# Big Data Analytics

**Homework 3 (X-Tree, M-Tree, and OMNI-Family)**

**Instructor: Ruoming Jin**

**Name: Shiva Kumar Peddapuram**

**KSU ID: 811235874** [**speddapu@kent.edu**](mailto:speddapu@kent.edu)

**In this homework, there are 4 questions + 1 bonus question, covering the X-tree, M-tree, and OMNI-family. If you can answer the bonus question correctly, you can obtain 20 extra points. The maximum mark for this homework is 120 points, which will be later scaled.**

1. **Please list two major differences (or improvement) of X-tree from R\*-tree. [10 points]**

**Answer:**

|  |  |
| --- | --- |
| **X-Tree** | **R\*-Tree** |
| 1. Insertions process for the X-Tree can be done very faster compared to R\* Tree | In R\*-tree the insertion process is very slow compared to X- Tree |
| 1. X-Tree is very fast in searching the data points | R\*-tree is very slow in searching the required data points |
| 1. As the dimensional data increases then the X-Tree provides high and efficient perfomance | R\*-tree does not give high performance for the higher dimensional data |
| 1. This type of the tree is widely used in spatial and geographical databases. | It is widely used in the multimedia databases. |

1. **Please list at least 2 distance functions that are *metric distances*, and at least 2 distance functions (or similarity measures) that are *non-metric distances*. For each distance function, give the reason why it is (or is not) a metric distance, and give references or URL links. (*Hint: please search on the Web or Wikipedia to find the answers*) [20 points].**

**Answer:**

**Metric Distances:**

A Metric Distance is defined as a function which is the distance in between each pair of the elements on a given set.

**Conditions are:**

|  |
| --- |
| **The Satisfactory condition to be a metric distance will be as follows**   * 1. **d(x,y)=0 if (x==y)**   2. **d(x , y) = d(y, x)**   3. **d(x , z) <= d(x , y) + d(y , z) triangle inequality** |

Example of metric distances:

Here I have taken some reference from wikipedia

***Taxicab distance:***

Taxicab distance is defined as the distance in between the two points in a defined vector space is the sum of the asolute figgerence of their cartesiaon coordinates in the n dimensions.

**It is represented as in formular way:**

**Euclidean distance:**

Euclidean distance is defined as the distance in between a normal straight line in between 2 points of a given Euclidean space.

**The formular representation is as below:**

Text

Description automatically generated with low confidence

***Graph Edit Distance:***

Graph edit distance is defined as the measurement of the similarity or dissimilarity in between given two graphs.

**The formular representation of the graph edit distance between two graphs GED( g1, g2 ) is defined as below:**

Text

Description automatically generated with medium confidence

Whereas the point P( g1, g2 ) = set of edit paths by transforming graph g1 isomorphically to g2

**Non-Metric Distances:**

Non-Metrics-Distance is defined as the distance that doesn’t obey the rules of triangle inequality is called as the Non-Metric distances.

***Partial Hausdorff distance***:

Partial hausdroff distance is generally defined/proposed for the shape based image retrieval.

We can represent the hausdroff distance for Given two sets S1, S2 of points based hausdroff distance is

Where dNP is defined as the Euclidean distance of the ith point in S1 to nearest point in S2.d

***Dynamic Time Wrapping distance:***

Dynamic wrapping distance is one of the algorithms which measure the similarit between two sequences. It uses multimedia data and applies Dynamic time wrapping algorithm to make the data linear and it will be analyzed. As mentioned it is not guaranteed to hold to triangle inequality so we took it as non metric distance.

**3. Please read the lecture slide of Chapter 5, "Range Queries Over M-Tree", and prove the pruning strategy for the range query below (*Hint: use the triangle inequality*) [30 points]:**

**If |*d*(*Op*, *Q*)-*d*(*Or*, *Op*)|>*r*(*Q*)+*r*(*Or*), then *d*(*Or*, *Q*) > *r*(*Q*) + *r*(*Or*) holds and node centered at *Or* with radius *r*(*Or*)can be safely pruned.**

A picture containing text, chain, metalware, watch

Description automatically generated

Answer:

For the following question I have derived the equation in pen and paper based method and attached below as it was convenient way for me.

Text, letter

Description automatically generated

**4. (The Curse of Dimensionality) [40 points]**

**4a. What is the curse of the dimensionality? Please provide the reason for the dimensionality curse. [10 points]**

**Answer:**

**Curse of the dimensionality:**

* The curse of dimensionality is defined as the different methodologies and techniques in which helps us in dealing with wide range of data in high dimensional environments.
* As the data gets high (huge) it becomes very much difficulty and challenging to analyze and process that huge data
* It also becomes very much difficult to represent the statistical significance from that huge data.
* As these high dimensional data is in unorganized form we have lot of issues and unnecessary noises so in order to overcome these issue we use many artificial and machine learning algorithms to sort out these issue and analyze the data.
* So in these scenarios the above technique is used to solve the high dimensional data problems.

**4b. Read Section 2 of the following paper, and write a short survey about existing dimensionality reduction techniques and high dimensional data structures mentioned in this section (*Please cite reference papers in your survey and provide a list of reference papers after the survey. You may need to read abstract or introduction of some reference papers, if they are unclear in the section. Note: please use your own words to describe the techniques; DO NOT copy any sentences from the paper*). [30 points]**

**H. T. Shen, X. Zhou, and A. Zhou. An adaptive and dynamic dimensionality reduction method for high-dimensional indexing. In *VLDBJ*, 2006. *Located in the Library Course Reserves on the left-hand course menu.***

**Answer:**

As there is so much data available in the real world, we often store this huge data in the form of high-dimensional storage to keep them safe and secure.

To efficiently obtain the data from a high dimensional space that has such a vast amount of data, indexing is used. For greater effectiveness, all multidimensional measures have a tendency to scale up to high dimensions. Moving to high-dimensional causes indexing to perform worse, which has an impact on how effective indexes are. Before creating or creating any form of index, we will reduce the dimensionality of the original dataset to solve this issue

Some techniques are:

***Global Dimension Reduction:***

Principal Component Analysis will be used to reduce the dimensionality of the data in this technique, and multidimensional indexing structures will be used to index the reduced spaces. Only when the data provided is globally related to one another will any of this be effective. Global dimension reduction refers to this. The main issue with this method is that the dataset is not properly globally correlated. As a result, some data is lost, and the cost of running queries rises as well.

**Local Dimension Reduction:**

The data can be connected locally as well as globally. With this method, we will first attempt to find local correlations in the data and conduct dimensionality reduction on those correlations. Range searches, k-nearest neighbor queries, and many other types of queries can be supported by indexing on such locally connected data. This results in reduced distance information loss and cheaper query costs when compared to GDR.

**Data transformation and approximation**:

Using a VA-File, or vector approximation file, we break down the original data into smaller data points in the form of bit sequences. This VA file's disadvantage is that it does not readily adjust to heavily skewed This VA file's disadvantage is that it does not readily adjust to heavily skewed data.

**Pyramid Technique:**

The pyramid mapping technique reduced high-dimensional space to a single dimension. It is based on a partitioning technique where the data space is split into two-dimensional pyramids. The single pyramid is then divided into slices that run parallel to its base. The data pages are actually made up of these slices. We employ a B+ tree to convert the data from a d-dimensional space to a single dimension. In comparison to previous techniques, this methodology is very adaptive to range queries and its speed is unaffected while processing range queries on data with higher dimensions.

**Proclus:**

Using this method, data are grouped together based on how well they correlate with the initial dimension. The algorithm contains three phases. The high-dimensional dataset is split into potential meloids in the initialization phase, and each dimensional data is iterated through in the iteration phase. Refinement is the third phase, where we replace a problematic meloid with a bad meloid and assess whether doing so improves performance.

**Optimal Grid Algorithm:**

The Optimal Grid is a clustering method that is based on creating an ideal grid split of the data. Using certain data projections, we will use this technique to determine the appropriate portioning hyperplane needed for each dimension. It is the most effective algorithm for massive, high-dimensional data sets. From the available clustering approaches, this one is the best for high dimensional data. Strong mathematical computations serve as its foundation.

**Bonus Question [20 extra points]**

5. Read Section 2.2 of the following paper, as well as the cited papers in this subsection, and write a short survey about the *intrinsic dimensionality*. (***Note:*** *please use your own words to describe the problem definition and solutions;* ***DO NOT*** *copy any sentences from the paper*).

R.F.S. Filho, A. Traina, C. Traina, and C. Faloutsos. Similarity search without tears: the OMNI-family of all-purpose access methods. In *ICDE*, 2001. *Located in the Library Course Reserves on the left-hand course menu.*

***Answer:***

***Intrinsic Dimensionality:***

* The dataset's attributes are determined by the ***embedded dimensionality***.
* The embedded data set, which is a portion of the high dimensional dataset from the entire dataset, is processed using intrinsic dimensionality.
* Intrinsic dimensionality uses closest neighbor estimators for nonlinear data sets and principal component analysis for linear spaces, making it very simple to do on such a tiny quantity of data compared to high dimensional data.

***Hausdorff Dimension:***

* Embedded dataspace is partitioned into cubic grid cells in the Hausdorff Dimension, and evaluation is carried out using mathematical procedures.
* For processing embedded data sets, we will perform better if we employ cubic grid cells.
* To provide a better performance while using with the one standard deviation, dimensions are used.

***Co-related Fractal Dimension:***

* Co-related fractal dimension is used to measure dimensionality for the random points that are present in the high dimensional data sets.
* By using this method it helps and boosts in the processing of the data sets and the best part is it is less nosy as compared to that of hausdroff dimensions.
* By keeping fewer properties in our dataset, this dimension helps and plays a vital role in the removal of unwanted variables.
* In order to get a desired and correct results in this dimensions we pass a constant number of passes to the data.