Data Mining (COL761) A3

March 2024

Instructions

- Submission deadline is **22 March 23:59 IST 2024** for code and **24 March 23:59 IST 2024** for report.
- You need to do the homework in your already formed teams. You will upload the code and report to the GitHub repository mentioned in A0.
- **Your code must compile and execute on HPC.**
- You will submit a script to **moodle** (details in submission instructions section).
- Do not copy the code from your friends or from the internet. **Plagiarism will result in -X marks where X is the total of the assignment.**
- You may choose any of the following languages to write your code in: C, C++, Python. Using Python to plot graphs is suggested.
- Please read the full document clearly.
- **Buffer day policy:** Since the midsem break starts from 23 March 2024, we will consider two late days as one buffer day. Buffer days used will be $\max\{b_{\text{code}}, b_{\text{report}}\}$, where b_{code} and b_{report} refer to buffer days used for code submission and report submission, respectively.

Q.1 This problem is about the behaviour of a uniform distribution of points in high-dimensional spaces. Generate a dataset of 1 million random points in d-dimensional space (d varying as $1, 2, 4, 8, 16, 32$, and 64). Assume that the points are uniformly distributed over $[0, 1000]$ in each dimension and that the dimensions are independent. Choose 100 query points at random from the dataset. Examine the farthest and the nearest data point from each query. Compute the distances using L_1, L_2, and L_∞. Plot the average ratio of farthest and the nearest distances versus d for the three distance measures. Make sure to not include the query point itself in the nearest data point computation. Explain the results. [20 points]

Q.2 Dataset [Marks: 10 points for correct algorithm. 50 points for correct implementation. 10 points for correct explanation.]

Preliminary Step: Reduce to dimensions $2, 4, 10$, and 20 using PCA (use the implementation of PCA from `sklearn.decomposition` with `random_state=42` option).

Part a) Index using KD-tree, M-tree, and LSH for L_2 distance. You can use off-the-shelf libraries if you wish to. Maintain the dataset in memory. For LSH, a popular library is LSH Primer · FALCONN-LIB/FALCONN Wiki · GitHub.

Part b) Write an algorithm to perform k-NN query using each of these index structures as well as without any indexing across all dimensions ranging from 2 to 20. You must explain the algorithm in your report.

Part c) Choose 100 random points as query and plot the average running time of 5-NN query with standard deviation as error bars for each index structure and sequential scan against dimension. For LSH also plot accuracy (Jaccard) against dimension. Explain the trends.

Part d) Plot the k-NN accuracy (Jaccard) of LSH against k (x-axis) for the following values of k: $1, 5, 10, 50, 100$, and 500.

Submission instructions.

These instructions will be constant across assignments, and have already been shared in A1. So, make sure that you have structured your A3 assignment accordingly. Like other assignments, you will be submitting

- `clone.sh` to moodle. Should contain just one line similar to `git clone https://github.com/my_repo` if your repository is named "my_repo". **Do not include personal access tokens, and don't put ssh clone commands.**
- This should clone your repository. Say your repository is named "my_repo", then we should find `my_repo/A3`.
- Within "A3" directory there should be
  - `compile.sh`,
  - and `interface.sh` (more details of the interface format will be announced later).
- You will submit a report containing algorithm description, plots, observations, and explanation or logical arguments supporting your observations.