Spatial Data Mining: Homework 1 (Due 09/23 at 11:59PM)

Use blue color to write your answers and submit on ELMS.

Writing Problems

1. Concepts:

1.1 (20 points) True or False. Review the following statements related to the concepts covered in lectures and mark each of them as "True" (T) or "False" (F). Provide a brief explanation if you mark a statement as false; no need for those marked as true (you can add an explanation if you are unsure about your answer).

(1) Calculating integrals and derivatives for non-trivial mathematical functions are considered as data mining tasks. False

(2) Spatial data mining is a sub-area of data mining, where methods developed for traditional data mining can be directly applied to get good solutions. True

(3) Fields such as data mining, data science, machine learning and AI have overlapping topics and techniques. True

(4) Locations, represented as points (i.e., latitude and longitude), are sufficient for navigation services such as Google Maps. True

(5) Large volumes of spatial data (e.g., satellite imagery) can be freely accessed online. True

(6) Traditional databases are limited for storing and querying spatial data. True

(7) Spatial data are both auto-correlated and heterogeneous. True

(8) Spatial datasets should be transformed into the same coordinate system and projection before they are used together for analysis. True

(9) Point datasets have fixed length to record spatial information (e.g., latitude and longitude), so they can be managed and queried efficiently using traditional non-spatial databases. True

(10) The purpose of database normalization (e.g., 3rd normal form) is to improve the efficiency of data processing. True

1.2 (20 points) Multiple choice questions (can be single choice).

(1) Which of the following is (are) considered high-dimensional data?

A. Data with a large number of attributes (i.e., a data table with many columns) True

B. Data with a large number of samples (i.e., a data table with many rows)

C. Data with attributes having a wide range of values

(2) Which of the following describes the Modifiable Areal Unit Problem (MAUP) or gerrymanding?

A. Statistical results or conclusions remain the same for different partitionings of underlying space

B. Statistical results or conclusions change by changing the partitioning of underlying space True

C. The area of polygons may vary by using different coordinate systems and projections

(3) Which of the following assumptions used in traditional statistical models and machine learning methods are violated by spatial data?

A. Data samples are independent to each other True

B. Data samples follow the same distribution (e.g., generated by normal distribution w. same mean and variance)

(4) Which of the following spatial data models can be directly used to compare heights of buildings (height is from ground to the top of the building)?

A. Digital Surface Model (DSM)
B. Digital Elevation Model (DEM)
C. None of the above True, typically one would use the CHM

(5) Which of the following can be possibly changed by changing the map projection?

A. Area relationship between objects (e.g., which country appears larger?) True
B. Topological relationship between two objects
C. Distance between objects

1.3 (10 points) Short narrative.

(1) Briefly describe why traditional database search index – using binary tree as an example -- is inefficient on spatial data. (Hint: what's the assumption that makes binary search faster than a linear scan? Or what has to be done so that binary tree can be used?) The assumption of binary search is that you have to basically eliminate or prune a chunk of the array (usually the middle number) if the comparing condition does not match. Essentially you are reducing the search space by half. This is great for traditional data, but for spatial data it is not efficient because you have to SORT the data first before you do the search, and it is difficult to sort spatial data because of its dimensionality, unlike traditional data, spatial data can be 2-dimensional or more.

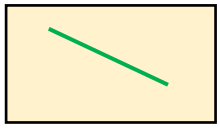
What is an example of efficient search index for spatial data? What is the key idea? Briefly describe.

R-Tree is an example of an efficient spatial index. And the main idea is creating bigger rectangles over the other polygons in the database so when you try to query it, it does a check of intersection, and if it doesn't intersect, it ignores it and narrows its scope to just the rectangle of interest and does the search there. That way it has saved a lot of time going through the whole database. And it has provided an efficient means for sorting the spatial data because we use rectangles to partition space at different levels

2. (30 points) Spatial data models and relationships.

This question will practice the 9-intersection model for spatial topological relationships.

(1) (10 points) Sequential thinking: We will start with a simple example where you will fill out 9-intersection models (9IM, and Dimension-Extended 9IM) for the following polygon and **line**. Recall that in DE-9IM you only need to convert each T (True) to a dimension (i.e., 0 for pointal, 1 for linear, and 2 for polygonal) and keep F (False) unchanged.



9IM

	Int (P)	Bd (P)	Ext (P)
Int (L)	T	F	F
Bd (L)	T	F	F
Ext (L)	T	F	F

DE-9IM

	Int (P)	Bd (P)	Ext (P)
Int (L)	1	F	F
Bd (L)	0	F	F
Ext (L)	2	F	F

L- line

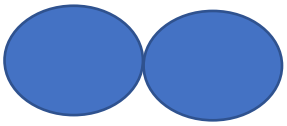
P- polygon

The exterior of the line does not touch the boundary of the polygon so I cannot say for a fact that there is a clear-cut intersection in their relationship. Unless I assume that polygon is the universe, but I am not certain about that fact, so I am leaving it as "F"

(2) (10 points)

General thinking: Now try to generalize from the specific scenario given above. A "meet" spatial relationship between two **polygons** means they only have intersection at their boundaries. What is the 9IM (no need for DE-9IM) that describes this relationship for two **polygons**? Can one 9IM filled with T and F cover all possible scenarios? True

In the following 9IM, fill in cells with a "T" or "F" value **ONLY IF** that cell must always have that value for this "meet" relationship; use "*" for uncertain cells, which might have different values in different scenarios.



9IM

	Int	Bd	Ext
Int	F	F	T
Bd	F	T	T
Ext	T	T	T

(3) (10 points) Practical use: DE-9IMs are commonly used in spatial database systems to control data quality or validate designs. Design a DE-9IM (replacing T with dimensions) for the following scenario:

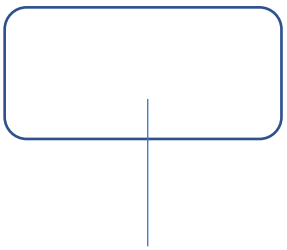
Docks are common in the US to load/unload passengers or cargos to boats (e.g., the following figure). Consider dock as **1D lines** and water bodies (e.g., lakes, rivers) as **2D polygons**. Design and specify the rules that someone can use to determine if the spatial relationship between a dock and water body is valid (i.e., possible), and then convert them to a DE-9IM by filling out the table.

Figure 1. An example dock.

DE-9IM

	Int	Bd	Ext
Int	1	0	1
Bd	0	0	0
Ext	2	1	2

**Use dimensions for T.*



If we model docks as 2D polygons, can you use DE-9IM to differentiate between bridges and docks? (Yes or No, and briefly explain why) Yes, because from the True or False/0,1,2 values inside the matrix, -one can tell the kind of spatial relationship that exists between the objects and thereafter differentiate one from another. A bridge goes from one end to the other end of the river, so it intersects the boundary of the river at both ends. But a dock doesn't extend that long, as a matter of fact, it only intersects the boundary of the river at one end. Going by that assumption, the DE-9IM model of a dock leading into a river is going to be different from that of bridge going across a river.

Programming (20 points)

1. Run the notebook provided in Google CoLab. Complete the two queries asked in the notebook and **put your results here,** including your SQL code, the number of returned rows, and the results from top 10 rows as they are printed by default. (Hint: remember the best practice in SQL formulation described in class, and the difference between cross-product and equal-join)

```
QUERY 1
cursor = conn.execute('''
SELECT e.FirstName, e.LastName
FROM invoices i, employees e, customers c
WHERE i.CustomerId = c.CustomerId and c.SupportRepId = e.EmployeeId and i.total > 25 ''')
result = cursor.fetchall()
print_sql(result, cursor)
```

```
Number of records:  1

Top 10 rows:
+----+-------------+------------+
|    | FirstName   | LastName   |
|----+-------------+------------|
|  0 | Steve       | Johnson    |
+----+-------------+------------+
```

```
QUERY 2
cursor = conn.execute('''
    SELECT p.Name
    FROM playlists p, playlist_track pt, tracks t, genres g
    WHERE p.PlaylistId = pt.PlaylistId and pt.TrackId = t.TrackId and t.GenreId = g.GenreId and
g.Name = 'Jazz'
''')
result = cursor.fetchall()
print_sql(result, cursor)
```

```
Number of records:  286
```

Top 10 rows:

	Name
0	Music
1	Music
2	Music
3	Music
4	Music
5	Music
6	Music
7	Music
8	Music
9	Music

```
cursor = conn.execute('''

SELECT e.FirstName, e.LastName
FROM invoices i, employees e, customers c
WHERE i.CustomerId = c.CustomerId and c.SupportRepId = e.EmployeeId and i.total > 25


 ''')

result = cursor.fetchall()
print_sql(result, cursor)
```

```
Number of records:  1
Top 10 rows:
+----+-------------+------------+
|    | FirstName   | LastName   |
|----+-------------+------------|
|  0 | Steve       | Johnson    |
+----+-------------+------------+
```

QUERY 1: Showing the code, the number of records and the First 10 rows of the results

```
cursor = conn.execute('''
      SELECT p.Name
      FROM playlists p, playlist_track pt, tracks t, genres g
      WHERE p.PlaylistId = pt.PlaylistId and pt.TrackId = t.TrackId and t.GenreId = g.GenreId and g.Name = 'Jazz'
''')

result = cursor.fetchall()
print_sql(result, cursor)
```

```
Number of records:  286
Top 10 rows:
+----+--------+
|    | Name   |
|----+--------|
|  0 | Music  |
|  1 | Music  |
|  2 | Music  |
|  3 | Music  |
|  4 | Music  |
|  5 | Music  |
|  6 | Music  |
|  7 | Music  |
|  8 | Music  |
|  9 | Music  |
+----+--------+
```

QUERY 2: Showing the code, the number of records and the First 10 rows of the results