Mining fuzzy correlated patterns in transactional database

What is fuzzy correlated pattern mining?

Fuzzy correlated pattern mining aims to discover all the interesting patterns using **support, all confidence** that have **support** no less than the user-specified **minimum support (minSup)**,**all confidence** no less than **minimum all confidence (minAllConf)**.It is based on traditional fuzzy frequent pattern mining.

Reference: Lin, N.P., & Chueh, H. (2007). Fuzzy correlation rules mining.

What is a fuzzy transactional database?

A fuzzy transactional database is a collection of transaction, where each transaction contains a set of items called **fuzzy terms** and a positive integer called **fuzzy value** respectively. A hypothetical fuzzy database with items ***a, b, c, d, e, f and g*** is shown below.

Transactions
(a,2) (b,3) (c,1) (g,1)
(b,3) (c,2) (d,3) (e,2)
(a,2) (b,1) (c,3) (d,4)
(a,3) (c,2) (d,1) (f,2)
(a,3) (b,1) (c,2) (d,1) (g,2)
(c,2) (d,2) (e,3) (f,1)
(a,2) (b,1) (c,1) (d,2)
(a,1) (e,2) (f,2)
(a,2) (b,2) (c,4) (d,2)
(b,3) (c,2) (d,2) (e,2)

Note: Duplicate items must not exist within a transaction.

What is the acceptable format of a fuzzy transactional database in PAMI?

The format we accept for fuzzy databases is the same as the utility database format.

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 statistics of a temporal database

The performance of a pattern mining algorithm primarily depends on the satistical nature of a database. Thus it is important to know the following details of a database:

- Total number of transactions (Database size)
- Total number of unique items in database
- Minimum lenth of transaction that exists in database
- Average length of all transactions that exists in database
- Maximum length of transaction that exists in database
- Minimum periodicity that exists in database
- Average periodicity hat exists in database
- Maximum periodicity that 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()
```

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

- **minSup**
  specified in

  > - **count (beween 0 to length of a database)** or
  > - [0, 1]

- **minAllConf**
  specified in

  > - [0, 1]

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

How to store the output of a fuzzy correlated pattern mining algorithm?

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

How to run the correlated periodic pattern mining algorithms in a terminal?

- Download the PAMI source code from github.
- Unzip the PAMI source code folder and enter in to fuzzy correlated pattern
- Enter into fuzzyCorrelated pattern 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>` `<minAllConf>` `<seperator>`

Example: python3 `FCPGrowth.py` `inputFile.txt` `outputFile.txt` `4` `0.5` `' '`

How to execute a fuzzy correlated 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 [16]:

```
import PAMI.fuzzyCorrelatedPattern.basic.FCPGrowth as alg

iFile = 'sample_Input.txt'  #specify the input temporal database <br>
minSup = 4  #specify the minSupvalue <br>    #specify the maxPerAllConfVa
seperator = ' ' #specify the seperator. Default seperator is tab space. <
minAllConf = 0.5
oFile = 'FuzzyCorrelatedPatterns.txt'   #specify the output file name<br>

obj = alg.FCPGrowth(iFile, minSup, minAllConf, seperator) #initialize the
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 p
```

```
Fuzzy Correlated Patterns Successfully generated using FCPGrowth algorith
ms
```

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

In [17]:

```
!cat FuzzyCorrelatedPatterns.txt
```

```
a.L : 5.3999999999999995 : 0.7714285714285714

b.L : 5.599999999999999 : 0.6222222222222221

b.L c.L : 4.6 : 0.5111111111111111

d.L : 6.199999999999999 : 0.6888888888888888

d.L c.L : 5.3999999999999995 : 0.6

c.L : 6.999999999999999 : 0.7777777777777777
```

The dataframe containing the patterns is shown below:

In [18]:

```
df
```

Out[18]:

	Patterns	Support
0	a.L	5.3999999999999995 : 0.7714285714285714\n
1	b.L	5.599999999999999 : 0.6222222222222221\n
2	b.L c.L	4.6 : 0.5111111111111111\n
3	d.L	6.199999999999999 : 0.6888888888888888\n
4	d.L c.L	5.3999999999999995 : 0.6\n
5	c.L	6.999999999999999 : 0.7777777777777777\n