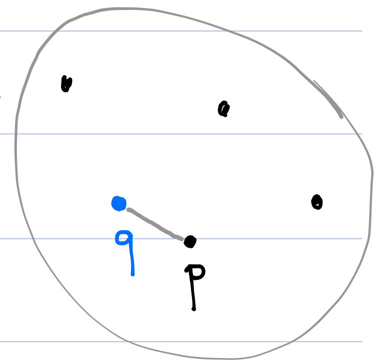
Locality Sensitive Hashing

Motivation: nearest neighbor search (NNS)

Preprocess: set S of n pts in metric space



Query: given pt q, find $p \in S$ s.t.

$d(p,q)$ min

Docs, images, etc often represented as pts in very high dimensional space

Similarity is fundamental problem in applications ranging from ecommerce to medical imaging, bioinformatics, astrophysics, finance web search, ...

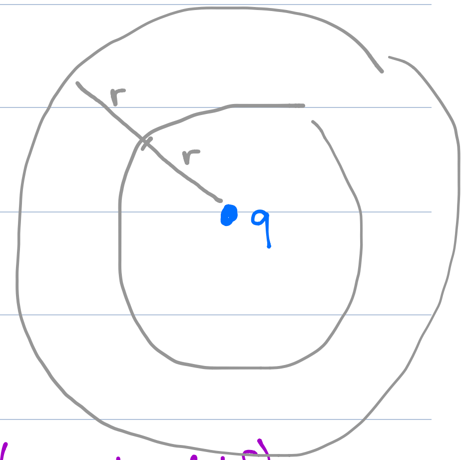
2D - Voronoi diagram $O(n)$ space query time $O(\log n)$

In high dimensions	query time	space
with preprocessing	$O(\log n\ d)$	$n^{O(d)}$
without	$O(nd)$	$O(nd)$

curse of dimensionality

Approximate NNS

c-approximate r-near neighbor search



Given query pt q, return

- all pts p s.t. $d(p,q) \le r$ (each with prob $1-\delta$)

- may return some pts s.t. $d(p,q) \le cr$

Do this with LSH [Indyk, Motwani]

This work won 2012 Kanellakis Theory and Practice Award.

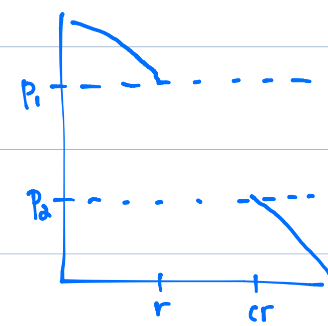
Locality sensitive hashing LSH

$\mathcal{H}$: family of hash fns mapping pts in metric space $\to \mathbb{R}$
for now $\mathbb{R}^d$

$\mathcal{H}$ is $\boxed{(r, cr, p_1, p_2) \text{ sensitive}}$ if $\forall p, q \in \mathbb{R}^d$

If $d(p,q) \le r \Rightarrow \Pr(h(p)=h(q)) \ge p_1$
random $h \in \mathcal{H}$

If $d(p,q) \ge cr \Rightarrow \Pr(h(p)=h(q)) \le p_2$



Example:

Suppose pts $\in \{0,1\}^d$

$d(p,q)$ = Hamming distance
= # bits where p & q differ
$= \sum_{1 \le i \le d} \mathbb{1}_{p_i \neq q_i}$

$\mathcal{H} = \{ h_i(p) = p_i \mid 1 \le i \le d \}$

$d(p,q) \le r$

$p_1 = \Pr(h(p) = h(q)) \ge 1 - \frac{r}{d}$
$\approx e^{-\frac{r}{d}}$

$d(p,q) \ge cr$

$p_2 = \Pr(h(p) = h(q)) \le 1 - \frac{cr}{d}$
$\approx e^{-\frac{cr}{d}}$

How do we use (r, cr, p_1, p_2) class of hash fns?

- amplify diff between p_1 & p_2 - concatenate hash fns in 3 steps

1. reduce prob that distant pts hash together
AND

2. increase prob that close pts hash together
OR

3. iterate to get failure prob below target

AND

Alg for approx NNS given (r, cr, p_1, p_2) sensitive family $\mathcal{H}$

each of these selected uniformly at random from $\mathcal{H}$

Define $g_1(p) = [\overbrace{h_1(p), \dots, h_k(p)}]$

e.g. Hamming dist $\qquad g : \{0,1\}^d \longrightarrow \{0,1\}^k$

$\forall p \in S$ hash $g_1(p) \longrightarrow$ table T_1 of size n
to reduce
space usage.

Let p, q be 2 pts s.t. $d(p,q) \geq 2r$
far

$Pr(g_1(p) = g_1(q)) \leq p_2^k$ choose k so this is $\leq \frac{1}{n}$

Why? so expected # of far pts reported on query to q is ≤ 1

$\boxed{\text{Defn:}}$ ρ satisfies $p_1 = p_2^{\rho}$

Suppose $d(\tilde{p}, q) \leq r$

Then $Pr(g_1(\tilde{p}) = g_1(q)) \geq p_1^k = (p_2^{\rho})^k = \frac{1}{n^{\rho}}$

To catch close pairs, create $L = O(n^{\rho})$ tables

and return $\tilde{p}$ if $g_i(q) = g_i(\tilde{p})$ $\forall\ 1 \leq i \leq L$

each uses fresh randomness

$$g_1(p) = \overbrace{[h_{11}(p), \dots, h_{1k}(p)]}^{\text{"signature"}} \Rightarrow \text{hash to } T_1 \text{ of size } n$$

$$g_2(p) = [h_{21}(p), \dots, h_{2k}(p)] \Rightarrow \text{hash to } T_2 \text{ of size } n$$

$$\vdots$$

$L = n^\rho$ hash tables

$$g_L(p) = [h_{L1}(p), \dots, h_{Lk}(p)] \Rightarrow \text{hash to } T_L \text{ of size } n$$

On query q, compute $T_1(g_1(q)), \dots, T_L(g_L(q))$

$\forall$ p in any of those buckets, compute $d(p,q)$

output all close points

Alg fails when an r-near neighbor p^* not in any of these buckets

$$\text{Prob of failure} = (1-p_1^k)^L = (1 - \underbrace{p_2^{\rho k}}_{\frac{1}{n^\rho}})^{n^\rho} \lesssim \frac{1}{e}$$

recall $L = n^\rho$ $\quad p_1 = p_2^\rho$

Runtime: • hash fn evaluation $O(Lk)$

• distance computations to pts in buckets

Distance computations:

care only about far pts $(> cr)$

$Pr(\text{far pt collides}) \le P_2^k = \frac{1}{n}$

$E(\#\text{ far pts in a bucket}) \le n \cdot \frac{1}{n} = 1$

$\Rightarrow E(\#\text{ far pts is } L)$

Total: $O(Lk + Ld) = O(n^{\rho}(\log n + d))$

$P_2^k = \frac{1}{n}$

$k \log(p) = -\log n$

$k = \frac{\log(n)}{\log(\frac{1}{p})}$

Space: $O(nL) = O(n^{1+\rho})$ plus space to store pts

Application: Hamming distance

Suppose pts $\in \{0,1\}^d$

$d(p,q)$ = Hamming distance
= # bits where p & q differ
$= \sum_{1 \le i \le d} \mathbb{1}_{p_i \neq q_i}$

$\mathcal{H} = \{ h_i(p) = p_i \mid 1 \le i \le d \}$

$d(p,q) \le r$

$p_1 = \Pr(h(p)=h(q)) \geq 1 - \frac{r}{d}$
$\approx e^{-\frac{r}{d}}$

$d(p,q) \geq cr$

$p_2 = \Pr(h(p)=h(q)) \le 1 - \frac{cr}{d}$
$\approx e^{-\frac{cr}{d}}$

Recall ρ defined by $p_1 = p_2^{\rho}$

$\Rightarrow \rho = \frac{1}{c}$

For example with $c=2$: query time $O(\sqrt{n}(\log n + d))$
space: $O(n^{3/2})$ + pts

LSH families

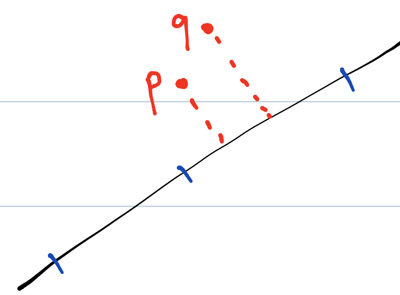
Euclidean distance in $\mathbb{R}^d$

Density of $N(0,1)$

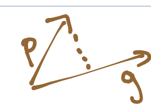
$$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$$

Let $\vec{g} = (g_1, \dots, g_d)$ iid $N(0,1)$ random vars

Fix $w \gg r$



$$h(p) = \left\lfloor \frac{p \cdot g + b}{w} \right\rfloor = \left\lfloor \frac{\sum p_i g_i + b}{w} \right\rfloor$$



$p \cdot g$ projects p onto random line

$$(p-q) \cdot g = \sum_i \underbrace{(p_i - q_i)\, g_i}_{N(0,\,(p_i-q_i)^2)}$$ weighted sum of $N(0,1)$ r.v.'s

$X_1 + \dots + X_k$ $\quad X_i \sim N(\mu_i, \sigma_i^2)$ $\implies X_1 + \dots + X_k \sim N(\mu_1 + \dots + \mu_k,\ \sigma_1^2 + \sigma_2^2 + \dots + \sigma_k^2)$

Projected distance

$$\Rightarrow \sum_i (p_i - q_i)\, g_i \quad \text{is } N\Big(0, \overbrace{\textstyle\sum (p_i - q_i)^2}^{\text{variance}}\Big)$$

$\Rightarrow$ exp distance^2 between projections

$$= E\left[(p\cdot g - q\cdot g)^2\right] = Var(p\cdot g - q\cdot g) = \underbrace{\textstyle\sum (p_i - q_i)^2}_{\text{Euclidean distance}^2}$$

Concentrated around expectation

projection approximately preserves distance

random shift ensures that likely to go to same bucket

$\rho < \frac{1}{c}$ for w carefully chosen

Jaccard

A, B docs $\equiv$ elts of $\{0,1\}^{|U|}$

$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$ $d(A,B) = 1 - J(A,B)$

Idea: permute rows of matrix at random

i.e. minhash $h_\pi = \min\{\pi(a) \mid a \in A\}$

$Pr(h_\pi(A) = h_\pi(B)) = \frac{|A \cap B|}{|A \cup B|}$

$d(A,B) \leq r$

$p_1 = Pr(h_\pi(A) = h_\pi(B)) \geq 1 - r$

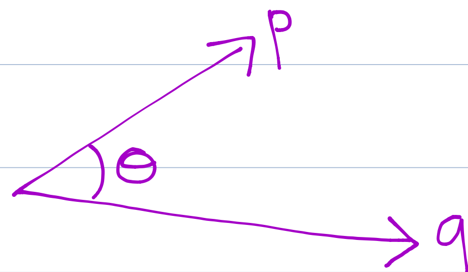
$d(A,B) \geq cr$

$p_2 = Pr(h_\pi(A) = h_\pi(B)) \leq 1 - cr$

Cosine distance   used in information retrieval / data mining

points $\in \mathbb{R}^d$

$d(p,q) = \arccos\left(\frac{p \cdot q}{\|p\| \|q\|}\right)$



pick random hyperplane thru origin $\vec{g} = (r_1, \dots, r_n)$ $r_i \sim N(0,1)$

$h(p) = \text{sign}(\vec{g} \cdot \vec{p})$ (rotationally symmetric)

$\Pr(h(p) = h(q)) = 1 - \frac{2\theta}{2\pi} = 1 - \frac{d(p,q)}{\pi}$

Other distances:

- ℓ_1
- edit distance