Approximate Nearest Neighbor Search in High Dims

Problem: Given $n$ pts $P$ in $\mathbb{R}^d$, build data structure s.t.
given query pt $q \in \mathbb{R}^d$, can find $p \in P$ minimizing $d(p, q)$.

Distance metrics:
- \(d_2\) (Euclidean):
  \[ d^2 = \sum_{i=1}^{d} (p_i - q_i)^2 \]
- \(d_1\) (Manhattan):
  \[ d = \sum_{i=1}^{d} |p_i - q_i| \]
- \(d_{\infty}\) (Chebyshev):
  \[ d = \max_{i=1}^{d} |p_i - q_i| \]

Known:
- \(d=2\): $O(n)$ space, $O(\log n)$ time by PC.
- Larger const $d$:
  - $d_1 / d_{\infty}$: $O(n \log^d n)$ space, $O(\log n)$ query time
  - $d_2$: $O(n d^{d/2})$ space, $O(\log n)$ time or $O(n)$ space, $O(n^{1 - \frac{d\log 2}{d(d+1)}})$ time

Terrible for large $d$!

Relax problem: given $q \in \mathbb{R}^d$, find $p \in P$ s.t.
\[ d(p,q) \leq c \cdot \min_{p \in P} d(p',q) \]

approximation factor
Suffice to solve approx decision problem, for fixed radius \( r \):

- given \( q \in \mathbb{R}^d \)
- return some pt \( p \in P \) with \( d(p, q) \leq r \)
- or declare all pts \( p \in P \) with \( d(p, q) > r \).

(Original prob can be solved by approx binary search extra factor \( \log U \))

Try all \( r = (1 + \epsilon)^i \), \( i = 0, \ldots, \log \frac{1}{1 + \epsilon} U \)

Method 0: Grid

- form uniform grid of side length \( Er/\sqrt{d} \)
- Store \( S = \) all nonempty grid cells

\[ \text{query}(q): \]
- check if any grid cell intersecting ball \((q, r)\) is in \( S \)

\[ \Rightarrow \quad r + 3r = (1 + 3) r \]

- \( 1 + 3 \) approx factor

\[ \text{query time } O \left( \# \text{ grid cells intersecting ball} \right) \]

\[ \leq O \left( \left( \frac{2r}{\sqrt{d}/3} \right)^d \right) \]

\[ = O \left( \left( \frac{2\sqrt{d}}{3} \right)^d \right) \]

Improves to \( O(1)^d \)
Space $O(dn)$

Alternative: to reduce query time

store $S = \bigcup \text{all grid cells intersecting } \bigcup_{p \in P} \text{ball}(\text{center})$

space $\left( \frac{C(d)}{\epsilon} \right)^d n$

query time $O(d)$

? how to avoid exponential dependence on $d$?

Method 1: Dimension Reduction (Indyk-Motwani '98)

Johnson-Lindenstrauss Lemma ('84)

For $n$ pts $P$ in $\mathbb{R}^d$, for Euclidean, $\approx <$

exists mapping $f: P \rightarrow \mathbb{R}^k$, s.t. $\forall p, q \in P$,

$d(p, q) \leq d(f(p), f(q)) \leq (1+\epsilon) d(p, q)$

with $k = O\left( \frac{1}{\epsilon^2 \log n} \right)$

(Pf: take a random projection $f$)

Chernoff bd ...
Method 2: Locality-Sensitive Hashing (LSH) (Indyk, Motwani '98)

idea: if \( p, q \) are close, more likely that \( h(p) = h(q) \)

Consider special case of Hamming space \( \{0,1\}^d \)

Given \( p, q \in \{0,1\}^d \),

define \( d(p, q) = \) # bit positions that are diff.

\[ d(p, q) = 2. \]

Will solve approx. decision problem for fixed \( r \), approx. factor \( c \).
Proc Alg:
let k be param.
define hash fn $h : \{0,1\}^d \rightarrow \{0,1\}^k$
by projecting to k rand dims

eg. $p = 101110011$
     $h(p) = 010$

for each $p \in P$
put $p$ in bucket for $h(p)$.

Query($q$): if each $p \in P$ in bucket for $h(q)$
if $d(p,q) \leq cr$ return $p$
return no

Prop
$\forall p, q \in \{0,1\}^d,$
$Pr [ h(p) = h(q) ] = \left( 1 - \frac{d(p,q)}{d} \right)^k$

eg. $p = 10\overline{0}1\overline{0}001$
    $h(p) = 010$
$q = 10\overline{0}1\overline{0}001$
    $h(q) = 010$
$d(p,q) = 2$. 