Corrigendum to "Efficient similarity search and classification via rank aggregation" by Ronald Fagin, Ravi Kumar and D. Sivakumar (Proc. SIGMOD'03)

Alexandr Andoni
MIT

Ronald Fagin
IBM Almaden

Ravi Kumar
Yahoo! Research

Mihai Pătraşcu
MIT

D. Sivakumar
Google Inc.

Categories and Subject Descriptors: E.1 [**Data structures**].

General Terms: Algorithms, theory.

Keywords: Nearest neighbor, rank aggregation, score aggregation, median.

In this corrigendum, we correct an error in the paper [1]. The error was discovered by Alexandr Andoni, and the corrected theorem is due to the three authors of [1], along with Alexandr Andoni and Mihai Pătraşcu.

Theorem 4 of [1] states:

> *Let D be a collection of n points in $\mathbb{R}^d$. Let $r_1, \ldots r_m$ be random unit vectors in $\mathbb{R}^d$, where $m = \alpha\epsilon^{-2}\log n$ with α suitably chosen. Let $q \in \mathbb{R}^d$ be an arbitrary point, and define, for each i with $1 \leq i \leq m$, the ranked list L_i of the n points in D by sorting them in increasing order of their distances to the projection of q along r_i. For each element x of D, let $medrank(x) = median(L_1(x), \ldots, L_m(x))$. Let z be a member of D such that $medrank(z)$ is minimized. Then with probability at least $1-1/n$, we have $\|z-q\|_2 \leq (1+\epsilon)\|x-q\|_2$ for all $x \in D$.*

As stated, the above theorem does not hold, but a version of it holds if one replaces the median over *ranks* by a median over suitably defined *scores*. Below, we give a counterexample to the original theorem, and then present our modification to the theorem, and the resulting algorithm.

1. A COUNTEREXAMPLE

Intuitively, the above theorem does not hold in the following situation. Suppose q is the query point, p is the nearest neighbor of q, and z is at distance $(1+\epsilon)\|p-q\|_2$. For a random unit vector r, let $rank_r(p)$ denote the rank of the point p in the list L_r of the set D of points sorted by their distance to the projection of q along r. While it is true that $rank_r(p) < rank_r(z)$ holds $1/2 + \Omega(\epsilon)$ fraction of the time (over the random choice of r), we cannot infer the same for the overall median rank when taking into the consideration the other points in D. In particular, a bad dataset is one where whenever $rank_r(p) < rank_r(z)$ then about half of the time both ranks are high, but when $rank_r(p) > rank_r(z)$ the point z has very small rank and p has a high rank. Then, in the end, p will have a high rank for about 75% of the time, while z has a high rank about 25% of time. Our counterexample constructs a set with (roughly) such characteristics.

We give a specific set of $n \geq 10$ points in 2-dimensional space. Consider the following point set for very small ϵ, illustrated in Fig. 1:

- point $q = (0,0)$, the query;
- point $p = (0,1)$, the nearest neighbor;
- point $z = (1+\epsilon, 0)$, the false nearest neighbor;
- a set H of $\frac{n-3}{2}$ points all at distance $(1+\epsilon)^2$ from q, specifically at $h = (1+\epsilon)^2 \cdot (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$;
- a set S of the same size as H, namely $\frac{n-3}{2}$ points, all situated at $s = (1+\epsilon)^2 \cdot (1,0)$.

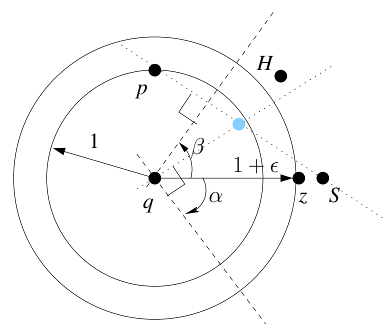

Figure 1: The pointset for our counterexample, where q is the query and p is the nearest neighbor. The grey point is the midpoint of the segment ps.

Let r be a random unit vector in $\mathbb{R}^2$, and let $L_r, rank_r(x)$ be as defined earlier. Then we have the following two claims. Below, $\Pr_r$ denotes probability over the random choice of r.

Claim 1.1. $\Pr_r[rank_r(z) \leq 2] \geq 1/2 + \Omega(\epsilon)$.

Claim 1.1 follows immediately from Lemma 3 of [1].

Claim 1.2. $\Pr_r[rank_r(p) > |H|] \geq 1/2 + \Omega(1)$.

We prove Claim 1.2 next. It is sufficient to consider r's with non-negative x coordinate (since r and $-r$ yield the same list L_r), and identify r's by their angle γ_r with the x axis. First, we note that $rank_r(p) \leq rank_r(s)$ iff $\gamma_r \in [\alpha, \beta]$, where α is angle formed by the perpendicular to the line

SIGMOD'08, June 9–12, 2008, Vancouver, BC, Canada.
ACM 978-1-60558-102-6/08/06.

Preprocessing. Input: a set D of points from $\mathbb{R}^d$, $|D| = n$, and $\epsilon > 0$.

1. Choose $k = O(\frac{\log n}{\epsilon^2})$ vectors $r_i \in \mathbb{R}^d$, $i = 1 \ldots k$, where each coordinate of r_i is drawn from a Gaussian $N(0,1)$ distribution. Vectors r_i identify some random projections.
2. Construct k lists, where the i^{th} list contains all the points $p \in D$ sorted according to the value $p \cdot r_i$.

Query. Input: a query point $q \in \mathbb{R}^d$.

1. For fixed i and $p \in D$, define $score_{r_i}(p) = p \cdot r_i - q \cdot r_i$.
2. Return the point $p^* \in D$ that minimizes $median_{i \in [k]}\{|score_{r_i}(p^*)|\}$.

Figure 2: The new algorithm for $1+\epsilon$ nearest neighbor data structure.

connecting q to the midpoint of the segment ps, and β is the angle formed by the perpendicular to ps. We can estimate α and β as follows, using the convention that $p = (p_x, p_y)$ and $s = (s_x, s_y)$. Since the midpoint of the segment ps is the point $(\frac{p_x+s_x}{2}, \frac{p_y+s_y}{2})$, we obtain that

$$\alpha = \arctan \tfrac{p_y+s_y}{p_x+s_x} - \pi/2 = \arctan \tfrac{1}{(1+\epsilon)^2} - \pi/2 = -\pi/4 - \Theta(\epsilon).$$

Similarly, the slope of the line ps is $\frac{p_y-s_y}{p_x-s_x}$ and thus the angle to its perpendicular line is

$$\beta = \arctan \tfrac{s_x-p_x}{p_y-s_y} = \arctan(1+\epsilon)^2 = \pi/4 + \Theta(\epsilon).$$

Thus, if $\gamma_r \notin [\alpha, \beta]$, then $rank_r(p) > rank_r(s)$, and so $rank_r(p) > |S| = |H|$.

Moreover, as we will see, if the angle of r is around $-\pi/4$, then $rank_r(p) > rank_r(h)$. Indeed, consider r with angle $\gamma_r \in [-\pi/4-\pi/16, -\pi/4+\pi/16]$ to the x axis. Then, $|p \cdot r| = |0 \cdot \cos\gamma_r + \sin\gamma_r| > 0.5$ and $|h \cdot r| = |(1+\epsilon)^2 \frac{1}{\sqrt{2}} \cdot (\sin\gamma_r + \cos\gamma_r)| < 0.2(1+\epsilon)^2$. Thus, when $\gamma_r \in [-\pi/4-\pi/16, -\pi/4+\pi/16]$, we have that $rank_r(p) > |H|$.

Combining the two ranges of the angle of r, we conclude that if the angle of r is in the range $(-\pi/2, -\pi/4+\pi/16)$ or $(\beta, \pi/2)$, we have $rank_r(p) > |H|$. This happens with probability at least $\frac{\pi/4+\pi/16+\pi/4-\Theta(\epsilon)}{\pi} = 1/2 + 1/16 - \Theta(\epsilon)$.

Standard high concentration bounds yield, with high probability, that $medrank(z) \leq 2$ and $medrank(p) \geq |H|$ and thus $medrank(z) < medrank(p)$. For completeness, we include one such lemma, due to Indyk:

Lemma 1.3 (cf. [2], Claim 2). *Let $\mathcal{D}$ be a distribution on $\mathbb{R}$ and F be its cumulative distribution function. Then, for $\epsilon, \delta > 0$ and some $k = O(\frac{\log 1/\delta}{\epsilon^2})$, if $X_1 \ldots X_k$ are iid from $\mathcal{D}$, then $X = median\{X_1, \ldots X_k\}$ satisfies* $\Pr[F(X) \in (1/2-\epsilon, 1/2+\epsilon)] \geq 1-\delta$.

2. A NEW ALGORITHM

To correct the theorem, we propose to replace $rank_r(x)$ by $|x \cdot r - q \cdot r|$, which we refer to as (the absolute value of) a *score*, and, consequently, we replace *medrank* by an alternative function $medscore(x) = median_i(|x \cdot r_i - q \cdot r_i|)$.

The rest of the algorithm remains unchanged. The resulting algorithm is presented in Fig. 2. Next, we show that we obtain a $1+\epsilon$ nearest neighbor data structure.

Theorem 2.1. *The algorithm from Figure 2 returns a $1+\epsilon$ nearest neighbor of q with probability at least $1-1/n$.*

Proof. Fix some p and let $\Delta = \|p-q\|_2$. For each $i \in [k]$, we have that $score_{r_i}(p) = (p-q) \cdot r_i$ is distributed as $N(0, \Delta^2)$, the normal distribution with standard deviation Δ. We will once again use Lemma 1.3 for estimating the median of iid samples.

Let $M_p = median_{i \in [k]}\{|score_{r_i}(p)|\}$. We apply Lemma 1.3 with $X_i = |score_{r_i}(p)|$ which is distributed as the absolute value of the Gaussian $N(0, \Delta^2)$ and thus has cumulative distribution function $F(x) = \mathrm{erf}(x/\Delta) = \frac{2}{\sqrt{\pi}} \int_0^{x/\Delta} e^{-t^2} dt$. We then conclude that, setting $\delta = 1/n^2$, we have $F(M_p) \in (1/2 - O(\epsilon), 1/2 + O(\epsilon))$ with probability at least $1 - 1/n^2$. Then, for $x^* = \Delta \cdot c$ where $c = \mathrm{erf}^{-1}(1/2)$, we have that $F(x^* - O(\epsilon) \cdot \Delta) = \frac{2}{\sqrt{\pi}} \int_0^{(x^*-O(\epsilon)\cdot\Delta)/\Delta} e^{-t^2} dt = \mathrm{erf}(c) + \frac{2}{\sqrt{\pi}} \int_c^{c-O(\epsilon)} e^{-t^2} dt = 1/2 - O(\epsilon)$ and similarly $F(x^* + O(\epsilon) \cdot \Delta) = 1/2 + O(\epsilon)$. We conclude, by the monotonicity of F, that, with probability at least $1 - 1/n^2$, we have that $M_p \in (x^* - O(\epsilon) \cdot \Delta, x^* + O(\epsilon) \cdot \Delta)$ and thus $M_p/c \in (\Delta - O(\epsilon), \Delta + O(\epsilon))$. Finally, choosing the implicit constant in k sufficiently high, we conclude that, with probability at least $1 - 1/n^2$,

$$(1-\epsilon/3)\|p-q\|_2 \leq M_p/c \leq (1+\epsilon/3)\|p-q\|_2. \quad (1)$$

By the union bound, (1) holds for all $p \in D$ with probability at least $1 - 1/n$. Now, minimizing M_p is equivalent to minimizing M_p/c, which we can bound as $\min_p M_p/c \leq \min_p (1+\epsilon/3)\|p-q\|_2 = (1+\epsilon/3)\|p^*-q\|_2$, where p^* is the nearest neighbor of q. Then, for any p with $\|p-q\|_2 > (1+\epsilon)\|p^*-q\|_2$, we have, by (1), that $M_p/c \geq (1-\epsilon/3)\|p-q\|_2 > (1-\epsilon/3)(1+\epsilon)\|p^*-q\| \geq \frac{(1-\epsilon/3)(1+\epsilon)}{1+\epsilon/3} \min_p M_p/c > \min_p M_p/c$. Thus, step (2) of the query algorithm returns a point p with $\|p-q\|_2 \leq (1+\epsilon)\|p^*-q\|_2$, with probability $\geq 1-1/n$. □

We note that, for Step 2 of the query algorithm, we can use also other aggregation functions instead of the median function. In particular, if we use the ℓ_2 norm of the *score* vector instead of the median, then the same theorem as above holds, implied by the Johnson–Lindenstrauss lemma [3]. Furthermore, if use the ℓ_1 norm of the *score* vector, then again the same theorem as above holds, and is implied by the ℓ_2 to ℓ_1 embedding of [4].

Finally, we note that instance-optimality claims similar to those in [1] carry over to our algorithm (except that random accesses are also required).

3. REFERENCES

[1] R. Fagin, R. Kumar, and D. Sivakumar. Efficient similarity search and classification via rank aggregation. In *Proceedings of the 2003 SIGMOD International Conference on Management of Data*, pages 301–312, 2003.

[2] P. Indyk. Stable distributions, pseudorandom generators, embeddings and data stream computation. *Journal of the ACM*, 53(3):307–323, 2006.

[3] W. Johnson and J. Lindenstrauss. Extensions of Lipshitz mapping into Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.

[4] W. Johnson and G. Schechtman. Embedding l_p^m into l_1^n. *Acta Mathematica*, 149:71–85, 1982.