Innovations in Computer Science 2011

Optimal Lower Bounds for Locality Sensitive Hashing (except when q is tiny)

Ryan O'Donnell[1] Yi Wu[3] Yuan Zhou[2]

[1]Computer Science Department, Carnegie Mellon University, Pittsburgh, PA. Supported by NSF grants CCF-0747250 and CCF-0915893, BSF grant 2008477, and Sloan and Okawa fellowships

[2]Computer Science Department, Carnegie Mellon University, Pittsburgh, PA

[3]IBM Almaden Research, San Jose, CA

odonnell@cs.cmu.edu wuyi@us.ibm.com yuanzhou@cs.cmu.edu

Abstract: We study lower bounds for Locality Sensitive Hashing (LSH) in the strongest setting: point sets in $\{0,1\}^d$ under the Hamming distance. Recall that $\mathcal{H}$ is said to be an (r, cr, p, q)-sensitive hash family if all pairs $x, y \in \{0,1\}^d$ with $\mathrm{dist}(x,y) \leqslant r$ have probability at least p of collision under a randomly chosen $h \in \mathcal{H}$, whereas all pairs $x, y \in \{0,1\}^d$ with $\mathrm{dist}(x,y) \geqslant cr$ have probability at most q of collision. Typically, one considers $d \to \infty$, with $c > 1$ fixed and q bounded away from 0.

For its applications to approximate nearest neighbor search in high dimensions, the quality of an LSH family $\mathcal{H}$ is governed by how small its "ρ parameter" $\rho = \ln(1/p)/\ln(1/q)$ is as a function of the parameter c. The seminal paper of Indyk and Motwani showed that for each $c \geqslant 1$, the extremely simple family $\mathcal{H} = \{x \mapsto x_i : i \in [d]\}$ achieves $\rho \leqslant 1/c$. The only known lower bound, due to Motwani, Naor, and Panigrahy, is that ρ must be at least $(e^{1/c} - 1)/(e^{1/c} + 1) \geqslant .46/c$ (minus $o_d(1)$). The contribution of our paper is twofold:

1. We show the "optimal" lower bound for ρ: it must be at least $1/c$ (minus $o_d(1)$). Our proof is very simple, following almost immediately from the observation that the noise stability of a boolean function at time t is a log-convex function of t.

2. We raise and discuss the following issue: neither the application of LSH to nearest neighbor search, nor the known LSH lower bounds, hold as stated if the q parameter is tiny. Here "tiny" means $q = 2^{-\Theta(d)}$, a parameter range we believe is natural.

Keywords: locality sensitive hashing, LSH, noise sensitivity, noise stability.

1 Locality sensitive hashing

Locality Sensitive Hashing (LSH) is a widely-used algorithmic tool which brings the classic technique of hashing to geometric settings. It was introduced for general metric spaces in the seminal work of Indyk and Motwani [8]. Indyk and Motwani showed that the important problem of (approximate) nearest neighbor search can be reduced to the problem of devising good LSH families. Subsequently, numerous papers demonstrating the practical utility of solving high-dimensional nearest neighbor search problems via the LSH approach[3-5,7,17,18]. For a survey on LSH, see Andoni and Indyk [2].

We recall the basic definition from [8]:

Definition 1. *Let (X, dist) be a distance space*[1]*, and let U be any finite or countably infinite set. Let $r > 0$, $c > 1$. A probability distribution $\mathcal{H}$ over functions $h : X \to U$ is* (r, cr, p, q)- sensitive *if for all $x, y \in X$,*

$$\mathrm{dist}(x,y) \leqslant r \Rightarrow \Pr_{\boldsymbol{h} \sim \mathcal{H}}[\boldsymbol{h}(x) = \boldsymbol{h}(y)] \geqslant p,$$
$$\mathrm{dist}(x,y) \geqslant cr \Rightarrow \Pr_{\boldsymbol{h} \sim \mathcal{H}}[\boldsymbol{h}(x) = \boldsymbol{h}(y)] \leqslant q,$$

where $q < p$. We often refer to $\mathcal{H}$ as a locality sensitive hash (LSH) family *for (X, dist).*

As mentioned, the most useful application of LSH is to the *approximate near neighbor problem* in high dimensions:

Definition 2. *For a set of n points P in a metric space (X, dist), the* (r, c)-near neighbor problem *is to process the points into a data structure that*

[1] A metric space where the triangle inequality need not hold.

supports the following type of query: given a point $x \in X$, if there exists $y \in P$ with $\mathrm{dist}(x,y) \leqslant r$, the data structure should return a point $z \in P$ such that $\mathrm{dist}(x,z) \leqslant cr$.

Several important problems in computational geometry reduce to the approximate near neighbor problem, including approximate versions of nearest neighbor, furthest neighbor, close pair, minimum spanning tree, and facility location. For a short survey of these topics, see Indyk [10].

Regarding the reduction from (r,c)-near neighbor problem to LSH, it is usual (see [6,9]) to credit roughly the following theorem to [7,8]:

Theorem 1.1. *Suppose $\mathcal{H}$ is an (r, cr, p, q)-sensitive family for the metric space (X, dist). Then one can solve the (r,c)-near neighbor problem with a (randomized) data structure that uses $O(n^{1+\rho} + dn)$ space and has query time dominated by $O(n^{\rho} \log_{1/q}(n))$ hash function evaluations. (The preprocessing time is not much more than the space bound.)*

Here we are using the following:

Definition 3. *The* rho parameter *of an (r, cr, p, q)* - sensitive LSH family $\mathcal{H}$ is

$$\rho = \rho(\mathcal{H}) = \frac{\ln(1/p)}{\ln(1/q)} \in (0,1).$$

Please note that in Theorem 1.1, it is implicitly assumed [11] that q is bounded away from 0. For "subconstant" values of q, the theorem does not hold. This point is discussed further in Section 4.

Because of Theorem 1.1, there has been significant interest [2,6,13,19] in determining the smallest possible ρ that can be obtained for a given metric space and value of c. Constant factors are important here, especially for the most natural regime of c close to 1. For example, shrinking ρ by an additive .5 leads to time and space savings of $\Theta(\sqrt{n})$.

2 Pvevious work

2.1 Upper bounds

The original work of Indyk and Motwani [8] contains the following simple yet strong result:

Theorem 2.1. *There is an LSH family $\mathcal{H}$ for $\{0,1\}^d$ under the Hamming distance which for each $c > 1$ has ρ parameter*

$$\rho(\mathcal{H}) \leqslant \frac{1}{c},$$

simultaneously for each $r < d/c$.

In this theorem, the family is simply the uniform distribution over the d functions $h_i(x) = x_i$. For a given c and r, this family is obviously $(r, cr, 1-r/d, 1-cr/d)$-sensitive, whence

$$\rho(\mathcal{H}) = \frac{\ln(1/(1-r/d))}{\ln(1/(1-cr/d))} \nearrow \frac{1}{c} \quad \text{as } r/d \to 0.$$

We remark that the upper bound of $1/c$ in Theorem 2.1 becomes tight only for asymptotically small r/d. Indyk and Motwani showed that the same bound holds for the closely related "Jaccard metric" (see [8]), and also extended Theorem 2.1 to an LSH family for the metric space ℓ_1 (see also [1]).

Perhaps the most natural setting is when the metric space is the usual d-dimensional Euclidean space ℓ_2^d. Here, Andoni and Indyk [2] showed, roughly speaking, that $\rho \leqslant 1/c^2$:

Theorem 2.2. *For any $r > 0$, $c > 1$, $d \geqslant 1$, there is a sequence of LSH families $\mathcal{H}_t$ for ℓ_2^d satisfying*

$$\limsup_{t \to \infty} \rho(\mathcal{H}_t) \leqslant \frac{1}{c^2}.$$

(The complexity of evaluating a hash function $\boldsymbol{h} \sim \mathcal{H}_t$ also increases as t increases.)

For other ℓ_s distance/metric spaces, Datar, Immorlica, Indyk, and Mirrokni [6] have similarly shown:[2]

Theorem 2.3. *For any $r > 0$, $c > 1$, $d \geqslant 1$, and $0 < s < 2$, there is a sequence of LSH families $\mathcal{H}_t$ for ℓ_s^d satisfying*

$$\limsup_{t \to \infty} \rho(\mathcal{H}_t) \leqslant \max\left\{\frac{1}{c^s}, \frac{1}{c}\right\}.$$

Other practical LSH families have been suggested for the Euclidean sphere [19] and ℓ_2 [13].

2.2 Lower bounds

There is one known result on lower bounds for LSH, due to Motwani, Naor, and Panigrahy [12]:

[2] Please note that in [12,15] it is stated that [6] also improves the Indyk–Motwani $1/c$ upper bound for ℓ_1 when $c \leqslant 10$. However this is in error.

Theorem 2.4. *Fix $c > 1$, $0 < q < 1$, and consider $d \to \infty$. Then there exists some $r = r(d)$ such that for any LSH family $\mathcal{H}$ for $\{0,1\}^d$ under Hamming distance which is (r, cr, p, q)-sensitive must satisfy*

$$\rho(\mathcal{H}) \geqslant \frac{\exp(1/c) - 1}{\exp(1/c) + 1} - o_d(1).$$

The metric setting of $\{0,1\}^d$ under Hamming distance is the most powerful setting for lower bounds; as Motwani, Naor, and Panigrahy note, one can immediately deduce a lower bound of

$$\frac{\exp(1/c^s) - 1}{\exp(1/c^s) + 1} - o_d(1)$$

for the setting of ℓ_s^d. This is simply because $\|x-y\|_s = \|x-y\|_1^{1/s}$ when $x, y \in \{0,1\}^d$.

As $c \to \infty$, the lower bound in Theorem 2.4 approaches $\frac{1}{2c}$. This is a factor of 2 away from the upper bound of Indyk and Motwani. The gap is slightly larger in the more natural regime of c close to 1; here one only has that $\rho(\mathcal{H}) \geqslant \frac{e-1}{e+1}\frac{1}{c} \approx \frac{.46}{c}$.

Note that in Theorem 2.4, the parameter q is fixed *before* one lets d tend to ∞; i.e., q is assumed to be at least a "constant". Even though this is the same assumption implicitly made in the application of LSH to near-neighbors (Theorem 1.1), we feel it is not completely satisfactory. In fact, as stated in [12], Theorem 2.4 still holds so long as $q \geqslant 2^{-o(d)}$. Our new lower bound for LSH also holds for this range of q. But we believe the most satisfactory lower bound would hold even for "tiny" q, meaning $q = 2^{-\Theta(d)}$. This point is discussed further in Section 4.

We close by mentioning the recent work of Panigrahy, Talwar, and Wieder [16] which obtains a time/space lower bound for the (r, c)-near neighbor problem itself in several metric space settings, including $\{0,1\}^d$ under Hamming distance, and ℓ_2.

3 Our result

In this work, we improve on Theorem 2.4 by obtaining a sharp lower bound of $\frac{1}{c} - o_d(1)$ for every $c > 1$. This dependence on c is optimal, by the upper bound of Indyk and Motwani. The precise statement of our result is as follows:

Theorem 3.1. *Fix $d \in \mathbb{N}$, $1 < c < \infty$, and $0 < q < 1$. Then for a certain choice of $0 < \tau < 1$, any $(\tau d, c\tau d, p, q)$-sensitive hash family $\mathcal{H}$ for $\{0,1\}^d$ under Hamming distance must satisfy*

$$\rho(\mathcal{H}) \geqslant \frac{1}{c} - \widetilde{O}\left(\frac{\ln(2/q)}{d}\right)^{1/3}. \tag{1}$$

Here, the precise meaning of the $\widetilde{O}(\cdot)$ expression is

$$K \cdot \frac{\ln(2/q)}{d} \cdot \ln\left(\frac{d}{\ln(2/q)}\right),$$

where K is a universal constant, and we assume $d/\ln(2/q) \geqslant 2$, say.

As mentioned, the lower bound is only of the form $\frac{1}{c} - o_d(1)$ under the assumption that $q \geqslant 2^{-o(d)}$. For q of the form $2^{-d/B}$ for a large constant B, the bound (1) still gives some useful information.

As with the Motwani–Naor–Panigrahy result, because our lower bound is for $\{0,1\}^d$ we may immediately conclude:

Corollary 3.2. *Theorem 3.1 also holds for LSH families for the distance space ℓ_s, $0 < s < \infty$, with the lower bound $1/c^s$ replacing $1/c$.*

This lower bound matches the known upper bounds for Euclidean space $s = 2$ ([2]) and $0 < s \leqslant 1$ ([6]). It seems reasonable to conjecture that it is also tight at least for $1 < s < 2$.

Finally, the lower bound in Theorem 3.1 also holds for the Jaccard distance on sets, matching the upper bound of Indyk and Motwani [8]. We explain why this is true in Section [3.2], although we omit the very minor necessary changes to the proof details.

3.1 Noise stability

Our proof of Theorem 3.1 requires some facts about boolean *noise stability*. We begin by recalling some basics of the analysis of boolean functions.

Definition 4. *For $0 \leqslant \epsilon < 1$, we say that $(\boldsymbol{x}, \boldsymbol{y})$ are $(1-\epsilon)$-*correlated random strings *in $\{0,1\}^d$ if $\boldsymbol{x}$ is chosen uniformly at random and $\boldsymbol{y}$ is formed by rerandomizing each coordinate of $\boldsymbol{x}$ independently with probability ϵ.*

Definition 5. *Given $f : \{0,1\}^d \to \mathbb{R}$, the* noise stability of f at $(1-\epsilon)$ *is defined to be*

$$\mathbb{S}_f(1-\epsilon) = \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \\ (1-\epsilon)\text{-correlated}}} [f(\boldsymbol{x}) f(\boldsymbol{y})].$$

We can extend the definition to functions $f : \{0,1\}^d \to \mathbb{R}^U$ *via*

$$\mathbb{S}_f(1-\epsilon) = \mathop{\mathbf{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \\ (1-\epsilon)\text{-correlated}}} [\langle f(\boldsymbol{x}), f(\boldsymbol{y})\rangle],$$

where $< w, z >= \sum_{i \in U} w_i z_i$ *is the usual inner product.*[3]

Proposition 3.3. *Let* $f : \{0,1\}^d \to \mathbb{R}^U$ *and write* $\widehat{f}(S)$ *for the usual Fourier coefficient of* f *associated with* $S \subseteq [d]$*; i.e.,*

$$\widehat{f}(S) = \frac{1}{2^d} \sum_{x \in \{0,1\}^d} f(x) \prod_{i \in S} (-1)^{x_i} \in \mathbb{R}^U.$$

Then

$$\mathbb{S}_f(1-\epsilon) = \sum_{S \subseteq [d]} \|\widehat{f}(S)\|_2^2 (1-\epsilon)^{|S|}.$$

(This formula is standard when f has range $\mathbb{R}$; see, e.g., [14]. The case when f has range $\mathbb{R}^U$ follows by repeating the standard proof.)

We are particularly interested in hash functions $h : \{0,1\}^d \to U$; we view these also as functions $\{0,1\}^d \to \mathbb{R}^U$ by identifying $i \in U$ with the vector $e_i \in \mathbb{R}^U$, which has a 1 in the ith coordinate and a 0 in all other coordinates. Under this identification, $< h(x), h(y) >$ becomes the 0-1 indicator of the event $h(x) = h(y)$. Hence for a fixed hash function h,

$$\mathbb{S}_h(1-\epsilon) = \mathop{\mathbf{Pr}}_{\substack{(\boldsymbol{x},\boldsymbol{y}) \\ (1-\epsilon)\text{-correlated}}} [h(\boldsymbol{x}) = h(\boldsymbol{y})]. \tag{2}$$

We also extend the notion of noise stability to hash *families*:

Definition 6. *If* $\mathcal{H}$ *is a hash family on* $\{0,1\}^d$*, we define*

$$\mathbb{S}_{\mathcal{H}}(1-\epsilon) = \mathop{\mathbf{E}}_{\boldsymbol{h} \sim \mathcal{H}} [\mathbb{S}_{\boldsymbol{h}}(1-\epsilon)].$$

By combining this definition with equation (2) and Proposition 3.3, we immediately deduce:

Proposition 3.4. *Let* $\mathcal{H}$ *be a hash family on* $\{0,1\}^d$*. Then*

$$\begin{aligned}\mathbb{S}_{\mathcal{H}}(1-\epsilon) &= \mathop{\mathbf{Pr}}_{\substack{\boldsymbol{h} \sim \mathcal{H}, \\ (\boldsymbol{x},\boldsymbol{y}) \ (1-\epsilon)\text{-corr'd}}} [\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{y})] \\ &= \sum_{S \subseteq [d]} \mathop{\mathbf{E}}_{\boldsymbol{h} \sim \mathcal{H}} [\|\widehat{\boldsymbol{h}}(S)\|_2^2] (1-\epsilon)^{|S|}.\end{aligned}$$

[3] In the case that U is countably infinite, we require our functions f to have $\|f(x)\|_2 < \infty$ for all $x \in \{0,1\}^d$.

Finally, it is often more natural to express the parameter $(1-\epsilon)$ as $(1-\epsilon) = e^{-t}$, where $t \in [0,\infty)$ is a "time" parameter. Here we think of a $(1-\epsilon)$-correlated pair $(\boldsymbol{x},\boldsymbol{y})$ as taking $\boldsymbol{x}$ to be uniformly random and $\boldsymbol{y}$ to be the string that results from running the standard continuous-time Markov Chain on $\{0,1\}^d$, starting from $\boldsymbol{x}$, for time td. We make the following definition:

Definition 7. *For* $t \in [0,\infty)$*, we define* $\mathbb{K}_h(t) = \mathbb{S}_h(e^{-t})$*, and we similarly define* $\mathbb{K}_{\mathcal{H}}(t)$*.*

3.2 The proof, modulo some tedious calculations

We now present the essence of our proof of Theorem 3.1. It will be quite simple to see how it gives a lower bound of the form $\frac{1}{c} - o_d(1)$ (assuming q is not tiny). Some very tedious calculations (Chernoff bounds, elementary inequalities, etc.) are needed to get the precise statement given in Theorem 3.1; the formal proof is therefore deferred to Appendix A.

Let $\mathcal{H}$ be a hash family on $\{0,1\}^d$, and let us consider

$$\mathbb{K}_{\mathcal{H}}(t) = \mathop{\mathbf{Pr}}_{\substack{\boldsymbol{h} \sim \mathcal{H}, \\ (\boldsymbol{x},\boldsymbol{y}) \ e^{-t}\text{-corr'd}}} [\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{h}(\boldsymbol{y})]. \tag{3}$$

Let us suppose that t is very small, in which case $e^{-t} \approx 1-t$. When $(\boldsymbol{x},\boldsymbol{y})$ are $(1-t)$-correlated strings, it means that $\boldsymbol{y}$ is formed from the random string $\boldsymbol{x}$ by rerandomizing each coordinate with probability t. This is the same as flipping each coordinate with probability $t/2$. Thus if we think of d as large, a simple Chernoff bound shows that the Hamming distance $\mathrm{dist}(\boldsymbol{x},\boldsymbol{y})$ will be very close to $(t/2)d$ with overwhelming probability.[4]

Suppose now that $\mathcal{H}$ is $((t/2)d + o(d), (ct/2)d - o(d), p, q)$-sensitive, so the distance ratio is $c - o_d(1)$. In (3), regardless of $\boldsymbol{h}$ we will almost surely have $\mathrm{dist}(\boldsymbol{x},\boldsymbol{y}) \leqslant (t/2) + o(d)$; hence $\mathbb{K}_{\mathcal{H}}(t) \geqslant p - o_d(1)$. Similarly, we deduce $\mathbb{K}_{\mathcal{H}}(ct) \leqslant q + o_d(1)$. Hence, neglecting the $o_d(1)$ terms, we get

$$\rho(\mathcal{H}) = \frac{\ln(1/p)}{\ln(1/q)} \gtrsim \frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t))}{\ln(1/\mathbb{K}_{\mathcal{H}}(ct))}.$$

[4] Similarly, if we think of $\boldsymbol{x}$ and $\boldsymbol{y}$ as subsets of $[d]$, their Jaccard distance will be very close to $t/(1+t/2) \approx t$ with overwhelming probability. With this observation, one obtains our lower bound on LSH families for the Jaccard distance on sets.

We then deduce the desired lower bound of $1/c$ from the following theorem and its corollary:

Theorem 3.5. *For any hash family $\mathcal{H}$ on $\{0,1\}^d$, the function $\mathbb{K}_{\mathcal{H}}(t)$ is log-convex in t.*

Proof. From Proposition 3.4 we have

$$\mathbb{K}_{\mathcal{H}}(t) = \sum_{S \subseteq [d]} \mathop{\mathbf{E}}_{\boldsymbol{h} \sim \mathcal{H}} [\|\widehat{\boldsymbol{h}}(S)\|_2^2] e^{-t|S|}.$$

Thus $\mathbb{K}_{\mathcal{H}}(t)$ is log-convex, being a nonnegative linear combination of log-convex functions $e^{-t|S|}$. □

Corollary 3.6. *For any hash family $\mathcal{H}$ on $\{0,1\}^d$, $t \geqslant 0$, and $c \geqslant 1$,*

$$\frac{\ln(1/\mathbb{K}_{\mathcal{H}}(t))}{\ln(1/\mathbb{K}_{\mathcal{H}}(ct))} \geqslant \frac{1}{c}.$$

Proof. By log-convexity, $\mathbb{K}_{\mathcal{H}}(t) \leqslant \mathbb{K}_{\mathcal{H}}(ct)^{1/c} \cdot \mathbb{K}_{\mathcal{H}}(0)^{1-1/c} = \mathbb{K}_{\mathcal{H}}(ct)^{1/c}$. Here we used the fact that $\mathbb{K}_{\mathcal{H}}(0) = 1$, which is immediate from the definitions because e^{-0}-correlated strings are always identical. The result follows. □

As mentioned, we give the careful proof keeping track of approximations in Appendix A. But first, we note what we view as a shortcoming of the proof: after deducing $\mathbb{K}_{\mathcal{H}}(ct) \geqslant q - o_d(1)$, we wish to "neglect" the additive $o_d(1)$ term. This requires that $o_d(1)$ indeed be negligible compared to q! Being more careful, the $o_d(1)$ arises from a Chernoff bound applied to a Binomial(d, ct) random variable, where $t > 0$ is very small. So to be more precise, the error term is of the form $\exp(-\epsilon d)$, and hence is only negligible if $q \geqslant 2^{-o(d)}$.

4 Discussion

4.1 On the reduction from LSH to near neighbor data structures

As described in Section 1, it is normally stated that the quality of an (r, cr, p, q)-sensitive LSH family $\mathcal{H}$ is governed by $\rho = \ln(1/p)/\ln(1/q)$, and more specifically that $\mathcal{H}$ can be used to solve the (r, c)-near neighbor problem with roughly $O(n^{1+\rho})$ space and query time $O(n^{\rho})$. However, this involves the implicit assumption that q is bounded away from 0.

It is easy to see that *some* lower bound on q is essential. Indeed, for any (finite, say) distance space (X, dist) there is a trivially "optimal" LSH family for any r and c: For each pair $x, y \in X$ with $\mathrm{dist}(x, y) \leqslant r$, define $h_{x,y}$ by setting $h_{x,y}(x) = h_{x,y}(y) = 0$ and letting $h_{x,y}(z)$ have distinct positive values for all $z \neq x, y$. If $\mathcal{H}$ is the uniform distribution over all such $h_{x,y}$, then $p > 0$ and $q = 0$, leading to $\rho(\mathcal{H}) = 0$.

To see why this trivial solution is not useful, and what lower bound on q is desirable, we recall some aspects of the Indyk–Motwani reduction from LSH families to (r, c)-near neighbor data structures. Suppose one wishes to build an (r, c)-near neighbor data structure for an n-point subset P of the metric space (X, dist). The first step in [8] is to apply the following:

1) Powering Construction:

Given an (r, cr, p, q)-sensitive family $\mathcal{H}$ of functions $X \to U$ and a positive integer k, we define the family $\mathcal{H}^{\otimes k}$ by drawing $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_k$ independently from $\mathcal{H}$ and forming the function $\boldsymbol{h} : X \to U^k$, $\boldsymbol{h}(x) = (\boldsymbol{h}_1(x), \ldots, \boldsymbol{h}_k(x))$. It is easy to check that $\mathcal{H}^{\otimes k}$ is (r, cr, p^k, q^k)-sensitive.

Indyk and Motwani show that if one has an (r, cr, p', q')-sensitive hash family with $q' \leqslant 1/n$, then one can obtain a (r, c)-near neighbor data structure with space roughly $O(n/p')$ and query time roughly $O(1/p')$. Thus given an arbitrary (r, cr, p, q)-sensitive family $\mathcal{H}$, Indyk and Motwani suggest using the Powering Construction with $k = \log_{1/q}(n)$. The resulting $\mathcal{H}^{\otimes k}$ is $(r, cr, p', 1/n)$-sensitive, with $p' = p^k = n^{-\rho}$, yielding an $O(n^{1+\rho})$ space, $O(n^{\rho})$ time data structure.

However this argument makes sense only if k is a positive integer. For example, with the trivially "optimal" LSH family, we have $q = 0$ and thus $k = -\infty$. Indeed, whenever $q \leqslant 1/n$ to begin with, one doesn't get $O(n^{1+\rho})$ space and $O(n^{\rho})$ time, one simply gets $O(n/p)$ space and $O(1/p)$ time. For example, a hypothetical LSH family with $p = 1/n^{.5}$ and $q = 1/n^{1.5}$ has $\rho = 1/3$ but only yields an $O(n^{1.5})$ space, $O(n^{.5})$ time near neighbor data structure.

The assumption $q > 1/n$ is still not enough for the deduction in Theorem 1.1 to hold precisely. The reason is that the Indyk–Motwani choice of k may not be an integer. For example, suppose we design an (r, cr, p, q)-sensitive family $\mathcal{H}$ with $p = 1/n^{.15}$ and $q = 1/n^{.3}$. Then $\rho = .5$. However, we cannot actually get an $O(n^{1.5})$ space, $O(n^{.5})$ time data structure from this $\mathcal{H}$. The reason is that to get $q^k \leqslant 1/n$, we need to take $k = 4$. Then $p^k = 1/n^{.6}$, so we only get an $O(n^{1.6})$ space, $O(n^{.6})$ time data structure.

The effect of rounding k up to the nearest integer is not completely eliminated unless one makes the assumption, implicit in Theorem 1.1, that $q \geqslant \Omega(1)$. Under the weaker assumption that $q \geqslant n^{-o(1)}$, the conclusion of Theorem 1.1 remains true up to $n^{o(1)}$ factors. To be completely precise, one should assume $q \geqslant 1/n$ and take $k = \lceil \log_{1/q}(n) \rceil$. If we then use $k \leqslant \log_{1/q}(n)+1$, the Powering Construction will yield an LSH family with $q' \leqslant 1/n$ and $p' = (n/q)^{-\rho}$. In this way, one obtains a refinement of Theorem 1.1 with no additional assumptions:

Theorem 4.1 *Suppose $\mathcal{H}$ is an (r, cr, p, q)-sensitive family for the metric space (X, dist). Then for n-point subsets of X (and assuming $q \geqslant 1/n$), one can solve the (r, c)-near neighbor problem with a (randomized) data structure that uses $n \cdot O((n/q)^\rho + d)$ space and has query time dominated by $O((n/q)^\rho \log_{1/q}(n))$ hash function evaluations.*

4.2 On assuming q is not tiny

Let us return from the near-neighbor problem to the study of locality sensitive hashing itself. Because of the "trivial" LSH family, it is essential to impose some kind of lower bound on how small the parameter q is allowed to be. Motwani, Naor, and Panigrahy carry out their lower bound for LSH families on $\{0,1\}^d$ under the assumption that $q \geqslant \Omega(1)$, but also note that it goes through assuming $q \geqslant 2^{-o(d)}$. Our main result, Theorem 3.1, is also best when $q \geqslant 2^{-o(d)}$, and is only nontrivial assuming $q \geqslant 2^{-d/B}$ for a sufficiently large constant B.

One may ask what the "correct" lower bound assumed on q should be. For the Indyk–Motwani application to (r, c)-near neighbor data structures, the answer seems obvious: "$1/n$". Indeed, since the Indyk–Motwani reduction immediately uses Powering to reduce the q parameter down to $1/n$, the most meaningful LSH lower bounds would simply involve fixing $q = 1/n$ and trying to lower bound p.

There is an obvious catch here, though, which is that in the definition of LSH, there *is no notion of "n"*! Still, in settings such as $\{0,1\}^d$ which have a notion of dimension, d, it seems reasonable to think that applications will have $n = 2^{\Theta(d)}$. In this case, to maintain the Indyk–Motwani Theorem 4.1 up to $n^{o(1)}$ factors one would require $q \geqslant 2^{-o(d)}$. This is precisely the assumption that this paper and the Motwani–Naor–Panigrahy paper have made. Still, we believe that the most compelling kind of LSH lower bound for $\{0,1\}^d$ would be nontrivial even for $q = 2^{-d/b}$ with a "medium" constant b, say $b = 10$. We currently do not have such a lower bound.

Acknowledgments

The authors would like to thank Alexandr Andoni, Piotr Indyk, Assaf Naor, and Kunal Talwar for helpful discussions.

References

[1] A. Andoni and P. Indyk. Efficient algorithms for substring near neighbor problem. In *Proceedings of the 17-th Ann. ACM-SIAM Symposium on Discrete Algorithm*, page 1212. ACM, 2006.

[2] A. Andoni and P. Indyk. Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. *Communications of the ACM*, 51(1):117–122, 2008.

[3] J. Buhler. Efficient large-scale sequence comparison by locality-sensitive hashing. *Bioinformatics*, 17(5):419–428, 2001.

[4] E. Cohen, M. Datar, S. Fujiwara, A. Gionis, P. Indyk, R. Motwani, J.D. Ullman, and C. Yang. Finding interesting associations without support pruning. *IEEE Transactions on Knowledge and Data Engineering*, 13(1):64–78, 2002.

[5] A.S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th Intl. Conf. on World Wide Web*, pages 271–280. ACM, 2007.

[6] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni. Locality-sensitive hashing scheme based on p-stable distributions. In *SCG '04: Proceedings of the 20th Ann. Symposium on Computational Geometry*, pages 253–262, New York, NY, USA, 2004. ACM.

[7] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *Proceedings of the 25-th International Conference on Very Large Data Bases*, 1999.

[8] P. Indyk and R. Motwani. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the 13-th Ann. ACM Symposium on Theory of Computing*, pages 604–613. ACM New York, NY, USA, 1998.

[9] P. Indyk. *High-dimensional computational geometry.* PhD thesis, Stanford University, 2001.

[10] P. Indyk. Nearest neighbors in high-dimensional spaces. *Handbook of Discrete and Computational Geometry*, pages 877–892, 2004.

[11] P. Indyk. Personal communication, 2009.

[12] R. Motwani, A. Naor, and R. Panigrahi. Lower bounds on locality sensitive hashing. *SIAM Journal on Discrete Mathematics*, 21(4):930–935, 2007.

[13] T. Neylon. A locality-sensitive hash for real vectors. In *Proceedings of the 21st Ann. ACM-SIAM Symposium on Discrete Algorithms*, pages 1179–1189, 2010.

[14] R. O'Donnell. *Computational applications of noise sensitivity.* PhD thesis, Massachusetts Institute of Technology, 2003.

[15] R. Panigrahy. Entropy based nearest neighbor search in high dimensions. In *Proceedings of the 17-th annual ACM-SIAM Symposium on Discrete Algorithm*, pages 1186–1195. ACM, 2006.

[16] R. Panigrahy, K. Talwar, and U. Wieder. A geometric approach to lower bounds for approximate near-neighbor search and partial match. In *Proceedings of the 49-th annual IEEE Symposium on Foundations of Computer Science*, pages 414–423. IEEE Computer Society, 2008.

[17] D. Ravichandran, P. Pantel, and E. Hovy. Randomized algorithms and NLP: Using locality sensitive hash function for high speed noun clustering. In *Proceedings of the 43rd Ann. Meeting on Association for Computational Linguistics*, pages 622–629. Association for Computational Linguistics, 2005.

[18] G. Shakhnarovich, P. Viola, and T. Darrell. Fast pose estimation with parameter-sensitive hashing. In *Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 750. Citeseer, 2003.

[19] K. Terasawa and Y. Tanaka. Spherical LSH for approximate nearest neighbor search on unit hypersphere. *Lecture Notes in Computer Science*, 4619:27, 2007.

A Proof details

We require the following lemma, whose proof follows easily from Proposition 3.4 and the definition of hash family sensitivity:

Lemma A.1. *Let $\mathcal{H}$ be an (r, cr, p, q)-sensitive hash family on $\{0,1\}^d$ and suppose $(\boldsymbol{x}, \boldsymbol{y})$ is a pair of e^{-u}-correlated random strings. Then*

$$p(1 - \mathbf{Pr}[\mathrm{dist}(\boldsymbol{x}, \boldsymbol{y}) > r]) \leqslant \mathbb{K}_{\mathcal{H}}(u) \leqslant q + \mathbf{Pr}[\mathrm{dist}(\boldsymbol{x}, \boldsymbol{y}) < cr].$$

We now prove Theorem 3.1, which for convenience we slightly rephrase as follows:

Theorem A.2. *Fix $d \in \mathbb{N}$, $1 < c < \infty$, and $0 < q < 1$. Then for a certain choice of $0 < \epsilon < 1$, any $((\epsilon/c)d, \epsilon d, p, q)$-sensitive hash family for $\{0,1\}^d$ under Hamming distance must satisfy*

$$\rho = \frac{\ln(1/p)}{\ln(1/q)} \geqslant \frac{1}{c} - K \cdot \lambda(d,q)^{1/3},$$

where K is a universal constant,

$$\lambda(d,q) = \frac{\ln(2/q)}{d} \ln\left(\frac{d}{\ln(2/q)}\right),$$

and we assume $d/\ln(2/q) \geqslant 2$, say.

Proof. Let $0 < \Delta = \Delta(c,d,q) < .005$ be a small quantity to be chosen later, and let $\epsilon = .005\Delta$. Suppose that $\mathcal{H}$ is an $((\epsilon/c)d, \epsilon d, p, q)$-sensitive hash family for $\{0,1\}^d$. Our goal is to lower bound $\rho = \ln(1/p)/\ln(1/q)$. By the Powering Construction we may assume that $q \leqslant 1/e$, and hence will use $\ln(1/q) \geqslant 1$ without further comment. Define also $t = 2\epsilon(1 + \Delta/2)$ and $c' = c(1+\Delta)$.

Let $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ be $\exp(-t/c')$-correlated random strings and let $(\boldsymbol{x}_2, \boldsymbol{y}_2)$ be $\exp(-t)$-correlated random strings. Using the two bounds in Lemma A.1 separately, we have

$$\mathbb{K}_{\mathcal{H}}(t/c') \geqslant p(1 - e_1),$$
$$\mathbb{K}_{\mathcal{H}}(t) \leqslant q + e_2,$$

where

$$e_1 = \mathbf{Pr}[\mathrm{dist}(\boldsymbol{x}_1, \boldsymbol{y}_1) > (\epsilon/c)d],$$
$$e_2 = \mathbf{Pr}[\mathrm{dist}(\boldsymbol{x}_2, \boldsymbol{y}_2) < \epsilon d].$$

By Corollary 3.6, we have

$$\frac{1}{c'} \leqslant \frac{\ln\Big(1/\mathbb{K}_{\mathcal{H}}(t/c')\Big)}{\ln\Big(1/\mathbb{K}_{\mathcal{H}}(t)\Big)} \leqslant \frac{\ln\left(\frac{1}{p(1-e_1)}\right)}{\ln\left(\frac{1}{q+e_2}\right)}$$
$$= \frac{\ln(1/p) + \ln(1/(1-e_1))}{\ln(1/q) + \ln(1/(1+e_2/q))}. \qquad (4)$$

We will use the following estimates:

$$\frac{1}{c'} = \frac{1}{c(1+\Delta)} \geqslant \frac{1}{c}(1-\Delta) = \frac{1}{c} - \frac{\Delta}{c}, \tag{5}$$

$$\ln(1/(1-e_1)) \leqslant 1.01e_1, \tag{6}$$

$$\begin{aligned}\ln(1/q) + \ln(1/(1+e_2/q)) &\geqslant \ln(1/q) - e_2/q \\ &= \ln(1/q)\Big(1 - \frac{e_2}{q\ln(1/q)}\Big).\end{aligned} \tag{7}$$

For (6) we made the following

$$\textbf{assumption:} \qquad e_1 \leqslant .01. \tag{8}$$

We will also ensure that the quantity in (7) is positive by making the following

$$\textbf{assumption:} \qquad e_2 < q\ln(1/q). \tag{9}$$

Substituting the three estimates (5)–(7) into (4) we obtain

$$\begin{aligned}&\frac{1}{c} - \frac{\Delta}{c} \leqslant \frac{\ln(1/p) + 1.01e_1}{\ln(1/q)\Big(1 - \frac{e_2}{q\ln(1/q)}\Big)} \\ \Rightarrow\ &\frac{\ln(1/p) + 1.01e_1}{\ln(1/q)} \\ &\qquad \geqslant \left(\frac{1}{c} - \frac{\Delta}{c}\right)\left(1 - \frac{e_2}{q\ln(1/q)}\right) \\ \Rightarrow\ &\frac{\ln(1/p)}{\ln(1/q)} \geqslant \frac{1}{c} - \frac{\Delta}{c} - \frac{e_2}{q\ln(1/q)} - \frac{1.01e_1}{\ln(1/q)}.\end{aligned}$$

Thus we have established

$$\rho \geqslant \frac{1}{c} - e, \text{where } e = \frac{\Delta}{c} + \frac{1.01e_1}{\ln(1/q)} + \frac{e_2}{q\ln(1/q)}. \tag{10}$$

We now estimate e_1 and e_2 in terms of Δ (and ϵ), after which we will choose Δ so as to minimize e. By definition, e_1 is the probability that a Binomial(d, η_1) random variable exceeds $(\epsilon/c)d$, where $\eta_1 = (1 - \exp(t/c'))/2$. Let us select δ_1 so that $(1+\delta_1)\eta_1 = \epsilon/c$. Thus

$$\begin{aligned}\delta_1 &= \frac{\epsilon}{c\eta_1} - 1 = \frac{2\epsilon/c}{1 - \exp(-t/c')} - 1 \\ &\geqslant \frac{2\epsilon/c}{t/c'} - 1 = \frac{1+\Delta}{1+\Delta/2} - 1 \geqslant .498\Delta.\end{aligned}$$

Here we used the definitions of t and c', and then the assumption $\Delta < .005$. Using a standard Chernoff bound, we conclude

$$\begin{aligned}e_1 &= \mathbf{Pr}[\text{Binomial}(d, \eta_1) > (1+\delta_1)\eta_1 d] \\ &< \exp\left(-\frac{\delta_1^2}{2+\delta_1}\eta_1 d\right) < \exp\left(-\frac{\Delta^2}{8.08}\eta_1 d\right),\end{aligned} \tag{11}$$

using the fact that $\delta^2/(2+\delta)$ is increasing in δ, and $\Delta < .005$ again. We additionally estimate

$$\begin{aligned}\eta_1 &= \frac{1 - \exp(t/c')}{2} \geqslant \frac{t/c' - (t/c')^2/2}{2} \\ &= (t/2c') - (t/2c')^2 \geqslant .99(t/2c') \\ &= .99\frac{\epsilon}{c}\left(\frac{1+\Delta/2}{1+\Delta}\right) \geqslant .98\frac{\epsilon}{c}.\end{aligned}$$

Here the second inequality used $t/2c' \leqslant .01$, which certainly holds since $t/2c' \leqslant \epsilon = .005\Delta$. The third inequality used $\Delta \leqslant .005$. Substituting this into (11) we obtain our upper bound for e_1,

$$\begin{aligned}e_1 &< \exp\left(-\frac{\Delta^2}{8.25}\frac{\epsilon}{c}d\right) \\ &= \exp\left(-\frac{.005\Delta^3}{8.25c}d\right) < \exp\left(-\frac{\Delta^3}{2000c}d\right).\end{aligned} \tag{12}$$

Our estimation of e_2 is quite similar:

$$\begin{aligned}e_2 &= \mathbf{Pr}[\text{Binomial}(d, \eta_2) < (1-\delta_2)\eta_2 d] \\ &< \exp\left(-\frac{\delta_2^2}{2}\eta_2 d\right),\end{aligned} \tag{13}$$

where $\eta_2 = (1 - \exp(-t))/2$ and δ_2 is chosen so that $(1-\delta_2)\eta_2 = \epsilon$. This entails

$$\begin{aligned}\delta_2 &= 1 - \frac{\epsilon}{\eta_2} = 1 - \frac{2\epsilon}{1 - \exp(-t)} \geqslant 1 - \frac{2\epsilon}{t - t^2/2} \\ &= 1 - \frac{1}{(t/2\epsilon) - \epsilon(t/2\epsilon)^2} \\ &= 1 - \frac{1}{(1+\Delta/2) - \epsilon(1+\Delta/2)^2}.\end{aligned}$$

This expression is the reason we were forced to take ϵ noticeably smaller than Δ. Using our specific setting $\epsilon = .005\Delta$, we conclude

$$\begin{aligned}\delta_2 &\geqslant 1 - \frac{1}{(1+\Delta/2) - \epsilon(1+\Delta/2)^2} \\ &= 1 - \frac{1}{1 + .495\Delta - .005\Delta^2 - .00125\Delta^3} \\ &\geqslant .49\Delta,\end{aligned}$$

where we used $\Delta \leqslant .005$ again. As for η_2, we can lower bound it similarly to η_1, obtaining

$$\eta_2 \geqslant .99(t/2) = .99\epsilon(1+\Delta/2) \geqslant .99\epsilon.$$

Substituting our lower bounds for δ_2 and η_2 into (13) yields

$$e_2 < \exp\left(-\frac{(.49\Delta)^2}{2} \cdot .99\epsilon d\right) < \exp\left(-\frac{\Delta^3}{2000} d\right). \tag{14}$$

Plugging our upper bounds (12), (14) for e_1, e_2 into (10) gives

$$e = \frac{\Delta}{c} + \frac{1.01 \exp(-\frac{\Delta^3}{2000c} d)}{\ln(1/q)} + \frac{\exp(-\frac{\Delta^3}{2000} d)}{q \ln(1/q)}. \tag{15}$$

Finally, we would like to choose

$$\Delta = K_1 c^{1/3} \lambda(d, q)^{1/3},$$

where K_1 is an absolute constant. For K_1 sufficiently large, this makes all three terms in the bound (15) at most

$$2K_1 \lambda(d, q)^{1/3} = \widetilde{O}\left(\frac{\ln(2/q)}{d}\right)^{1/3}.$$

This would establish the theorem.

It only remains to check whether this is a valid choice for Δ. First, we note that with this choice, assumptions (8) and (9) follow from (12) and (14) (and increasing K_1 if necessary). Second, we required that $\Delta \leqslant .005$. This may not hold. However, if it fails then we have

$$\lambda(d, q)^{1/3} > \frac{.005}{K_1 c^{1/3}}.$$

We can then trivialize the theorem by taking $K = (K_1/.005)^3$, making the claimed lower bound for ρ smaller than $1/c - 1/c^{1/3} \leqslant 0$. □