

# Carbon

Jiayu Wang

## 3.Frequent Set

### 1. Load data

```
#load module data
import data
#create a dataset object and read data from file sample.txt
sample = data.DataSet()
#To read nominal data, you have to add argument 'nominal',
default is 'numeric'
#Read data from 'sample.txt'
sample.read('sample.txt', 'numeric')
#create train dataset and test dataset using 1:10 hold out
train,test = data.holdOut(sample,0.1)
```

### 2. Apriori

The algorithm only works for: numerical data and nominal data

Parameters:

minSupport: float(0,1)

minConf: float(0,1)

```
#load apriori module
from freqSet import apriori
#Find the frequent set with support bigger than 0.5
L, supportData = apriori.apriori(train.x, minSupport = 0.5)
#L contains the frequent sets and we can get exact supports from
supportData
L
[[frozenset({2.0}), frozenset({1.0}), frozenset({3.0})],
[frozenset({1.0, 2.0}), frozenset({1.0, 3.0})], []]
#Find the association rules from L with confidence bigger than
0.7
rules = apriori.generateRules(L, supportData, minConf=0.7)
frozenset({2.0}) --> frozenset({1.0}) conf: 0.9330453563714902
frozenset({1.0}) --> frozenset({2.0}) conf: 0.7811934900542495
frozenset({3.0}) --> frozenset({1.0}) conf: 0.8756756756756755
#rules contains the rules above
rules
[(frozenset({2.0}), frozenset({1.0}), 0.9330453563714902),
(frozenset({1.0}), frozenset({2.0}), 0.7811934900542495),
(frozenset({3.0}), frozenset({1.0}), 0.8756756756756755)]
```

### 3. FP-growth

The algorithm only works for: numerical data and nominal data

Parameters:

minSupport: int(1,inf)

```
#load fpGrowth module
from freqSet import fpGrowth
#Find the frequent set with frequency bigger than 50
freqSet = fpGrowth.fpGrowth(train.x,minSupport=50)
```

```
#freqSet contains the frequent sets
```

```
freqSet
```

```
[{2.0}, {2.0, 10.0}, {1.0, 2.0}, {6.0}, {8.0, 6.0}, {5.0, 6.0},  
{3.0, 6.0}, {4.0, 6.0}, {1.0, 6.0}, {10.0, 6.0}, {7.0}, {1.0,  
7.0}, {3.0, 7.0}, {5.0, 7.0}, {8.0, 7.0}, {4.0, 7.0}, {10.0,  
7.0}, {8.0}, {8.0, 4.0}, {8.0, 5.0}, {8.0, 3.0}, {8.0, 1.0},  
{8.0, 10.0}, {4.0}, {4.0, 5.0}, {3.0, 4.0}, {1.0, 4.0}, {10.0,  
4.0}, {5.0}, {3.0, 5.0}, {1.0, 5.0}, {10.0, 5.0}, {3.0}, {10.0,  
3.0}, {1.0, 3.0}, {1.0}, {1.0, 10.0}, {10.0}]
```

```
#fast start
```

```
import data
```

```
sample = data.DataSet()
```

```
sample.read('sample.txt')
```

```
train,test = data.holdOut(sample,0.1)
```

```
from imp import reload
```