Mining GeoReferenced Fuzzy Periodic Patterns in Fuzzy Databases

What is Geo referenced Fuzzy Periodic Frequent pattern mining?

Fuzzy Frequent Spatial Pattern mining aims to discover all Spatially frequent fuzzy patterns in a fuzzy database that have **support** no less than the user-specified **minimum support** (**minSup**) constraint, **periodicity** no greater than user-specified **maximum periodicity** (**maxPer**) constraint and **distance** between two items is no less than **maximum distance** (**maxDist**). The **minSup** controls the minimum number of transactions that a pattern must appear in a database and the **maxPer** controls the maximum time interval within which a pattern must reappear in the database.

What is a fuzzy database?

A fuzzy database is a collection of transaction, where each transaction contains a set of items and spatial(neighbourhood) information of items. It contains the items and its nearset neighbours satisfying the **maxDist** constraint.

A hypothetical utility database with items ***a, b, c, d, e, f and g*** and its fuzzy values are shown below:

Transactions	fuzzy values
a b c g	5 4 3 2
b c d e	5 2 9 3
a b c d	2 3 5 6
a c d f	1 3 4 6
a b c d g	2 5 3 6 1
c d e f	2 3 4 5
a b c d	5 4 3 2
a e f	4 8 3
a b c d	7 4 9 8
b c d e	5 9 10 24

Note: Duplicate items must not exist in a transaction.

What is spatial Database?

A spatial database files consists item and set of its neighbours. A hypothetical spatial database is defined below

Item	Neighbours
a	b c d
b	a e g
c	a d
d	a c
e	b f
f	e g
g	b f

Accepted format of spatial database in PAMI

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

What is the acceptable format of a fuzzy databases in PAMI?

Each row in a utility database must contain only items, total sum of utilties and utility values. A sample transactional database, say sampleInputFile.txt, is provided below.

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

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
- Minimum utility value exists in database
- Average utility exists in database
- Maximum utility exists in database
- Standard deviation of transaction length
- Variance in transaction length
- Sparsity of database

The below sample code prints the statistical details of a database.

In [ ]:
```
import PAMI.extras.dbStats.utilityDatabaseStats as stats

obj = stats.utilityDatabaseStats('sampleInputFile.txt', ' ')
obj.run()
obj.printStats() <br>
```

What are the input parameters

The input parameters to a frequent pattern mining algorithm are:

- **Fuzzy database**
  Acceptable formats:

  > - String : E.g., 'fuzzyDatabase.txt'
  > - URL : E.g., https://u-aizu.ac.jp/~udayrage/datasets/transactionalDatabases/transactional_T10
  > - DataFrame with the header titled 'Transactions', 'Utility' and 'TransactionUtility'

- **Spatial database**
  Acceptable formats:

  > - String : E.g., 'spatialDatabase.txt'
  > - URL : E.g., https://u-aizu.ac.jp/~udayrage/datasets/transactionalDatabases/transactional_T10
  > - DataFrame with the header titled 'item' and 'Neighbours'

- minSup should be mentioned in **count (beween 0 to length of database)**

  > - [0, 1]

- maxPer should be mentioned in **count (beween 0 to length of database)**

  > - [0, 1]

- **seperator**
  default seperator is '\t' (tab space)

How to store the output of a geo referenced fuzzy periodic frequent pattern mining algorithm?

The patterns discovered by a fuzzy periodic frequent pattern mining algorithm can be saved into a file or a data frame.

How to run the geo referenced periodic-frequent pattern algorithm in terminal

- Download the PAMI source code from github.
- Unzip the PAMI source code folder
- Enter into fuzzySpatialPeriodicFrequentPattern folder
- Enter into the folder and execute the following command on terminal.

syntax: python3 algorithmName.py `<path to the input file>` `<path to the output file>` `<minSup>` `<maxPer>` `<seperator>`

Example: python3 `FGPFPMiner.py` `inputFile.txt` `outputFile.txt` `neighbourFile.txt` `$5$` `$3$` `' '`

How to execute a fuzzy spatial periodic frequent pattern mining algorithm in a Jupyter Notebook?

- Install the PAMI package from the PYPI repository by executing the following command: **pip3 install PAMI**
- Run the below sample code by making necessary changes

In [ ]:
```
import PAMI.fuzzySpatialPeriodicFrequentPattern.FGPFPMiner as alg

iFile = 'sample_Input.txt'  #specify the input utility database <br>
minSup = 5  #specify the minSupvalue <br>
maxPer = 3
seperator = ' '
oFile = 'fuzzySpatialPeriodicPatterns.txt'   #specify the output file nam
nFile = 'sampleNeighbourFile.txt'      #specify the neighbour file of dat
fuzFile = 'fuzFile.txt'

obj = alg.FGPFPMiner(iFile, nFile, fuzFile, minSup,maxPer, seperator) #in
obj.startMine()                        #start the mining process <br>
obj.savePatterns(oFile)                #store the patterns in file <br>
df = obj.getPatternsAsDataFrame()      #Get the patterns discovered into a
obj.printStats()                       #Print the statistics of mining pro
```

The fuzzySpatialPeriodicPatterns.txt file contains the following patterns (*format:* pattern:support):!cat fuzzySpatialPeriodicPatterns.txt

In [8]:
```
!cat fuzzySpatialPeriodicPattern.txt
```

```
a.L : 5.4
b.L : 5.6
d.L : 6.199999999999999
d.L c.L : 5.4
c.L : 7.0
```

The dataframe containing the patterns is shown below:

In [5]:
```
df
```

Out[5]:

	Patterns	Support
0	a.L	5.4
1	b.L	5.6
2	d.L	6.199999999999999
3	d.L c.L	5.4
4	c.L	7.0