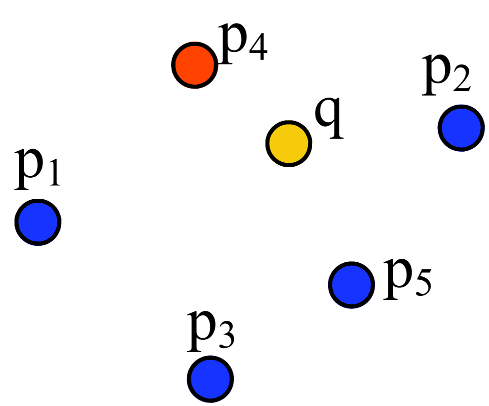
Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality

Piotr Indyk and Rajeev Motwani

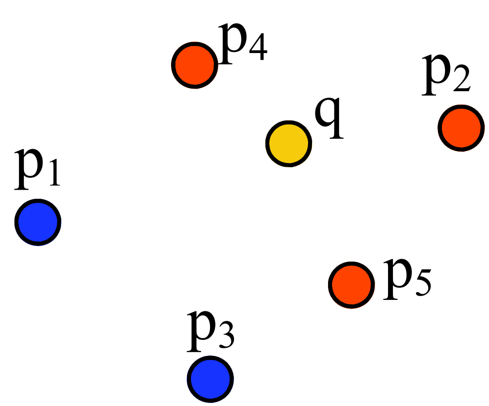
NNS definition

- Prob. Given n points $P = \{p_1, p_2, \ldots, p_n\}$, preprocess P so as to efficiently answer queries for finding the point in P closest to a query point q.



ϵ-NNS definition

- Prob. Given n points $P = \{p_1, p_2, \ldots, p_n\}$, preprocess P so as to efficiently answer queries for finding a point $p \in P$ satisfying $(\forall p' \in P)[d(p,q) \leq (1+\epsilon)d(p',q)]$, for a query point q.



Results

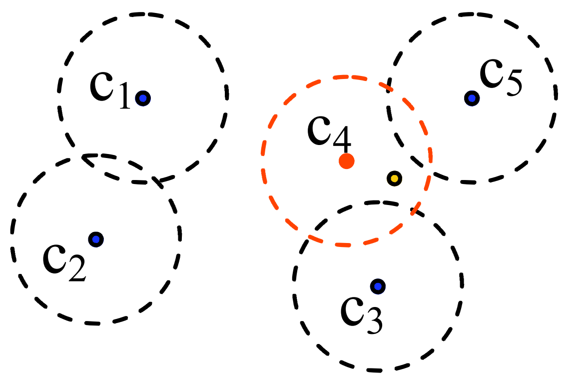
Algorithms for ϵ-NNS in l_p^d with:

ϵ	p	preprocessing	query
$\epsilon > 0$	1 or 2	$\tilde{O}(n^{1+1/(1+\epsilon)} + dn)$	$\tilde{O}(dn^{1/(1+\epsilon)})$
$0 < \epsilon < 1$	arbitrary	$\tilde{O}(n) \times O(1/\epsilon)^d$	$\tilde{O}(d)$
$\epsilon > 0$	1 or 2	$(nd)^{O(1)}$	$\tilde{O}(d)$

PLEB definition

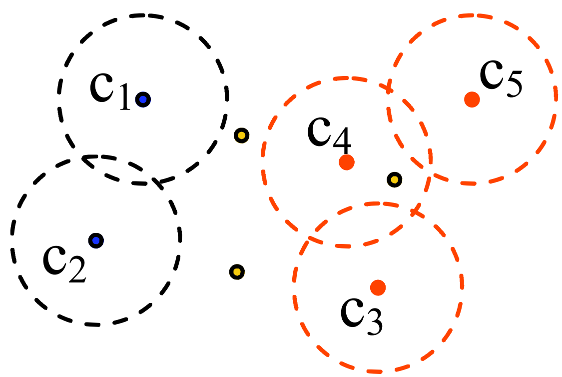
- Prob. Given $r > 0$ and n points $C = \{c_1, c_2, \ldots, c_n\}$ in (X, d), devise a data structure that, for any query $q \in X$:

  if $\exists c_i \in C$ s.t. $q \in B(c_i, r)$, returns c_i, else returns NO



ϵ-PLEB definition

- Prob. Given $r > 0$ and n points $C = \{c_1, c_2, \ldots, c_n\}$ in (X, d), devise a data structure that, for any query $q \in X$:
  - if $\exists c_i \in C$ s.t. $q \in B(c_i, r)$, returns YES and some $c'_i \in C$ s.t. $q \in B(c'_i, (1+\epsilon)r)$.
  - if $\forall c_i \in C$ $q \notin B(c_i, (1+\epsilon)r)$, returns NO.
  - otherwise, returns either YES or NO.



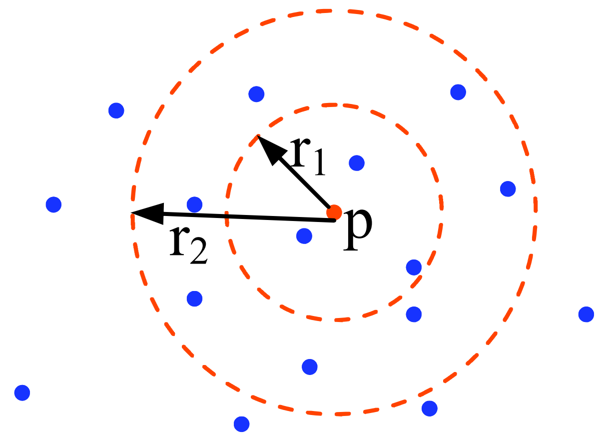
A simple reduction from ϵ-NNS to ϵ-PLEB

- let m be the smallest and M be the largest inter-point distances in P.
- Find the smallest l such that for some p_i, $q \in B(p_i, m(1+\epsilon)^l)$.
  - binary search in $\{m, m(1+\epsilon), m(1+\epsilon)^2, \ldots, M\}$.
- Return p_i as an approximate nearest neighbour.

$$d(q, p_i) > (1+\epsilon) d(q, p^\star) \Rightarrow$$

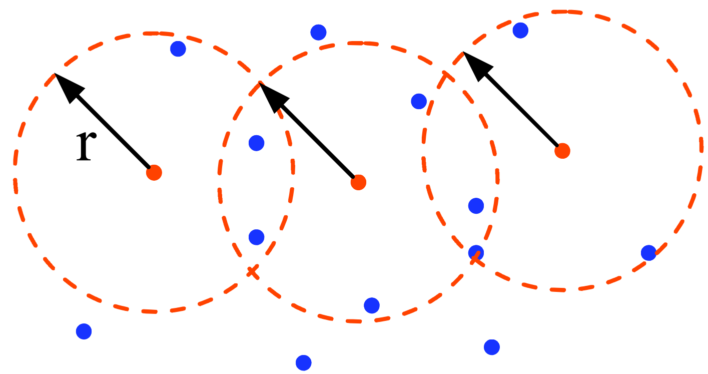
$$d(q, p^\star) < \frac{d(q,p_i)}{1+\epsilon} \leq \frac{m(1+\epsilon)^l}{1+\epsilon} = m(1+\epsilon)^{l-1}$$

Definition of $(\alpha_1, \alpha_2, \beta)$-ring separator for P



- #points inside the r_1 circle $\geq$ a fraction α_1 of $|P|$
- #points outside the r_2 circle $\geq$ a fraction α_2 of $|P|$
- $\frac{r_2}{r_1} = \beta$
- $\beta > 1$, $\alpha_1, \alpha_2 > 0$

Definition of (γ, δ)-cluster for P



A subset $S \subset P$ such that, for each $p \in S$:

- #points in $B(p, r) \leq \delta |P|$
- $r = \gamma \Delta(S)$
- $0 < \gamma, \delta < 1$

Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.

Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.

Proof. For any point $p \in P$ define functions:

- $f_p^{\infty}(r) = |P - B(p, \beta r)|$
- $f_p^{0}(r) = |P \cap B(p, r)|$

$\exists r_p$ such that $f_p^0(r_p) = f_p^{\infty}(r_p)$.

Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.

Assuming P does not have an (α, α, β)-ring separator, we must have:

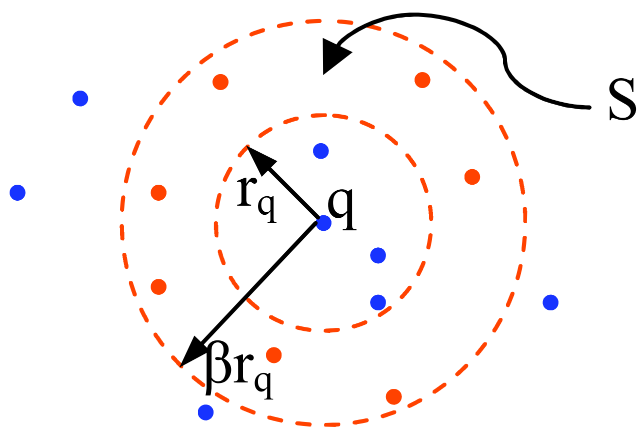
$$f_p^0(r_p) = f_p^\infty(r_p) \leq \alpha n$$

Pick a point q with minimum r_q.

Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

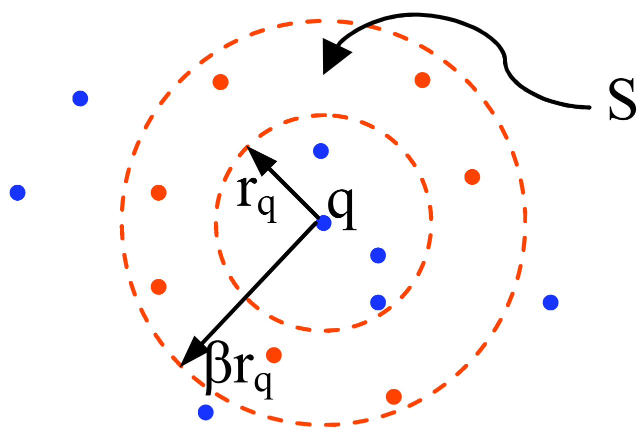
- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.



Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.

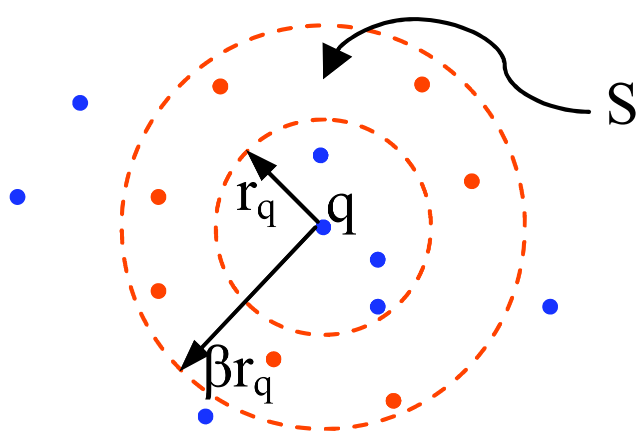


- $|S| \geq (1 - 2\alpha)n$
- $\Delta(S) \leq 2\beta r_q$
- $|P \cap B(s, \gamma\Delta(S))| \leq$
  $|P \cap B(s, r_q)| \leq$
  $|P \cap B(s, r_s)| \leq \alpha n$

Existence of ring-separator or cluster

Thm. For any P, $0 < \alpha < 1$, $\beta > 1$:

- either P has an (α, α, β)-ring separator,
- or P has a $(\frac{1}{2\beta}, \alpha)$-cluster of size at least $(1 - 2\alpha)|P|$.



- $|S| \geq (1 - 2\alpha)n$
- $\Delta(S) \leq 2\beta r_q$
- $|P \cap B(s, \gamma\Delta(S))| \leq$
  $|P \cap B(s, r_q)| \leq$
  $|P \cap B(s, r_s)| \leq \alpha n \qquad \square$

Definition of (b, c, d)-cover for $\mathcal{SCP}$

A sequence $A_1, \ldots, A_l$ of sets $A_i \subset P$ with $S \subset A \triangleq \bigcup_i A_i$ such that $\exists r \geq d\Delta(A)$ satisfying, for each $i = 1, \ldots, l$:

$$\frac{1}{b} \left| P \cap \left(\bigcup_{p \in A_i} B\left(p, r\right) \right) \right| \leq |A_i| \leq c\,|P|$$

where $b > 1$, $0 < c < 1$, $d > 0$.

Constructing a cover for a cluster

Thm. S: (γ, δ)-cluster for $P \rightsquigarrow A_1, \ldots, A_k \subset P$: $(b, \delta, \frac{\gamma}{(1+\gamma)\log_b n})$-cover for S, $\forall b > 1$.

Alg. Cover:

$r \leftarrow \frac{\gamma \Delta(S)}{\log_b n}$; $j \leftarrow 0$;

repeat $j \leftarrow j + 1$; pick $p_j \in S$; $A_j \leftarrow \{p_j\}$;

while $|P \cap \bigcup_{q \in A_j} B(q, r)| > b|A_j|$ do

$A_j \leftarrow P \cap \bigcup_{q \in A_j} B(q, r)$;

$S \leftarrow S - A_j$; $P \leftarrow P - A_j$;

until $S = \emptyset$;

$k \leftarrow j$;

Towards Ring-Cover Trees

Cor. For any P, $0 < \alpha < 1$, $\beta > 1$, $b > 1$, one of the following holds:

- either P has a (α, α, β)-ring separator $R(p, r, \beta r)$,
- or there is a (b, α, d)-cover for some $S \subset P$ with $|S| \geq (1 - 2\alpha)|P|$ and $d = \frac{1}{(1+2\beta)\log_b |P|}$.

At the root of the Ring-Cover Tree is P. Its nodes are tagged as ring nodes or cover nodes according to which of the above cases holds.

Construction of Ring-Cover Trees

work with $\beta = 2(1 + \frac{1}{\epsilon})$, $b = 1 + \frac{1}{\log^2 n}$, $\alpha = \frac{1}{2}(1 - \frac{1}{\log n})$

Case 1. P is a ring node with separator $R(p, r, \beta r)$.

- children: $S_1 = P \cap B(p, \beta r)$, $S_2 = P - B(p, r)$.
- store ring separator data in the node.

Construction of Ring-Cover Trees

Case 2. P is a cover node with cover $\{A_i\}$ for $S \subset P$.

- children: $S_0 = P - A$, $S_i = P \cap \bigcup_{p \in A_i} B(p, r)$, where $r = \frac{\gamma \Delta(S)}{\log_b n} = \frac{\Delta(S)}{2\beta \log_b n}$.
- set $r_0 \leftarrow (1 + \frac{1}{\epsilon})\Delta(S)$, $r_i \leftarrow \frac{r_0}{(1+\epsilon)^i}$, for $i \in \{1, \ldots, k\}$ where $k = \log_{1+\epsilon} \frac{(1+1/\epsilon)\log_b n}{\gamma} + 1$. Store PLEB instances $\langle r_i, A \rangle$ in the node.

Thm. The Ring-Cover Tree can be constructed in deterministic $\tilde{O}(n^2)$ time.

Searching in a Ring-Cover Tree

Search(q, P)

If P is a ring node with an (α, α, β)-ring separator $R(p, r, \beta r)$, and if $d(p, q) \leq r(1 + 1/\epsilon)$:

return Search(q, S_1)

Searching in a Ring-Cover Tree

Search(q, P)

If P is a ring node with an (α, α, β)-ring separator $R(p, r, \beta r)$, and if $d(p, q) \leq r(1 + 1/\epsilon)$:

return Search(q, S_1)

Proof. For any $s \in P - S_1$,
$d(s, q) \geq d(s, p) - d(q, p) \geq \beta r - d(q, p) \geq$
$2(1 + 1/\epsilon)r - (1 + 1/\epsilon)r = (1 + 1/\epsilon)r \geq d(q, p)$

Searching in a Ring-Cover Tree

Search(q, P)

If P is a ring node with an (α, α, β)-ring separator $R(p, r, \beta r)$, and if $d(p, q) > r(1 + 1/\epsilon)$:

compute $p' = \text{Search}(q, S_2)$ and return $\min_q(p, p')$

Searching in a Ring-Cover Tree

Search(q, P)

If P is a ring node with an (α, α, β)-ring separator $R(p, r, \beta r)$, and if $d(p, q) > r(1 + 1/\epsilon)$:

compute $p' = \text{Search}(q, S_2)$ and return $\min_q(p, p')$

Proof. For any $s \in P - S_2 = B(p, r)$,
$d(q, s) \geq d(q, p) - d(s, p) \geq d(q, p) - r$. Therefore,
$\frac{d(q,p)}{d(q,s)} \leq \frac{d(q,p)}{d(q,p)-r} = 1 + \frac{r}{d(q,p)-r} \leq 1 + \epsilon$

Searching in a Ring-Cover Tree

Search(q, P)

If P is a cover node with a (b, α, d)-cover $A_1, \ldots, A_l$ of radius r for $S \subset P$, and if $(\forall a \in A)[d(q, a) > r_0]$:

compute $p = \text{Search}(q, P - A)$, pick any $a \in A$ and return $\min_q(p, a)$

Searching in a Ring-Cover Tree

Search(q, P)

If P is a cover node with a (b, α, d)-cover $A_1, \ldots, A_l$ of radius r for $S \subset P$, and if $(\exists a \in A)[d(q,a) \leq r_0]$ but $(\forall a' \in A)[d(q,a') > r_k]$:

find an ϵ-NN p of q in A by binary search in r_i's, compute $p' = \text{Search}(q, P - A)$ and return $\min_q(p, p')$

Searching in a Ring-Cover Tree

Search(q, P)

If P is a cover node with a (b, α, d)-cover $A_1, \ldots, A_l$ of radius r for $S \subset P$, and if $(\exists a \in A_i)[d(q, a) \leq r_k]$:

return Search(q, S_i)

Properties of Ring-Cover Trees

- depth $O(\log^2 n)$.
- the Search procedure requires $O(\log^2 n \times \log k)$ distance computations or PLEB queries.
- space $O(n \text{polylog} n)$ not counting PLEB implementation space.
- given an $f(n)$-space algorithm for PLEB, total space is $O(f(n \text{polylog} n))$.
- for any PLEB instance $\langle A, r \rangle$ in a cover node, $\frac{\Delta(A)}{r} = O(\frac{1+\epsilon}{\gamma} \log_b n)$.

Solving ϵ-PLEB

2 methods:

- Bucketing
  - works for $0 < \epsilon < 1$ and any l_p norm
- Locality-Sensitive Hashing
  - applies directly only to Hamming spaces
  - works for l_1^d and l_2^d by embedding those into suitable Hamming spaces

The Bucketing Method

- Impose a uniform grid of spacing $\epsilon/d^{1/p}$ on $\mathfrak{R}^d$.
- For each ball B, store all cuboids intersecting B in a hash table (together with information about B).
- To answer query q, compute the cell containing q and check if it is in the table.
- #cuboids intersecting B is $O(1/\epsilon)^d$, therefore space is $O(n) \times O(1/\epsilon)^d$.
- Hash function evaluation in $O(d)$ time, access time $O(1)$.

Similarity measures

- Define a ball of radius r for a similarity measure D as $B(q,r) = \{p : D(q,p) \geq r\}$.
- Generalize ϵ-PLEB to (r_1, r_2)-PLEB, where for any query point q:
  - if $P \cap B(q, r_1) \neq \emptyset$ answer YES
  - if $P \cap B(q, r_2) = \emptyset$ answer NO

Definition of locality-sensitive functions

Def. A family $\mathcal{H} = \{h : S \to U\}$ is called (r_1, r_2, p_1, p_2)-sensitive for D if for any $q, p \in S$:

- $p \in B(q, r_1) \Rightarrow Pr_{\mathcal{H}}[h(q) = h(p)] \geq p_1$
- $p \notin B(q, r_2) \Rightarrow Pr_{\mathcal{H}}[h(q) = h(p)] \leq p_2$

When D is a similarity measure, we should have $r_1 > r_2$ and $p_1 > p_2$.

An algorithm for (r_1, r_2)-PLEB

For k and l to be fixed later:

- define a family $\mathcal{G} = \{g : S \rightarrow U^k\}$ with $g(p) = (h_1(p), \ldots, h_k(p))$. Pick $g_1, \ldots, g_l \in \mathcal{G}$ uniformly at random.
- preprocessing: store each $p \in P$ in buckets $g_1(p), \ldots, g_l(p)$.
- query: search buckets $g_1(q), \ldots, g_l(q)$ until you find $2l$ elements, resulting in t discrete elements $p_1, \ldots, p_t$. If for some p_j, $p_j \in B(q, r_2)$ then return YES and p_j, else return NO.

Fixing the parameters k and l

We fix the parameters k and l to ensure that with constant probability the following properties hold:

- if $\exists p^\star \in B(q, r_1)$ then $(\exists j)[g_j(p^\star) = g_j(q)]$.
- the total number of collisions of q with points in $P - B(q, r_2)$ is less than $2l$.

Can be done, given an (r_1, r_2, p_1, p_2)-sensitive family for D. Results in a $O(dn + n^{1+\rho})$-space and $O(n^\rho)$-time algorithm for (r_1, r_2)-PLEB.

Existence of a locality-sensitive family for the Hamming metric

Thm. For any $r, \epsilon > 0$ the family $\mathcal{H} = \{h_i : h_i((b_1, \ldots, b_d)) = b_i, i = 1, \ldots, n\}$ is $\left(r, r(1+\epsilon), 1 - \frac{r}{d}, 1 - \frac{r(1+\epsilon)}{d}\right)$-sensitive for the Hamming distance in $\mathcal{H}^d$.

Cor. For any $\epsilon > 0$, there is an algorithm for ϵ-PLEB in $\mathcal{H}^d$ using $O(dn + n^{1+1/(1+\epsilon)})$ space and $O(n^{1/(1+\epsilon)})$ hash function evaluations for each query. The hash function can be evaluated using $O(d)$ operations.