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 (*$\mathrm{OnNN_p}$ *and* γ-$\mathrm{OnNN_p}$*):*

INSTANCE : (To Preprocess) A set of points A *in* R^d. *We will assume there are* n *points.*

INSTANCE : A query point x.

QUESTION : ($\mathrm{OnNN_p}$*) Which point* $y \in A$ *is closest to* x *in the* p*-norm?*

QUESTION : (γ-$\mathrm{OnNN_p}$ *where* $\gamma > 1$*) We will call the distance to the closest point* OPT. *Obtain a* $y \in A$ *such that* $\|x - y\|_p \le \gamma\mathsf{OPT}$. *(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* $\mathrm{OnNN_p}$ *has (1) preprocessing time* $O(n^c)$, *(2) query ties* $O(n^{1-\delta})$ *and solves* $(1 + \epsilon)$-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* $\mathrm{OnNN_p}$ *has (1) preprocessing time polynomial in* n, *(2) query time* $O(n^{1-\delta)}$ *and solves* $(1 + \epsilon)$-$\mathrm{OnNN_p}$.

2 PUT INTO THE FPT STUFF

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 $k \in \mathsf{N}$. Let DOMk 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:
`https://www.cs.umd.edu/~gasarch/BLOGPAPERS/maxplankfinegrained.pdf`

Theorem 2.1

1. *Asume* ETH. *There exists* $\delta > 0$ *such that, for large* k*,* DOMk *requires time* $\Omega(n^{\delta k})$*.*
2. *Assume* SETH. *Let* $k \geq 3$ *and* $\epsilon > 0$*.* DOMk *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

[1] F. Eisenbrand and F. Grandoni. On the complexity of fixed parameter clique and dominating set. *Theoretical Computer Science*, 326(1-3):57–67, 2004.
`https://doi.org/10.1016/j.tcs.2004.05.009`.

[2] Y. K. Ko and M. J. Song. Hardness of approximate nearest neighbor search under l-infinity, 2020.
`https://arxiv.org/abs/2011.06135`.

[3] A. Rubinstein. Hardness of approximate nearest neighbor search. In I. Diakonikolas, D. Kempe, and M. Henzinger, editors, *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC*

2018, Los Angeles, CA, USA, June 25-29, 2018, pages 1260–1268. ACM, 2018.
`https://doi.org/10.1145/3188745.3188916`.

[4] Unknown. (strong) Exponetial Time Hypothesis, 2019.
`https://www.mpi-inf.mpg.de/fileadmin/inf/d1/teaching/summer19/finegrained/lec2.pdf`.