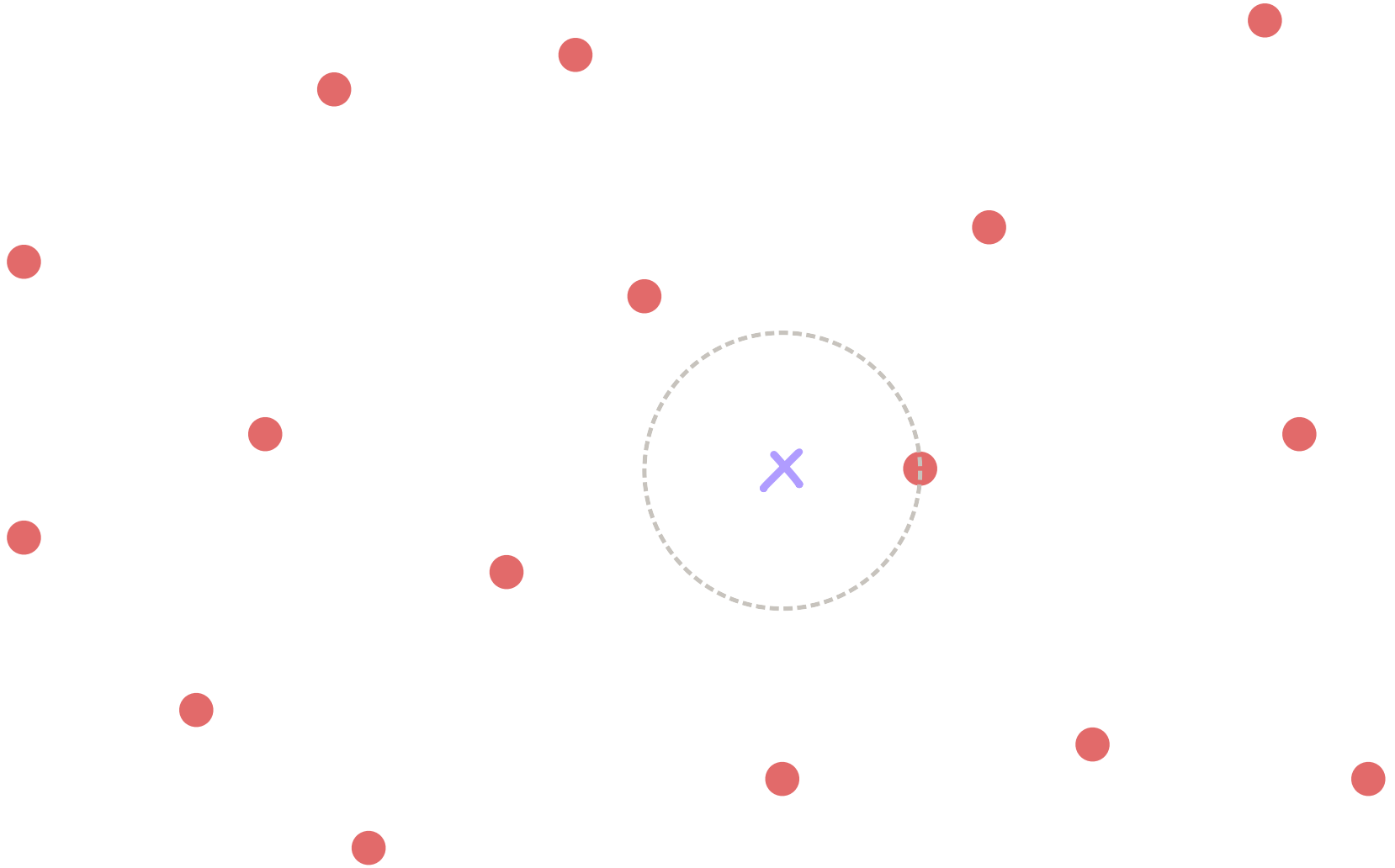
d-dimensional
sphere

when querying y :

- if there is $x \in P$ w/ $\angle(x, y) \leq \theta$, then it collides
w/ probability $\gtrsim 1/\sqrt{n}$

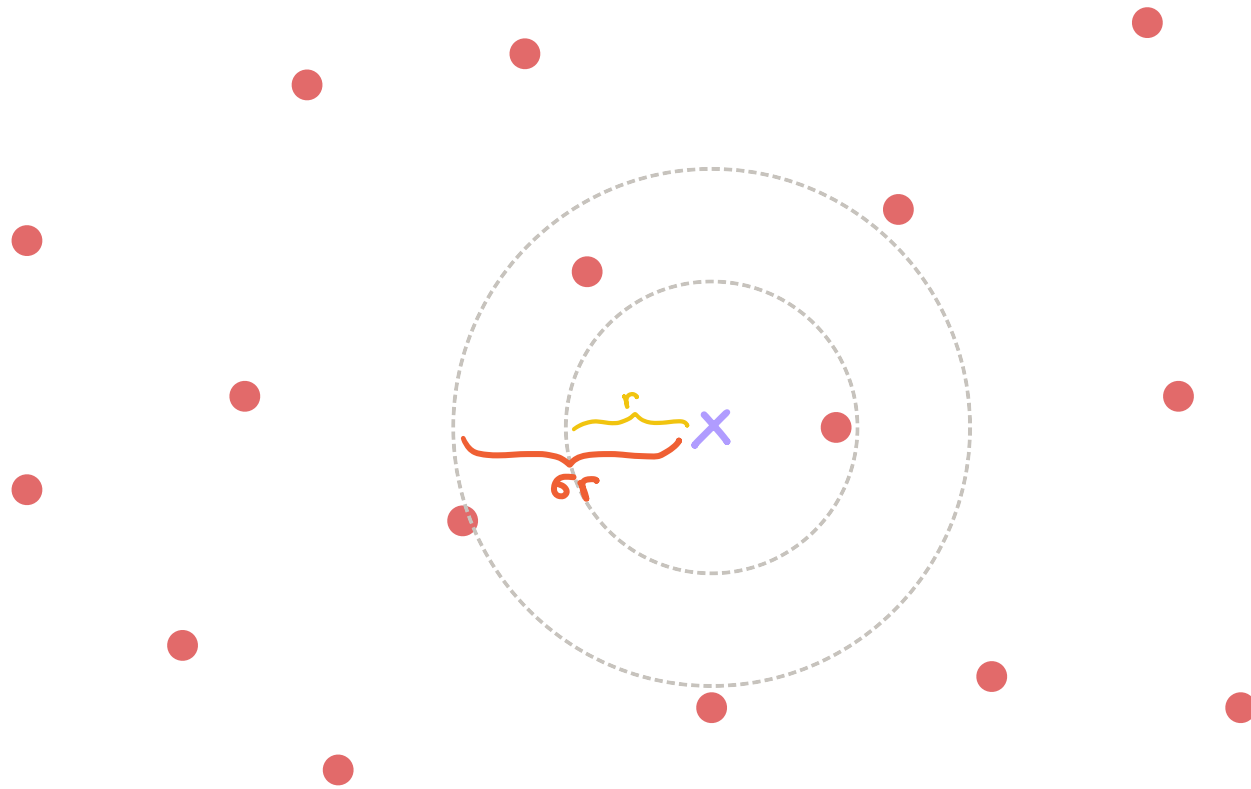
- expect to collide w/ at most one point $x \in P$
w/ $\angle(x, y) > 2\theta$

Euclidean LSH



Query: given $q \in \mathbb{R}^d$, find closest point in P to q

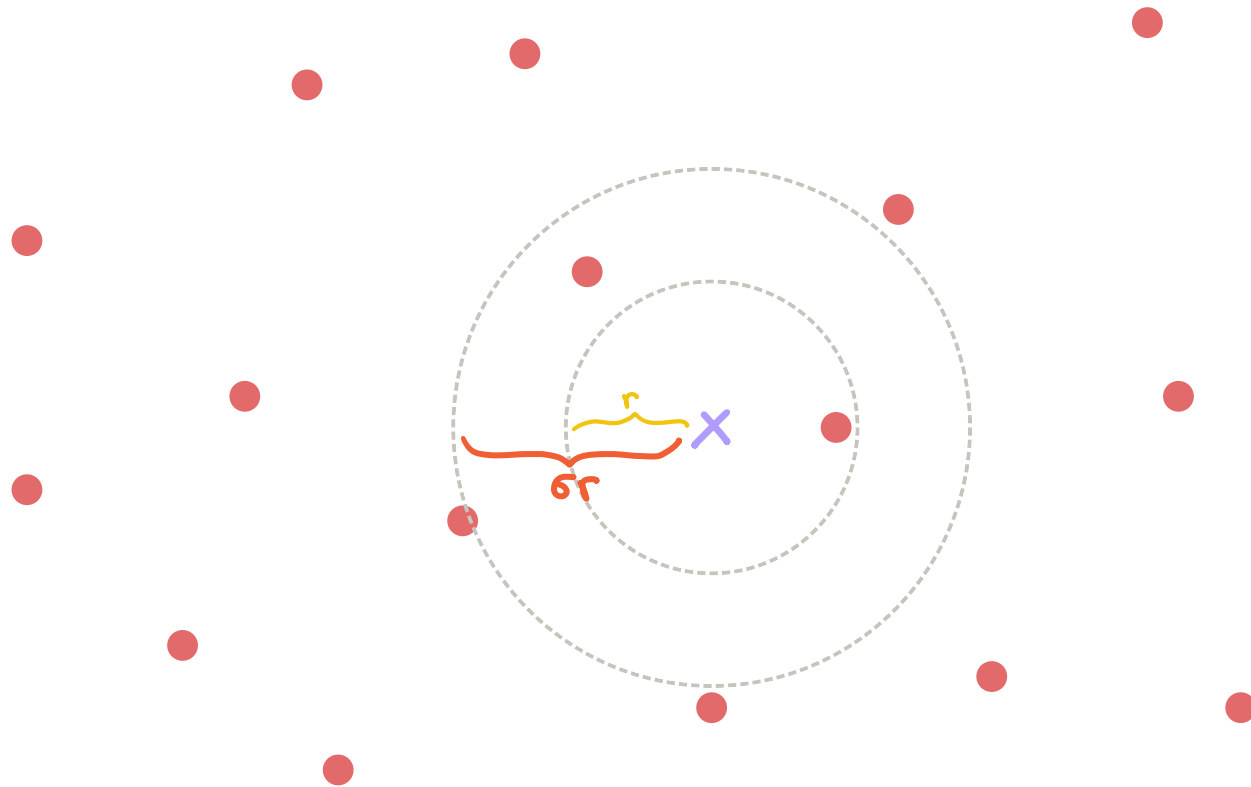
Euclidean LSH - parametrized and approximate



Fix $r > 0$, $\sigma > 1$. given query $y \in \mathbb{R}^d$ either:

- return $x \in P$ w/ $\|x - y\| \leq \sigma r$
- declare no $x \in P$ w/ $\|x - y\| \leq r$

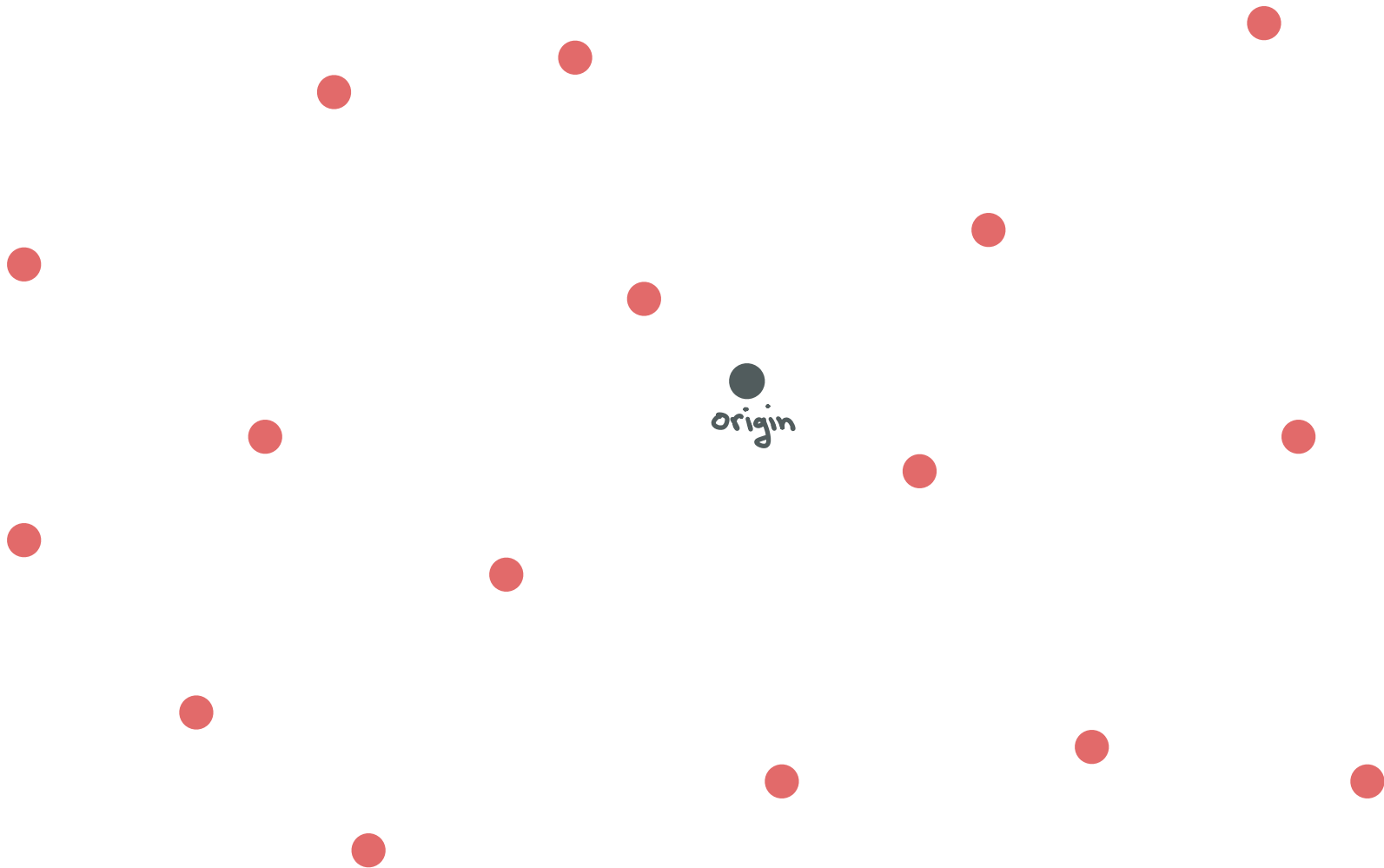
Euclidean LSH - parametrized and approximate



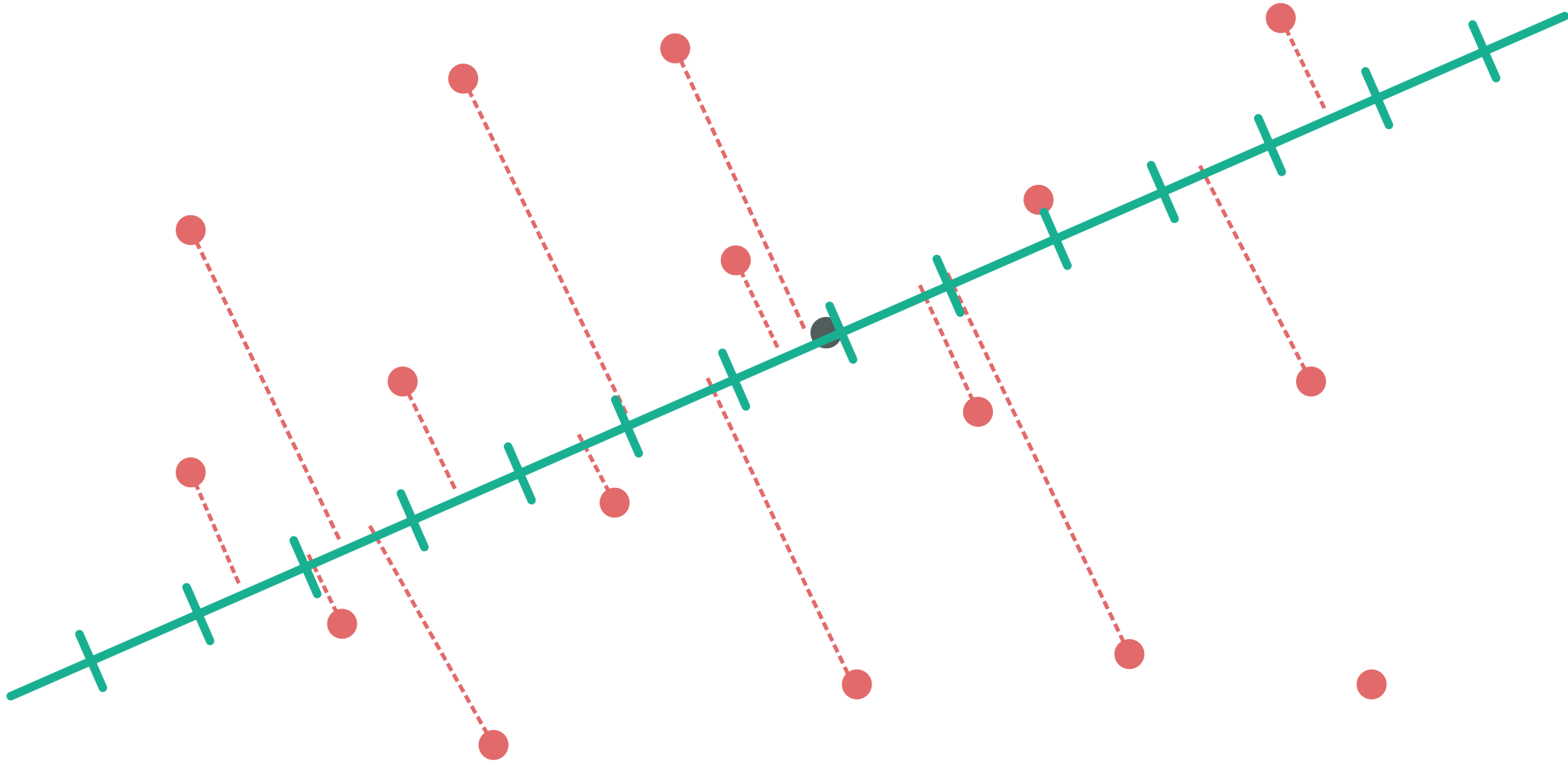
Fix $r=1$, $\sigma > 1$. given query $y \in \mathbb{R}^d$ either:

- return $x \in P$ w/ $\|x - y\| \leq \sigma$
- declare no $x \in P$ w/ $\|x - y\| \leq 1$

Ideas?

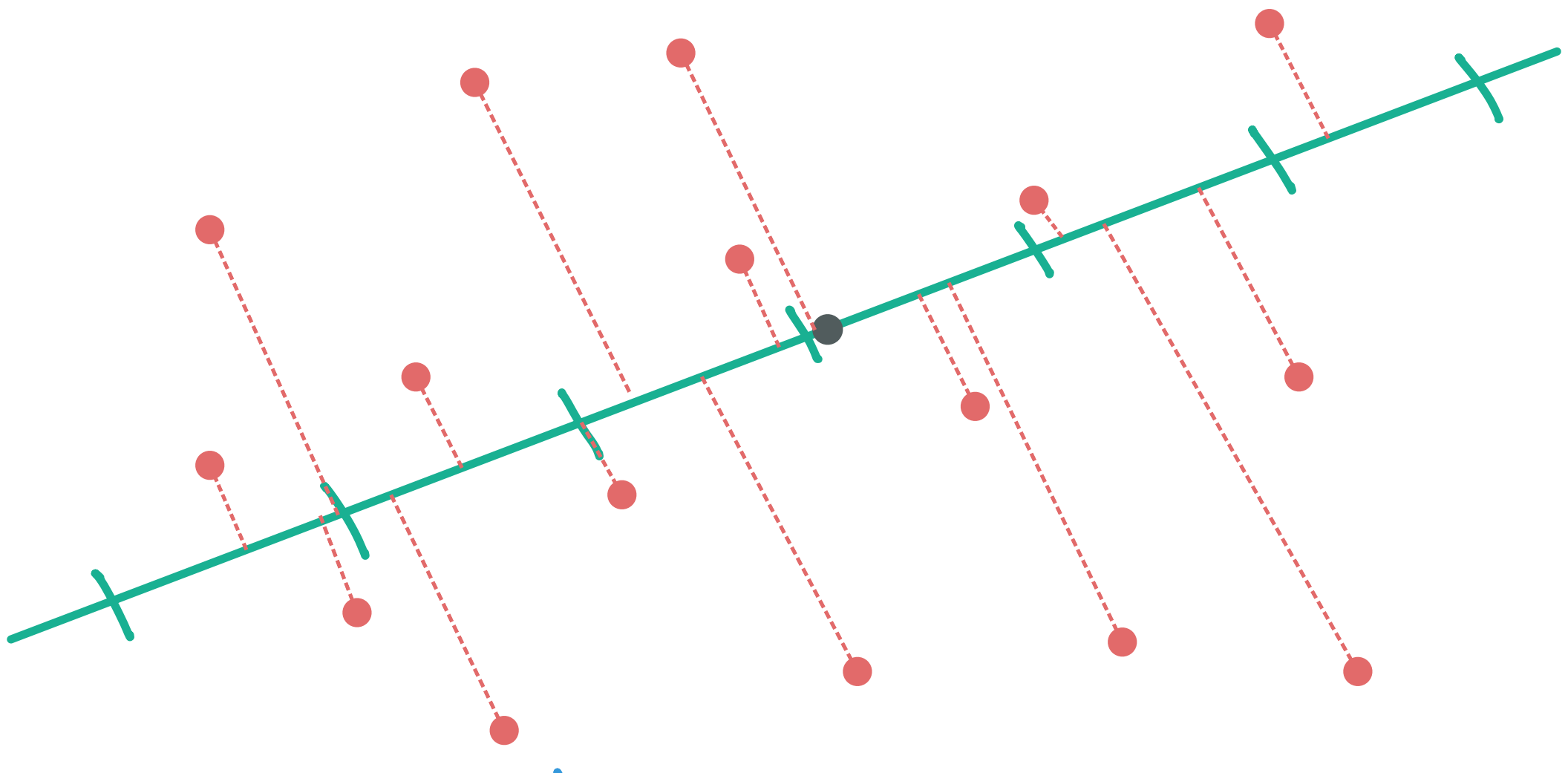


Random line embedding & $\approx$



Random line embedding & Buckets

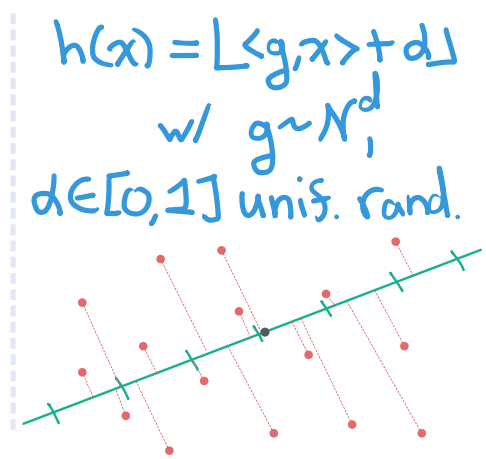
$$h(x) = \lfloor \langle g, x \rangle + d \rfloor \quad \text{w/ } g \sim \mathcal{N}^d, \\ d \in [0, 1] \text{ unif. rand.}$$



Lemma let $x, y \in \mathbb{R}^d$

$$P[h(x)=h(y) | g] = \max \{0, 1 - |\langle x - y, g \rangle|\}$$

Proof



Lemma let $x, y \in \mathbb{R}^d$, $\phi(t)$ be $\mathcal{N}(0,1)$ -density function

$$P[h(x)=h(y)] = \frac{2}{\|x-y\|} \int_0^1 (1-t) \phi\left(\frac{t}{\|x-y\|}\right) dt$$

proof $\langle g, x-y \rangle \sim \mathcal{N}(0, \|x-y\|^2)$

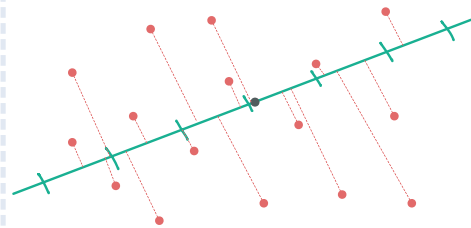
w/ density $g(t) = \frac{1}{\|x-y\|} \phi\left(\frac{t}{\|x-y\|}\right)$

$$P[h(x)=h(y)] = \int_0^\infty P[h(x)=h(y) | \langle g, x-y \rangle = t] g(t) dt$$

$$= \frac{2}{\|x-y\|} \int_0^\infty P[h(x)=h(y) | \langle g, x-y \rangle = t] \phi\left(\frac{t}{\|x-y\|}\right) dt$$

$$= \frac{2}{\|x-y\|} \int_0^1 (1-t) \phi\left(\frac{t}{\|x-y\|}\right) dt$$

$h(x) = \lfloor \langle g, x \rangle + d \rfloor$
w/ $g \sim \mathcal{N}^d$,
 $d \in [0,1]$ unif. rand.



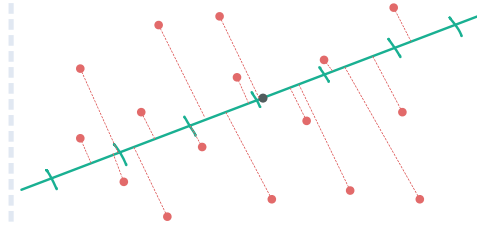
Lemma let $x, y \in \mathbb{R}^d$
 $P[h(x)=h(y) | g] =$
 $\max\{0, 1 - |\langle x-y, g \rangle|\}$

Lemma let $x, y \in \mathbb{R}^d$, $f(t)$ be $N(0,1)$ -density function

$$P[h(x)=h(y)] = \frac{2}{\|x-y\|_0} \int_0^1 (1-t) f\left(\frac{t}{\|x-y\|}\right) dt$$

decreasing in $\|x-y\|$

$h(x) = \lfloor \langle g, x \rangle + d \rfloor$
 w/ $g \sim N^d$,
 $d \in [0,1]$ unif. rand.



Lemma let $x, y \in \mathbb{R}^d$
 $P[h(x)=h(y) | g] =$
 $\max\{0, 1 - \langle x-y, g \rangle\}$

let

$$p = 2 \int_0^1 (1-t) f(t) dt \approx .368746$$

($\leq$ collision probability when $\|x-y\| \leq 1$)

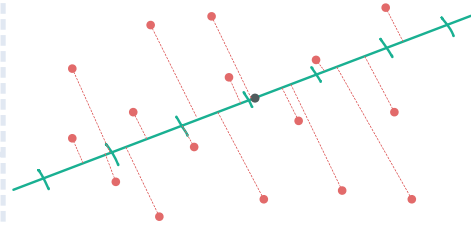
$$q = \frac{2}{\sigma} \int_0^1 (1-t) f\left(\frac{t}{\sigma}\right) dt$$

($\geq$ collision probability when $\|x-y\| \geq \sigma$)

Lemma let $x, y \in \mathbb{R}^d$, $f(t)$ be $N(0,1)$ -density function

$$P[h(x)=h(y)] = 2 \int_0^1 (1-t) f(t\|x-y\|) dt$$

$h(x) = \lfloor \langle g, x \rangle + d \rfloor$
 w/ $g \sim N^d$,
 $d \in [0,1]$ unif. rand.



$$p = 2 \int_0^1 (1-t) f(t) dt$$

($\leq$ collision probability when $\|x-y\| \leq 1$)

$$q = \frac{2}{\sigma} \int_0^1 (1-t) f\left(\frac{t}{\sigma}\right) dt$$

($\geq$ collision probability when $\|x-y\| \geq \sigma$)

Lemma let $x, y \in \mathbb{R}^d$
 $P[h(x)=h(y) | g] =$
 $\max\{0, 1 - \langle x-y, g \rangle\}$

Algo: group P by $h(x)$

to query y , scan all $x \in P$ w/ $h(x)=h(y)$

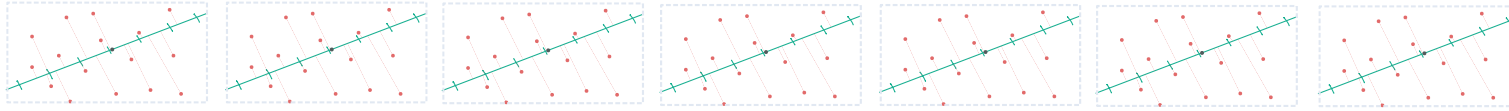
Prob. success = p $E[\text{time}] = 1 + qn$

For constant prob. success:

repeat $\frac{1}{p}$ times, $\Rightarrow E[\text{time}] = \frac{1}{p} + \frac{q}{p}n$

Amplification

$$h: \mathbb{R}^d \rightarrow \mathbb{R}^k \quad \text{w/} \quad h_i(x) = \lfloor \langle g_i, x \rangle + d_i \rfloor \quad i=1, \dots, k$$



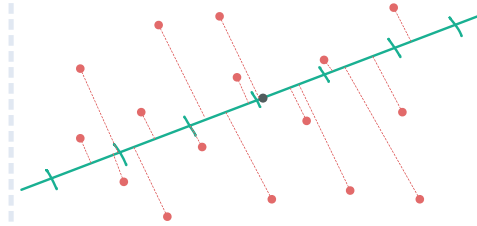
$$P[h(x)=h(y)] = \left(\frac{2}{\|x-y\|_0} \int_0^1 (1-t) f\left(\frac{t}{\|x-y\|}\right) dt \right)^k$$

$$P[h(x)=h(y)] \geq p^k \quad \text{when} \quad \|x-y\| \leq 1$$

$$P[h(x)=h(y)] \leq q^k \quad \text{when} \quad \|x-y\| \geq 6$$

$$h(x) = \lfloor \langle g, x \rangle + d \rfloor$$

w/ $g \sim \mathcal{N}^d$,
 $d \in [0,1]$ unif. rand.



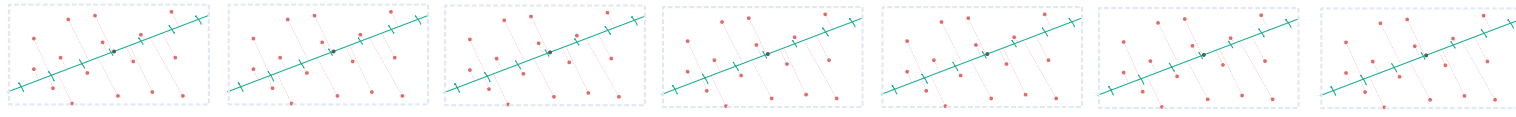
$$p = 2 \int_0^1 (1-t) f(t) dt$$

($\leq$ collision probability when $\|x-y\| \leq 1$)

$$q = 2 \int_0^1 (1-t) f(6t) dt$$

($\geq$ collision probability when $\|x-y\| \geq 6$)

$$h: \mathbb{R}^d \rightarrow \mathbb{R}^k \quad \text{w/} \quad h_i(x) = \lfloor \langle g_i, x \rangle + d_i \rfloor \quad i=1, \dots, k$$

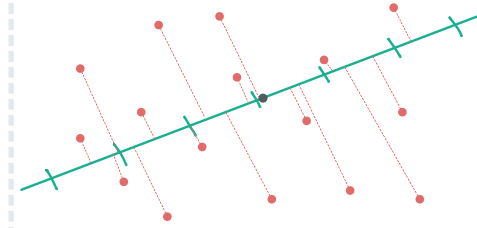


$$P[h(x) = h(y)] \geq p^k \quad \text{when} \quad \|x - y\| \leq 1$$

$$P[h(x) = h(y)] \leq q^k \quad \text{when} \quad \|x - y\| \geq \sigma$$

$$h(x) = \lfloor \langle g, x \rangle + d \rfloor$$

w/ $g \sim \mathcal{N}^d$,
 $d \in [0, 1]$ unif. rand.



$$p = 2 \int_0^1 (1-t) f(t) dt$$

($\leq$ collision probability when $\|x - y\| \leq 1$)

$$q = 2 \int_0^1 (1-t) f(\sigma t) dt$$

($\geq$ collision probability when $\|x - y\| \geq \sigma$)

Algo: group P by $h(x)$

to query y , scan all $x \in P$ w/ $h(x) = h(y)$

$$\text{Prob. success} = p^k \quad E[\text{time}] = 1 + q^k n$$

(worse) (better)

For constant prob. success:

$$\text{repeat } \frac{1}{p^k} \text{ times} \Rightarrow E[\text{time}] = \frac{1}{p^k} + \left(\frac{q}{p}\right)^k n$$

$$P[h(x)=h(y)] \geq p^k \text{ when } \|x-y\| \leq 1$$

$$P[h(x)=h(y)] \leq q^k \text{ when } \|x-y\| \geq \sigma$$

Algo: group P by $h(x)$

to query y , scan all $x \in P$ w/ $h(x) = h(y)$

$$E[\text{time}] = k\left(\frac{1}{p}\right)^k + \left(\frac{q}{p}\right)^k n$$

$$= \left(\frac{1}{p}\right)^k (k + q^k n)$$

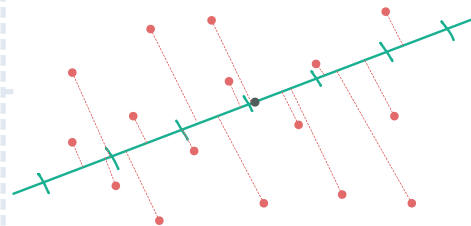
$$k = \frac{\log(n)}{\log(1/q)} \Rightarrow q^k n = 1,$$

$$k\left(\frac{1}{p}\right)^k = \frac{\log(n)}{\log(1/q)} n^{p(\sigma)}$$

$$\text{w/ } p(\sigma) = \frac{\log(1/p)}{\log(1/q)}$$

$$h(x) = \lfloor \langle g, x \rangle + d \rfloor$$

w/ $g \sim \mathcal{N}^d$,
 $d \in [0, 1]$ unif. rand.



$$p = 2 \int_0^1 (1-t) f(t) dt$$

($\leq$ collision probability when $\|x-y\| \leq 1$)

$$q = 2 \int_0^1 (1-t) f(\sigma t) dt$$

($\geq$ collision probability when $\|x-y\| \geq \sigma$)

$$P[h(x)=h(y)] \geq p^k \text{ when } \|x-y\| \leq 1$$

$$P[h(x)=h(y)] \leq q^k \text{ when } \|x-y\| \geq \sigma$$

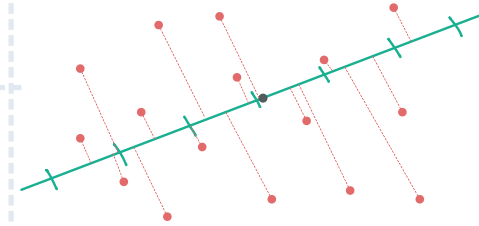
Algo: group P by $h(x)$

to query y , scan all $x \in P$ w/ $h(x) = h(y)$

$$E[\text{time}] =$$

$$h(x) = \lfloor \langle g, x \rangle + d \rfloor$$

w/ $g \sim \mathcal{N}^d$,
 $d \in [0, 1]$ unif. rand.

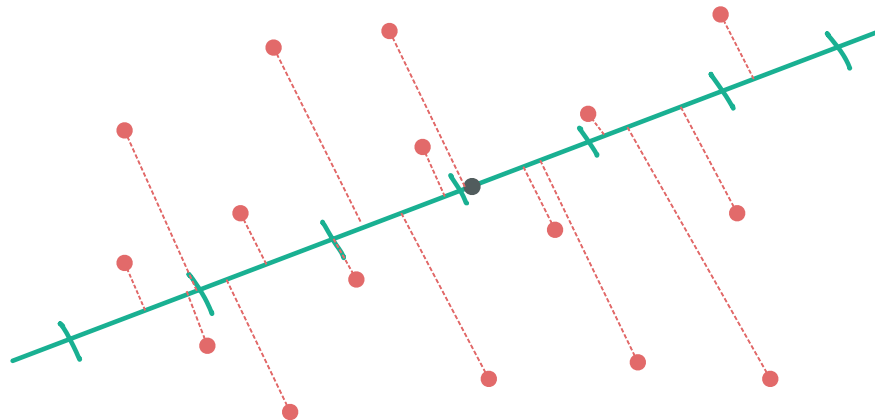


$$p = 2 \int_0^1 (1-t) f(t) dt$$

($\leq$ collision probability when $\|x-y\| \leq 1$)

$$q = 2 \int_0^1 (1-t) f(\sigma t) dt$$

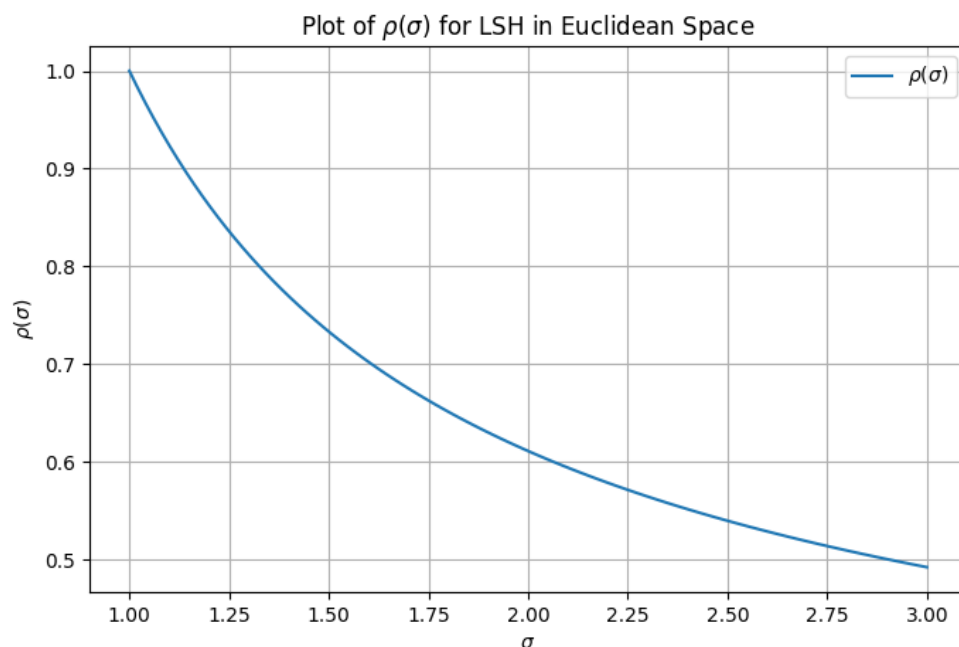
($\geq$ collision probability when $\|x-y\| \geq \sigma$)



Theorem 3.3. *One can compute a σ -approximate nearest neighbor w/r/t Euclidean distance with high probability in $\tilde{O}\left(dn^{\rho(\sigma)}\right)$ randomized time, where*

$$\rho(\sigma) \stackrel{\text{def}}{=} \frac{\log(1/p)}{\log(1/q)}, \quad p \stackrel{\text{def}}{=} 2 \int_0^1 (1-t)f(t) dt, \quad q = \frac{2}{\sigma} \int_0^1 (1-t)f(t/\sigma) dt,$$

and $f(t) = e^{-t^2/2}/\sqrt{2\pi}$ is the density function of the standard Gaussian.



running time is the above quantity times d , since we need $O(d)$ time to hash a query point or compute the distance between two points.

We can rewrite (1) as

$$\left(\frac{1}{p}\right)^k (k + q^k n).$$

Let $k = \log(n)/\log(1/q) = \log(1/n)/\log(q)$. Then $q^k n = 1$, and

$$k \left(\frac{1}{p}\right)^k = \frac{\log(n)}{\log(1/q)} n^{\log(1/p)/\log(1/q)}.$$

Consider the exponent $\log(1/p)/\log(1/q)$: since $p > q$, this quantity is less than 1.

Theorem 3.3. *One can compute a σ -approximate nearest neighbor w/r/t Euclidean distance with high probability in $\tilde{O}(dn^{\rho(\sigma)})$ randomized time, where*

$$\rho(\sigma) \stackrel{\text{def}}{=} \frac{\log(1/p)}{\log(1/q)}, \quad p \stackrel{\text{def}}{=} 2 \int_0^1 (1-t)f(t) dt, \quad q = \frac{2}{\sigma} \int_0^1 (1-t)f(t/\sigma) dt,$$

and $f(t) = e^{-t^2/2}/\sqrt{2\pi}$ is the density function of the standard Gaussian.

What is $\rho(\sigma)$? Ask the computer:

