1 PUT IN INAPPROX

1.1 Lower Bounds on Approximate Nearest Neighbor

A useful problem in data structures is to store a set of points $A$ (in some space) so that, given a point $x$ (not in $A$), you can determine the point in $A$ that is closest to $x$. You may be allowed to prepossess the points.

Problem 1.1 Online Nearest Neighbor ($\text{OnNN}_p$ and $\gamma$-$\text{OnNN}_p$):

INSTANCE : (To Preprocess) A set of points $A$ in $\mathbb{R}^d$. We will assume there are $n$ points.

INSTANCE : A query point $x$.

QUESTION : ($\text{OnNN}_p$) Which point $y \in A$ is closest to $x$ in the $p$-norm?

QUESTION : ($\gamma$-$\text{OnNN}_p$ where $\gamma > 1$) We will call the distance to the closest point $\text{OPT}$. Obtain a $y \in A$ such that $||x - y||_p \leq \gamma \text{OPT}$. (We will also allow distances other than $p$-norms such as edit distance and Hamming distance.)

1. If you do no preprocessing and, given $x$, compute its distance to every point in $A$, this takes $O(n)$ time (assuming that distances takes $O(1)$ time). This is considered a lot of time for data structures since (1) $n$ is large and (2) computing a distance is costly even if it $O(1)$.

2. Assume you knew ahead of time the set of query points. You could, in the preprocessing stage, determine for each query point which point of $A$ it is closest to. This would yield query time $O(1)$ but an absurd (1) time for preprocessing, and (2) and space for the data structure.

Is there a way to get both quick preprocessing and quick query times? What if you settle for an approximation? Assuming SETH the answer is no:

Theorem 1.2

1. (Rubinstein [3]) Let $p \in \{1,2\}$ Assume SETH. Let $\delta, c > 0$. There exists $\epsilon = \epsilon(\delta, c)$ such that no algorithm for $\text{OnNN}_p$ has (1) preprocessing time $O(n^c)$, (2) query ties $O(n^{1-\delta})$ and solves $(1 + \epsilon)$-$\text{OnNN}$. (The result also holds for edit-distance and Hamming-distance.)

2. (Ko & Song [2]) Assume SETH. Let $\delta, c > 0$. There exists $\epsilon \in [0,1)$ such that no algorithm for $\text{OnNN}_p$ has (1) preprocessing time polynomial in $n$, (2) query time $O(n^{1-\delta})$ and solves $(1 + \epsilon)$-$\text{OnNN}_p$. 
We have stated that DOM is $W[2]$-complete and hence unlikely to be in FPT. However, using ETH and SETH, one can obtain sharper bounds on the parameterized complexity of DOM.

Let $k \in \mathbb{N}$. Let $\text{DOM}_k$ be the problem of, given a graph $G$, is there a Dominating set of size $k$. Clearly this problem is in time $O(n^{k+1})$. Eisenbrand and Grandoni [1] have obtaines slightly better algorithms. We state two known lower bounds. They are probably folklore since our only source is a workshop on fine-grained complexity held by the Max Plank Institute in 2019:


**Theorem 2.1**

1. Assume ETH. There exists $\delta > 0$ such that, for large $k$, $\text{DOM}_k$ requires time $\Omega(n^{\delta k})$.

2. Assume SETH. Let $k \geq 3$ and $\epsilon > 0$. $\text{DOM}_k$ requires time $\Omega(n^{k-\epsilon})$.

Those same notes leave the following as an exercise:

**Exercise 2.2** Assume ETH. Show that SUBSETSUM cannot be solved in time $2^{o(n)}$.

**References**


https://doi.org/10.1016/j.tcs.2004.05.009.

