COL 761

Final Exam

Total: 50 points + 30 points for extra credit (to be added to HW component)

---

1. **(4 points)** What assumption(s) does MCL make about clusters in a graph in addition to the property that nodes in a cluster have large number of paths between them and low connectivity among nodes across clusters?

2. **(5 points)** Which of the following clustering algorithms can be used to cluster graphs in a graph database with a metric distance function for graphs. [1 point for correct options, -1 for incorrect options. 5 points (i.e., 1 point bonus) if you get all correct options and 0 incorrect options]
    a. K-means
    b. DBSCAN
    c. K-medoid
    d. Single-linkage

3. **(5 points)** It is clear that parameters are easier to set in OPTICS than in DBSCAN. But if parameter selection is not a problem (let's say some oracle tells us the best parameters), would you still say OPTICS is better? Explain.

4. **(8 points)** What is the time complexity of the fastest possible algorithm for single-linkage hierarchical clustering? Write the algorithm and the complexity analysis.

5. **(6 points)** Suppose you have three distance functions, d_1, d_2, d_3 to rank webpages for a given query keyword. To identify which is the best distance function, you conducted a survey across 1000 people, where each person searched for a web query, and were shown the top-ranked page by each of the three distance functions. The users were asked to choose the result that they liked the most. You found out that 500 people voted for d_1. Similarly, d_2 received 300 votes and d_3 received 200 votes. How can you infer if this distribution of votes is purely due to chance or there is a definite preference towards d_1? Explain precisely and formally.

6. **(5 points)** Let G be an edge-weighted (only positive edge-weights) undirected graph. Let the distance *d(u,v)* between two nodes in the graph be the length of the shortest path from *u* to *v*. The length of a path is the sum of its constituent edge weights. Prove $d(u, v)$ satisfies triangular inequality.

7. **(5+2=7 points)** Derive the querying time complexity of range query in a *d*-dimensional KD-tree. Write down the recursion you will have for the maximum number of intersections with the query region in terms of *both n and d*, and the final complexity. You must provide the detailed derivation in addition to writing down the expressions below. No points will be awarded for just expressions.

Answer: Q(n)=_____, O(_____)

8. **(10=6+2+2 points)** Suppose you have a database of 10×10^6 text documents, where each document is a d-dimensional bit vector. The similarity between two documents is the Jaccard similarity between them. The Jaccard distance can analogously be defined as (1 - Jaccard Similarity). Given a query document, you want to use LSH to identify its 1-NN. You are **not** allowed to convert the dataset into Hamming Space or perform any other space transformations. It is given to you that the 1-NN always resides within a Jaccard Similarity of 0.8. In other words, the 1-NN has a similarity of 0.8 or more with any query. You are allowed to absorb an approximation error of $\epsilon = 1$ in the LSH. Answer the following questions with respect to this problem.
    a. Propose a locality sensitive hash function with parameters (r1, r2, p1, p2) as defined in the slides. Specifically, i) mention your hash code generation policy, and ii) the values of r1, r2, p1, p2. These guarantees must hold on the original Jaccard Similarity (or distance) itself

and not on Hamming distance or some other converted space. Note that r1 and r2 are distance radii. So, convert Jaccard similarity to distance accordingly.

b. What should be the value of H, i.e., the number of hash codes per table?
c. What should be the value of L, i.e., the number of hash tables?

[Note: You can leave the answers to part a, b and c above at an expression level. You don't need to solve them]

Extra Credit Questions. The marks you obtain in this section will be added to your Homework component. **[20 points]**

9. **(10 points)** True/False questions [2 points for correct answer, -2 for incorrect answer]
   a. The event of finding 20 heads and 30 tails out of 50 coin tosses has a p-value below 0.05.
   b. With increase in inflation parameter, MCL would identify a smaller number of clusters.
   c. MBRs in R-tree may overlap in space but not in actual data points.
   d. The Space-saving algorithm is likely to work better for uniform frequency distribution than power-law distribution.
   e. Complete linkage clustering tries to minimize the diameter (farthest distance between any pair of points) of clusters.

10. **(10 points)** Is the distance function dynamic time warping (DTW) metric? Prove or disprove. DTW between two time series sequences T1 and T2 is defined as the following:

$$\text{DTW}(T_1, T_2) = \begin{cases} 0 & \text{if } |T_1| = |T_2| = 0 \\ \infty & \text{if } |T_1| = 0 \text{ or } |T_2| = 0 \\ dist(T_1.s_1, T_2.s_1) + & \text{otherwise} \\ min\{DTW(Rest(T_1), Rest(T_2)), & \\ DTW(Rest(T_1), T_2), DTW(T_1, Rest(T_2))\} & \end{cases}$$

A time series sequence $T=[s_1,\ldots,s_n]$ is a sequence of points. You are free to choose any distance function as dist() as long as it satisfies metric properties. Rest(T) is the sub-sequence containing all points of T except T.s1.

11. **(10 points)** In Bloom filters, we have an array of *n* bits, where n is the maximum number of bits that can be maintained in memory and *k* hash functions that hash to these *n* bits.

a. The number of hash functions, k, allows us to improve the false positive rate. Are there any disadvantages of setting a very high value of k? Explain. [4 points]

b. Consider an alternative hashing scheme where we have k different bit vectors, all of equal sizes. We choose the size of the bit vectors such that all k of them can be maintained in memory. We also have k hash functions, but the i^{th} hash function can hash only into the i^{th} bit vector. We have m "good" objects that we hash in pre-processing. A new object is classified as positive only if it hashes into 1-bits (i.e., a bit with value 1) for all *k* hash functions. Would the false positive rate be worse or better in this modified scheme if the memory budget (total number of bits) for both schemes are same? Prove or disprove. [6 points]