Beyond Locality-Sensitive Hashing

Alexandr Andoni[1] Piotr Indyk[2] Huy L. Nguyễn[3]
Ilya Razenshteyn[2]

[1]Microsoft Research SVC

[2]MIT, CSAIL

[3]Princeton

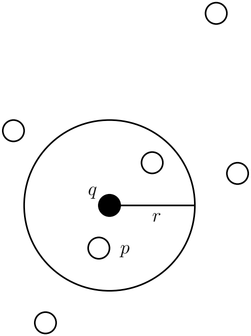
SODA 2014

The Near Neighbor Problem

- Let P be an n-point subset of a metric (X, D), $r > 0$
- For $q \in X$ find any $p \in P$ with $D(p, q) \leq r$

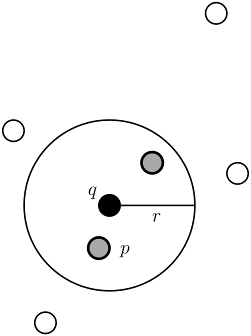
The Near Neighbor Problem

- Let P be an n-point subset of a metric (X, D), $r > 0$
- For $q \in X$ find any $p \in P$ with $D(p, q) \leq r$



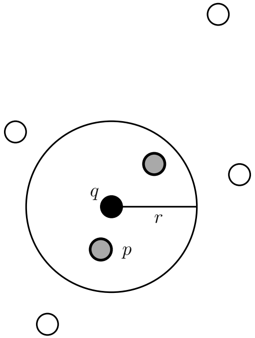
The Near Neighbor Problem

- Let P be an n-point subset of a metric (X, D), $r > 0$
- For $q \in X$ find any $p \in P$ with $D(p, q) \leq r$



The Near Neighbor Problem

- Let P be an n-point subset of a metric (X, D), $r > 0$
- For $q \in X$ find any $p \in P$ with $D(p, q) \leq r$
- Hard, if (X, D) is *high-dimensional* (space or query time is exponential in the dimension)

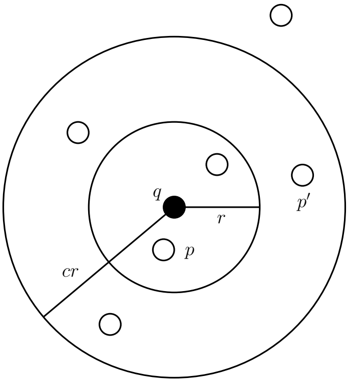


The Approximate Near Neighbor Problem (ANN)

- Let P be an n-point subset of a metric (X, D), $r > 0$, $c > 1$
- For $q \in X$ find any $p' \in P$ with $D(p', q) \leq cr$, provided that there exists $p \in P$ with $D(p, q) \leq r$

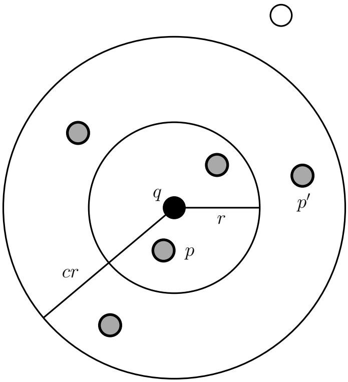
The Approximate Near Neighbor Problem (ANN)

- Let P be an n-point subset of a metric (X, D), $r > 0$, $c > 1$
- For $q \in X$ find any $p' \in P$ with $D(p', q) \leq cr$, provided that there exists $p \in P$ with $D(p, q) \leq r$



The Approximate Near Neighbor Problem (ANN)

- Let P be an n-point subset of a metric (X, D), $r > 0$, $c > 1$
- For $q \in X$ find any $p' \in P$ with $D(p', q) \leq cr$, provided that there exists $p \in P$ with $D(p, q) \leq r$



Literature

- **Exponential dependence on the dimension:**
  (Arya, Mount 1993), (Meister 1993), (Clarkson 1994), (Arya, Mount, Netanyahu, Silverman, We, 1998), (Kleinberg, 1997), (Har-Peled 2002)
- **Polynomial dependence on the dimension:**
  (Indyk, Motwani 1998), (Kushilevitz, Ostrovsky, Rabani 1998), (Indyk 1998), (Indyk 2001), (Gionis, Indyk, Motwani 1999), (Charikar 2002), (Datar, Immorlica, Indyk, Mirrokni 2004), (Chakrabarti, Regev 2004), (Panigrahy 2006), (Ailon, Chazelle 2006), (Andoni, Indyk 2006), (Indyk, Kapralov 2013), (Nguyễn 2013)

Locality-Sensitive Hashing (LSH)

- The goal: solve ANN with polynomial in the dimension space and query time, near-linear in n space, and sublinear in n query time

Locality-Sensitive Hashing (LSH)

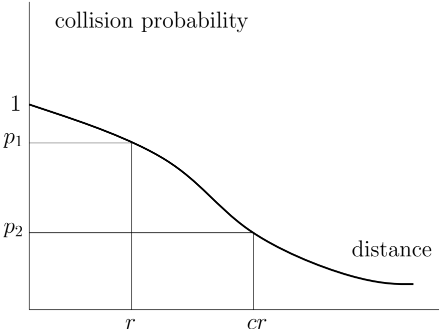
- The goal: solve ANN with polynomial in the dimension space and query time, near-linear in n space, and sublinear in n query time
- The only known technique: **Locality-Sensitive Hashing (LSH)** (Indyk, Motwani 1998)

Locality-Sensitive Hashing (LSH)

- The goal: solve ANN with polynomial in the dimension space and query time, near-linear in n space, and sublinear in n query time
- The only known technique: **Locality-Sensitive Hashing (LSH)** (Indyk, Motwani 1998)
- A hash family $\mathcal{H}$ on (X, D) is (r, cr, p_1, p_2)-sensitive, if for every $p, q \in X$:
  - if $D(p, q) \leq r$, then $\mathrm{Pr}_{h\sim\mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p, q) \geq cr$, then $\mathrm{Pr}_{h\sim\mathcal{H}}[h(p) = h(q)] \leq p_2$

Locality-Sensitive Hashing (LSH)

- The goal: solve ANN with polynomial in the dimension space and query time, near-linear in n space, and sublinear in n query time
- The only known technique: **Locality-Sensitive Hashing (LSH)** (Indyk, Motwani 1998)
- A hash family $\mathcal{H}$ on (X, D) is (r, cr, p_1, p_2)-sensitive, if for every $p, q \in X$:
  - if $D(p, q) \leq r$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p, q) \geq cr$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \leq p_2$



From LSH to ANN

- Let $\mathcal{H}$ be a "reasonable" (r, cr, p_1, p_2)-sensitive family

From LSH to ANN

- Let $\mathcal{H}$ be a "reasonable" (r, cr, p_1, p_2)-sensitive family
- Define "quality" of $\mathcal{H}$ as

$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)}$$

From LSH to ANN

- Let $\mathcal{H}$ be a "reasonable" (r, cr, p_1, p_2)-sensitive family
- Define "quality" of $\mathcal{H}$ as

$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)}$$

- Then, can solve ANN with roughly $O(n^{1+\rho} + nd)$ space and $O(d \cdot n^\rho)$ query time (Indyk, Motwani 1998)

From LSH to ANN

- Let $\mathcal{H}$ be a "reasonable" (r, cr, p_1, p_2)-sensitive family
- Define "quality" of $\mathcal{H}$ as

$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)}$$

- Then, can solve ANN with roughly $O(n^{1+\rho} + nd)$ space and $O(d \cdot n^{\rho})$ query time (Indyk, Motwani 1998)
- Example: $\{0,1\}^d$ with Hamming distance; Let $\mathcal{H} = \{h_1, \ldots, h_d\}$, where $h_i(x) = x_i$; One can check that $\rho \leq 1/c$

From LSH to ANN

- Let $\mathcal{H}$ be a "reasonable" (r, cr, p_1, p_2)-sensitive family
- Define "quality" of $\mathcal{H}$ as

$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)}$$

- Then, can solve ANN with roughly $O(n^{1+\rho} + nd)$ space and $O(d \cdot n^{\rho})$ query time (Indyk, Motwani 1998)
- Example: $\{0,1\}^d$ with Hamming distance; Let $\mathcal{H} = \{h_1, \ldots, h_d\}$, where $h_i(x) = x_i$; One can check that $\rho \leq 1/c$

11101110
10111101

Known LSH constructions

(Indyk, Motwani 1998), (Andoni, Indyk 2006),
(Motwani, Naor, Panigrahy 2007), (O'Donnell, Wu, Zhou 2011),
(Indyk, Kapralov 2013), (Nguyễn 2013)

Bounds on $\rho = \ln(1/p_1)/\ln(1/p_2)$ for various spaces:

Space	Upper bound	Lower bound
ℓ_1	$\rho \leq 1/c$	$\rho \geq 1/c - o(1)$
ℓ_p $1 < p < 2$	$\rho \leq O(1/c^p)$	$\rho \geq 1/c^p - o(1)$
ℓ_2	$\rho \leq 1/c^2 + o(1)$	$\rho \geq 1/c^2 - o(1)$

Known LSH constructions

(Indyk, Motwani 1998), (Andoni, Indyk 2006),
(Motwani, Naor, Panigrahy 2007), (O'Donnell, Wu, Zhou 2011),
(Indyk, Kapralov 2013), (Nguyễn 2013)

Bounds on $\rho = \ln(1/p_1)/\ln(1/p_2)$ for various spaces:

Space	Upper bound	Lower bound
ℓ_1	$\rho \le 1/c$	$\rho \ge 1/c - o(1)$
ℓ_p $1 < p < 2$	$\rho \le O(1/c^p)$	$\rho \ge 1/c^p - o(1)$
ℓ_2	$\rho \le 1/c^2 + o(1)$	$\rho \ge 1/c^2 - o(1)$

This work: ANN in space $O(n^{1+\tau} + nd)$ and time $O(dn^{\tau})$, where

- $\tau \le \frac{7}{8c} + O\left(\frac{1}{c^{3/2}}\right) + o(1)$ for ℓ_1
- $\tau \le \frac{7}{8c^2} + O\left(\frac{1}{c^3}\right) + o(1)$ for ℓ_2

The first improvement upon (Indyk, Motwani 1998) for ℓ_1 and (Andoni, Indyk 2006) for ℓ_2!

The main idea

- LSH is oblivious, can we construct a hash family that would depend on the data?

The main idea

- LSH is oblivious, can we construct a hash family that would depend on the data?
- $\mathcal{H}$ is (r, cr, p_1, p_2)-*sensitive*, if *for every* $p, q \in X$
  - if $D(p, q) \le r$, then $\mathrm{Pr}_{h\sim\mathcal{H}}[h(p) = h(q)] \ge p_1$;
  - if $D(p, q) \ge cr$, then $\mathrm{Pr}_{h\sim\mathcal{H}}[h(p) = h(q)] \le p_2$

The main idea

- LSH is oblivious, can we construct a hash family that would depend on the data?
- $\mathcal{H}$ is (r, cr, p_1, p_2)-*sensitive*, if *for every* $p, q \in X$
  - if $D(p,q) \leq r$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p,q) \geq cr$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \leq p_2$
- **Too strong!** Enough to satisfy these for $p \in P$ and $q \in X$. Can exploit the geometry of P to construct a better family

The main idea

- LSH is oblivious, can we construct a hash family that would depend on the data?
- $\mathcal{H}$ is (r, cr, p_1, p_2)*-sensitive*, if *for every* $p, q \in X$
  - if $D(p, q) \leq r$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p, q) \geq cr$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \leq p_2$
- **Too strong!** Enough to satisfy these for $p \in P$ and $q \in X$. Can exploit the geometry of P to construct a better family
- Parallels with practice!
  - PCA trees (Sproull 1991), (McNames 2001), (Verma, Kpotufe, Dasgupta 2009)
  - Spectral Hashing (Weiss, Torralba, Fergus 2008)
  - Semantic Hashing (Salakhutdinov, Hinton 2009)
  - WTA Hashing (Yagnik, Strelow, Ross, Lin 2011)

The main idea (contd)

- From now on, looking at the Euclidean case and trying to improve upon $\rho \leq 1/c^2$ (Andoni, Indyk 2006)

The main idea (contd)

- From now on, looking at the Euclidean case and trying to improve upon $\rho \leq 1/c^2$ (Andoni, Indyk 2006)
- Partition P into low-diameter clusters (of diameter $O(cr)$)
- Improve upon $1/c^2$ for the low-diameter case

The low-diameter case

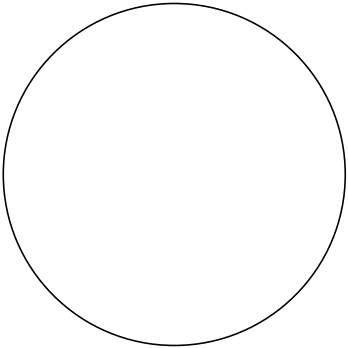
- All points and queries are on a sphere of radius $O(cr)$

The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$
using "ball carving" (similar to (Karger, Motwani, Sudan 1998))

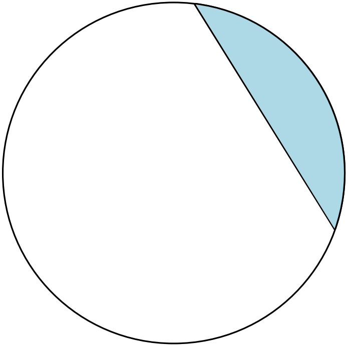
The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$
using "ball carving" (similar to (Karger, Motwani, Sudan 1998))



The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$
using "ball carving" (similar to (Karger, Motwani, Sudan 1998))

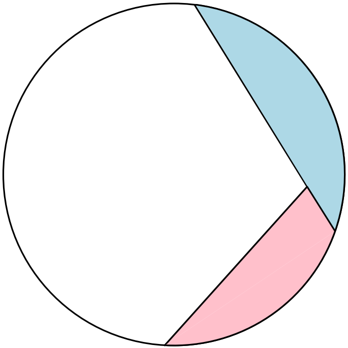


The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve

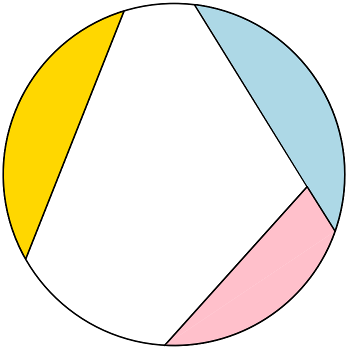
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$

using "ball carving" (similar to (Karger, Motwani, Sudan 1998))



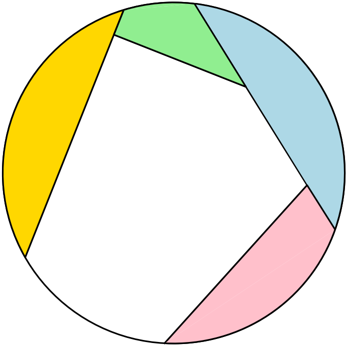
The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$
using "ball carving" (similar to (Karger, Motwani, Sudan 1998))



The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$
using "ball carving" (similar to (Karger, Motwani, Sudan 1998))

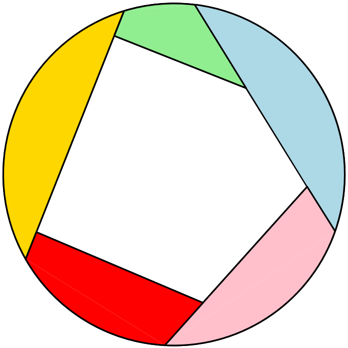


The low-diameter case

- All points and queries are on a sphere of radius $O(cr)$
- Can achieve

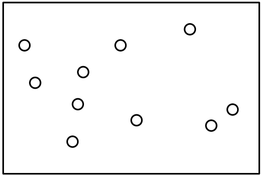
$$\rho = \frac{\ln(1/p_1)}{\ln(1/p_2)} \leq \frac{1 - \Omega(1)}{c^2}$$

using "ball carving" (similar to (Karger, Motwani, Sudan 1998))



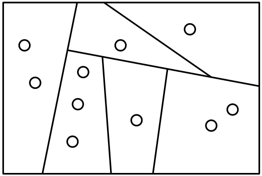
From LSH to ANN: the basic reduction

- Let $\mathcal{H}$ be a (r, cr, p_1, p_2)-sensitive family: for every $p, q \in X$
  - if $D(p, q) \leq r$, then $\mathrm{Pr}_{h \sim \mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p, q) \geq cr$, then $\mathrm{Pr}_{h \sim \mathcal{H}}[h(p) = h(q)] \leq p_2$



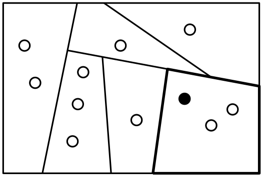
From LSH to ANN: the basic reduction

- Let $\mathcal{H}$ be a (r, cr, p_1, p_2)-sensitive family: for every $p, q \in X$
  - if $D(p, q) \leq r$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \geq p_1$;
  - if $D(p, q) \geq cr$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \leq p_2$
- Hash the dataset P using a concatenation of k functions from $\mathcal{H}$: $x \mapsto (h_1(x), h_2(x), \ldots, h_k(x))$



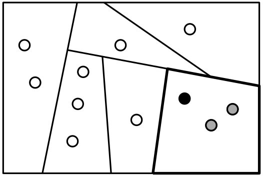
From LSH to ANN: the basic reduction

- Let $\mathcal{H}$ be a (r, cr, p_1, p_2)-sensitive family: for every $p, q \in X$
  - if $D(p, q) \le r$, then $\Pr_{h\sim\mathcal{H}}[h(p) = h(q)] \ge p_1$;
  - if $D(p, q) \ge cr$, then $\Pr_{h\sim\mathcal{H}}[h(p) = h(q)] \le p_2$
- Hash the dataset P using a concatenation of k functions from $\mathcal{H}$: $x \mapsto (h_1(x), h_2(x), \ldots, h_k(x))$
- Locate a query q and enumerate all points from the coresponding bucket



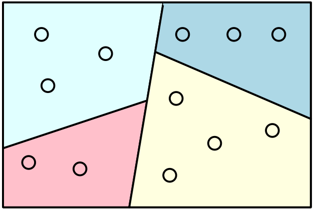
From LSH to ANN: the basic reduction

- Let $\mathcal{H}$ be a (r, cr, p_1, p_2)-sensitive family: for every $p, q \in X$
  - if $D(p,q) \le r$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \ge p_1$;
  - if $D(p,q) \ge cr$, then $\Pr_{h \sim \mathcal{H}}[h(p) = h(q)] \le p_2$
- Hash the dataset P using a concatenation of k functions from $\mathcal{H}$: $x \mapsto (h_1(x), h_2(x), \ldots, h_k(x))$
- Locate a query q and enumerate all points from the coresponding bucket
- The optimal choice of k leads to the need in n^ρ independent hash tables
- Overall: $n^{1+\rho}$ space, n^ρ query time



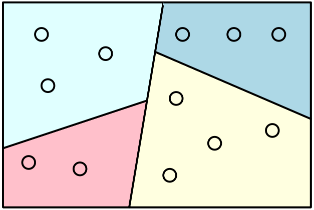
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)



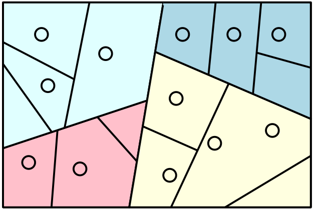
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)



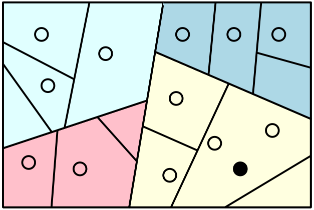
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer



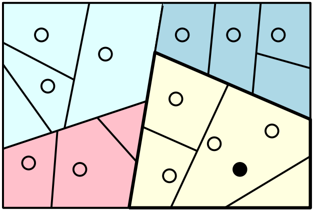
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer
- "Outer" (data-independent, $\rho \leq 1/c^2$) + "inner" (data-dependent, "low-diameter" family) hash tables



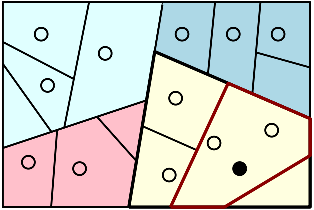
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer
- "Outer" (data-independent, $\rho \leq 1/c^2$) + "inner" (data-dependent, "low-diameter" family) hash tables



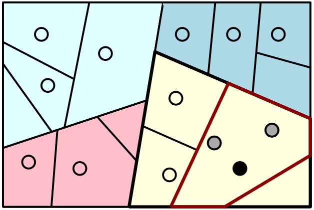
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer
- "Outer" (data-independent, $\rho \leq 1/c^2$) + "inner" (data-dependent, "low-diameter" family) hash tables



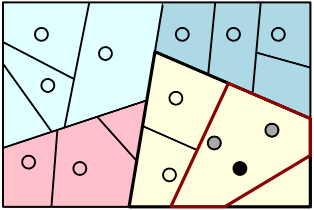
From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer
- "Outer" (data-independent, $\rho \leq 1/c^2$) + "inner" (data-dependent, "low-diameter" family) hash tables



From LSH to ANN: two-level hashing

- Partition space somewhat coarsely (using smaller k than before)
- Argue that every part has a low diameter (aim at $O(cr)$)
- Use the better family for the low-diameter case to partition space finer
- "Outer" (data-independent, $\rho \le 1/c^2$) + "inner" (data-dependent, "low-diameter" family) hash tables
- Get $\rho \le (1 - \Omega(1))/c^2$, since in inner tables we get better relation between p_1 and p_2!



Reflections

Why is this family data-dependent?

- Use a point from P as a center for an inner hash table
- If $q \in X$ is far from the center of the outer bin, then we can not handle it (but we do not care about this case)

Smallest enclosing balls

- Jung's theorem: any set of *diameter* D lies in a ball of *radius* $D/\sqrt{2}$
- For each bin find a smallest enclosing ball and hash wrt its center

Smallest enclosing balls

- Jung's theorem: any set of *diameter* D lies in a ball of *radius* $D/\sqrt{2}$
- For each bin find a smallest enclosing ball and hash wrt its center
- Careful analysis leads to

$$\rho \leq \frac{7}{8c^2} + O\left(\frac{1}{c^3}\right) + o_c(1)$$

Extensions

- Can embed ℓ_1 into ℓ_2-squared, which gives an algorithm with

$$\rho \leq \frac{7}{8c} + O\left(\frac{1}{c^{3/2}}\right) + o_c(1)$$

for ℓ_1 (in particular, Hamming distance for binary strings)

Extensions

- Can embed ℓ_1 into ℓ_2-squared, which gives an algorithm with

$$\rho \leq \frac{7}{8c} + O\left(\frac{1}{c^{3/2}}\right) + o_c(1)$$

for ℓ_1 (in particular, Hamming distance for binary strings)

- Instead of two-level hashing can consider many levels; preliminary computations give

$$\rho \leq \frac{1}{2c^2 \ln 2} + O\left(\frac{1}{c^3}\right) + o_c(1)$$

for the Euclidean case (and the similar result for ℓ_1 and Hamming)

Extensions

- Can embed ℓ_1 into ℓ_2-squared, which gives an algorithm with

$$\rho \leq \frac{7}{8c} + O\left(\frac{1}{c^{3/2}}\right) + o_c(1)$$

  for ℓ_1 (in particular, Hamming distance for binary strings)
- Instead of two-level hashing can consider many levels; preliminary computations give

$$\rho \leq \frac{1}{2c^2 \ln 2} + O\left(\frac{1}{c^3}\right) + o_c(1)$$

  for the Euclidean case (and the similar result for ℓ_1 and Hamming)
- Using this multilevel partitioning can improve known constructions for spanners for subsets of ℓ_1 and ℓ_2
  (upon (Har-Peled, Indyk, Sidiropoulos 2013))

Conclusions and open problems

- Able to overcome the LSH barrier for the case of ℓ_1 and ℓ_2 using data-dependent hashing
- Can one improve our bounds?

Conclusions and open problems

- Able to overcome the LSH barrier for the case of ℓ_1 and ℓ_2 using data-dependent hashing
- Can one improve our bounds?
- For a certain random instance can achieve $1/(2c)$ and $1/(2c^2)$, which is tight for the data-dependent hashing by (Motwani, Naor, Panigrahy 2007)

Conclusions and open problems

- Able to overcome the LSH barrier for the case of ℓ_1 and ℓ_2 using data-dependent hashing
- Can one improve our bounds?
- For a certain random instance can achieve $1/(2c)$ and $1/(2c^2)$, which is tight for the data-dependent hashing by (Motwani, Naor, Panigrahy 2007)
- Can one get these exponents for the general case?

Conclusions and open problems

- Able to overcome the LSH barrier for the case of ℓ_1 and ℓ_2 using data-dependent hashing
- Can one improve our bounds?
- For a certain random instance can achieve $1/(2c)$ and $1/(2c^2)$, which is tight for the data-dependent hashing by (Motwani, Naor, Panigrahy 2007)
- Can one get these exponents for the general case?
- Can one improve the bound for this random instance further? (Looks hard!)

Thank you!