It is NP-hard to verify an LSF on the sphere

Thomas Dybdahl Ahle
March 2017

A locality sensitive filter system, LSF, on a sphere is a matrix $A \in \mathbb{R}^{n \times d}$ where the rows are vectors of approximately unit length. (It could for example have Gaussian $\mathcal{N}(0, 1/d)$ elements.) The LSF can be used to create a nearest-neighbour data-structure on a set of points on the unit sphere $X \subseteq S^{d-1}$, by creating a 'bucket' $B_a$ for each row $a \in A$. For each $x \in X$ we add $x$ to $B_a$ if $\langle x, a \rangle \geq \tau$ for some constant $\tau$. We say an LSF is 'correct' for a value $r$, $0 < r < 1$, if for all $x \in X$ and $y \in S^{d-1}$ with $\langle x, y \rangle \geq r$ there is an $a \in A$ such that $\langle x, a \rangle \geq \tau$ and $\langle y, a \rangle \geq \tau$. Intuitively an LSF is correct if two points, that are close to each other, are guaranteed to fall in a shared bucket.

An important problem is whether we can verify that an $A$ is correct for a value $r$. In this note we show that such a verification is not possible in time polynomial in $n$, unless $P = NP$. In particular we show this for the case of a data structure with just a single point. That is $|X| = 1$. The approach is inspired by [1].

**Definition 1** (Problem 1: Verification). Given constants $0 < \tau < r < 1$, a vector $x \in S^{d-1}$ and a matrix $A$ with $Ax \geq \tau$, return a point $y \in S^{d-1}$ such that $Ay < \tau$ and $\langle y, x \rangle = r$.

Importantly, if an LSF is correct for $r$, the above problem should fail for any $x$. On the other hand, if the LSF is not correct, the above problem will find a $y$ that that proves it bad.

We show that the 3-Sat problem can be reduced to the verification problem.

**Definition 2** (Problem 2: 3-Sat). Given $n$ boolean variables, $x_i$, and $m$ clauses on the form $(\neg)x_i \lor (\neg)x_j \lor (\neg)x_k$, determine if there is an assignment to the variables that make all clauses true.

We will reduce 3-Sat to the verification problem with $r = 1/\sqrt{2}$, $\tau = \alpha/\sqrt{n}$, $\alpha = \sqrt{2/3}/(2-\sqrt{2})$ and $x = (1, 0, \ldots, 0) \in \mathbb{R}^{n+1}$. Other values are also possible, but these are pretty typical for the values that would be used in practice. Here $\alpha$ was chosen such that $\alpha/\sqrt{2} + 1/\sqrt{6} = \alpha < \alpha/\sqrt{2} + 3/\sqrt{6}$.

TODO: Decide whether to use $d$ or $n$.

For each clause $(\neg)x_i \lor (\neg)x_j \lor (\neg)x_k$ with $1 \leq i < j < k \leq n$ we define a row $a \in \mathbb{R}^{n+1}$. We set $a_0 = \alpha/\sqrt{d}$ and $a_i = 1/\sqrt{3}$ if $x_i$ is positive in the clause, and $a_i$ if $x_i$ is negative $(\neg)$ we set $a_i = -1/\sqrt{3}$. If $x_i$ is not the the clause, we
set \( a_i = 0 \). (Note that \( \|a\|^2 = 1 + \alpha^2/d \approx 1 \), which is similar to what it would be with gaussian values.)

We further define rows \( b_i \in \mathbb{R}^{n+1} \) for \( 1 \leq i \leq 2n \) such that \( b_{i,0} = \alpha/\sqrt{d} \), \( b_{2i,2i+1} = 1/\sqrt{3} \) and \( b_{2i+1,2i+2} = -1/\sqrt{3} \). In total we get a matrix \( A \) with \( m+2n \) rows and \( n+1 \) columns. For all \( a \) and \( b \) we have \( \langle a, x \rangle = \langle b, x \rangle = \alpha/\sqrt{d} = \tau \).

(Note we don’t quite have \( \|b\| \approx 1 \), but we could fix that by a \( \sqrt{2/3} \) coordinate and 0 coordinates on the other vectors.)

Visually the different vectors look like this:

\[
\begin{align*}
y &= (1/\sqrt{2}, \pm 1/\sqrt{2d}, \ldots) \\
x &= (1, 0, \ldots, 0) \\
a &= (\alpha/\sqrt{d}, 0, \ldots, \pm 1/\sqrt{3}, \ldots, 0) \\
b &= (\alpha/\sqrt{d}, 0, \ldots, \pm 1/\sqrt{3}, \ldots, 0)
\end{align*}
\]

**Theorem 1.** The verification problem for \( A, \tau, r, x \) will find a counter example \( y \) if and only if the 3-Sat problem is satisfiable.

**Proof.** We first show that if the clauses are all satisfiable, we can find a \( y \) as by the verification problem. Let \( x_i \in \{\text{true, false}\} \) for \( 1 \leq i \leq n \) be an assignment satisfying the clauses. We let \( y_0 = 1/\sqrt{2} \) and \( y_i = \pm 1/\sqrt{2n} \) where the sign is negative if \( x_i \) is true and positive if \( x_i \) is false.

This makes \( \|y\|^2 = 1 \) and \( \langle x, y \rangle = 1/\sqrt{2} = r \). For each \( a \) in \( A \) we have \( \langle y, a \rangle = \alpha/\sqrt{2d} + \{-3, -1, 1, 3\}/\sqrt{6d} \), depending on how many of the signs in \( a \) match those in \( y \). Importantly, by the way \( y \) is build from an assignment satisfying the clause, at least once the signs differ. Hence \( \langle y, a \rangle \leq \alpha/\sqrt{2d} + 1/\sqrt{6d} = \alpha/\sqrt{d} = \tau \). Finally for each even \( i \) and \( b = b_i \) in \( A \), we have \( \langle y, b \rangle = \alpha/\sqrt{2d} \pm 1/\sqrt{6d} \leq \alpha/\sqrt{d} = \tau \)

TODO: Make \( b \) a little bit smaller, so it is strictly smaller than \( \tau \), or the intersection with \( a \) larger.

In the other direction, we’ll show that given a \( y \) from the verification problem, we can find a satisfying assignment for the 3-Sat problem.

First notice that \( y_0 = \langle y, x \rangle = 1/\sqrt{2} \). Then from the \( b \) rows, we have \( y_i/\sqrt{3} + \alpha/\sqrt{2d} \leq \tau \) and \( -y_i/\sqrt{3} + \alpha/\sqrt{2d} \leq \tau \) for \( i \geq 1 \). This implies for all \( i \geq 1 \) that \( -1/\sqrt{2d} \leq y_i \leq 1/\sqrt{2d} \). Since \( \|y\|^2 = 1 \), the extreme values have to be achieved, hence \( y_i \in \{-1/\sqrt{2d}, 1/\sqrt{2d}\} \).

Now for each clause, there is an \( a \in A \) with corresponding signs. Since we have \( \langle y, a \rangle = \alpha/\sqrt{2d} + \{-3, -1, 1, 3\}/\sqrt{6d} \leq \tau \) depending on the number of satisfying clauses, we must have the signs not matching at least once, meaning \( y \) satisfies the clause. \( \square \)

**References**