Mining Frequent Spatial Patterns in Transactional Databases

What is frequent spatial pattern mining?

Frequent pattern mining aims to discover all interesting patterns in a transactional database that have **support** no less than the user-specified **minimum support** (**minSup**) constraint and **distance** no greater than user-specified **maximum distance** (**maxDist**). The **minSup** controls the minimum number of transactions that a pattern must appear in a database. The **maxDist** controls the maximum distance between two items in pattern should be less.

What is the transactional database?

A transactional database is a collection of transactions, where each transaction contains a transaction-identifier and a set of items.
A hypothetical transactional database containing the items *a, b, c, d, e, f, and g* as shown below

tid	Transactions
1	a b c g
2	b c d e
3	a b c d
4	a c d f
5	a b c d g
6	c d e f
7	a b c d
8	a e f
9	a b c d
10	b c d e

Note: Duplicate items must not exist in a transaction.

Acceptable format of transactional databases in PAMI

Each row in a transactional database must contain only items. PAMI algorithms implicitly consider the row number as the transactional-identifier to reduce storage and processing costs.

a b c g
b c d e
a b c d
a c d f
a b c d g
c d e f
a b c d
a e f
a b c d
b c d e

What is the spatial database?

Spatial database contain the spatial (neighbourhood) information of items. It contains the items and its nearset neighbours satisfying the **maxDist** constraint.

Items	neighbours
a	b, c, d
b	a, e, g
c	a, d
d	a, c
e	b, f
f	e, g
g	b, f

Understanding the statisctics of database

To understand about the database. The below code will give the detail about the transactional database.

- Total number of transactions (Database size)
- Total number of unique items in database
- Minimum lenth of transaction that existed in database
- Average length of all transactions that exists in database
- Maximum length of transaction that existed in database
- Standard deviation of transaction length
- Variance in transaction length
- Sparsity of database

The sample code

import PAMI.extras.dbStats.transactionalDatabaseStats as stats

obj = stats.transactionalDatabaseStats('sampleInputFile.txt', ' ')
obj.run()
obj.printStats()

What is the input to frequent spatial pattern mining algorithms

Algorithms to mine the frequent spatial patterns requires transactional database, spatial database and minSup (specified by user).

- Transactional database in following formats:

  - In string format
    (`/Users/Likhitha/Downlaods/sampleInputFile.txt`)
  - In URL format (`https://www.u-` `aizu.ac.jp/~udayrage/datasets/transactionalDatabases/tra`
  - In DataFrame format (dataframe variable with heading `Transactions`)

- Spatial database in following formats:

  - In string format
    (`/Users/Likhitha/Downlaods/sampleNeighbourFile.txt`)
  - In URL format (`https://www.u-` `aizu.ac.jp/~udayrage/datasets/transactionalDatabases/tra`
  - In DataFrame format (dataframe variable with headings `item` and `Neighbours`)

- minSup should be mentioned in **count (beween 0 to length of database)** or __percentage (multiplied with length of database)
- Seperator of the input file should be mentioned

What is the output of frequent spatial pattern mining algorithms

The output of these algorithms is in two ways:

- Saving the patterns in user specified output file.
- Returns the patterns in dataframe variable.

How to run the frequent spatial pattern algorithm in terminal

- Download the code from github.
- Navigate to PAMI folder where you downloaded the file.
- Go to periodicFrequentSpatialPattern folder

Execute the following command on terminal.

python3 algorithmName.py `path of Sample input file` `path of neighbour file` `path of output file` minSup maxPer `seperator`

Sample command to execute the code in frequentSpatialPattern/basic folder

python3 `FSPGrowth.py` `/Users/Donwloads/inputFile.txt` `/Users/Downloads/neighbourFile.txt` `/Users/Downloads/outputFile.txt` 3 `' '`

How to implement the code by importing PAMI package

Import the PAMI package executing: **pip3 install PAMI**

Run the below sample code by making simple changes

- Replace sampleInputFile name or path in place of iFile, sampleNeighbourFile in place of nFile and sampleOutputFile name or path in place of oFile
- Specify the minSup (like 10 or 0.1) in place of minSup
- Specify the seperator of input file after minSup. (If no seperator is specified the default tab seperator is considered for input file)

import PAMI.frequentSpatialPattern.basic.FSPGrowth as alg

obj = alg.FSPGrowth(iFile, nFile, minSup, ',')

obj.startMine()

obj.savePatterns(oFile) (to store the patterns in file).

Df = obj.getPatternsAsDataFrame() (to store the patterns in dataframe)

obj.printStats()

What is the output of frequent spatial pattern mining algorithms

Returns the pattern and support respectively with $minSup=5$ for above sample data

The output in file format:

a: 7
b: 7
c: 9
d: 8
a b: 5
a c: 6
a d: 5
c d: 8

The output in DataFrame format:

	Patterns	Support
0	a	7
1	b	7
2	c	9
3	d	8
4	a b	5
5	a c	6
6	a d	5
7	c d	8