NAME: Sejal Gurkhe

UID: 2019130017

TE Computer

Experiment 5

Aim: To apply Apriori algorithm to given dataset Association Rule miningwith WEKA.

Exercise 1:

```
[ ] import pandas as pd
    from mlxtend.frequent_patterns import apriori, association_rules
```

Excercise 1

```
[ ] df1Lst = [
              [1, 1, 0, 1, 0, 1],
              [1, 1, 1, 1, 1, 0],
              [1, 1, 1, 0, 1, 0],
              [1, 1, 0, 1, 0, 0],
    ]

    df1 = pd.DataFrame(df1Lst, columns=['A', 'B', 'C', 'D', 'E', 'K'])
    df1
```

	A	B	C	D	E	K
0	1	1	0	1	0	1
1	1	1	1	1	1	0
2	1	1	1	0	1	0
3	1	1	0	1	0	0

```
[ ] frq_items = apriori(df1, min_support = 0.6, use_colnames = True)
    print(frq_items)
    rules = association_rules(frq_items, metric ="lift", min_threshold = 0.6)
    print(rules)
```

```
   support   itemsets
0     1.00        (A)
1     1.00        (B)
2     0.75        (D)
3     1.00     (B, A)
4     0.75     (D, A)
5     0.75     (B, D)
6     0.75  (D, B, A)
```

	confidence	lift	leverage	conviction
0	1.00	1.0	0.0	inf
1	1.00	1.0	0.0	inf
2	1.00	1.0	0.0	inf
3	0.75	1.0	0.0	1.0
4	0.75	1.0	0.0	1.0
5	1.00	1.0	0.0	inf
6	1.00	1.0	0.0	inf
7	1.00	1.0	0.0	inf
8	0.75	1.0	0.0	1.0
9	1.00	1.0	0.0	inf
10	0.75	1.0	0.0	1.0
11	0.75	1.0	0.0	1.0

Association rules found:

{A} => {B}

{B} => {A}

{D} => {A}

{D} => {B}

{D, B} => {A}

{D, A} => {B}

{D} => {B, A}

Exercise 2:

1. Create a .arff file for given dataset.

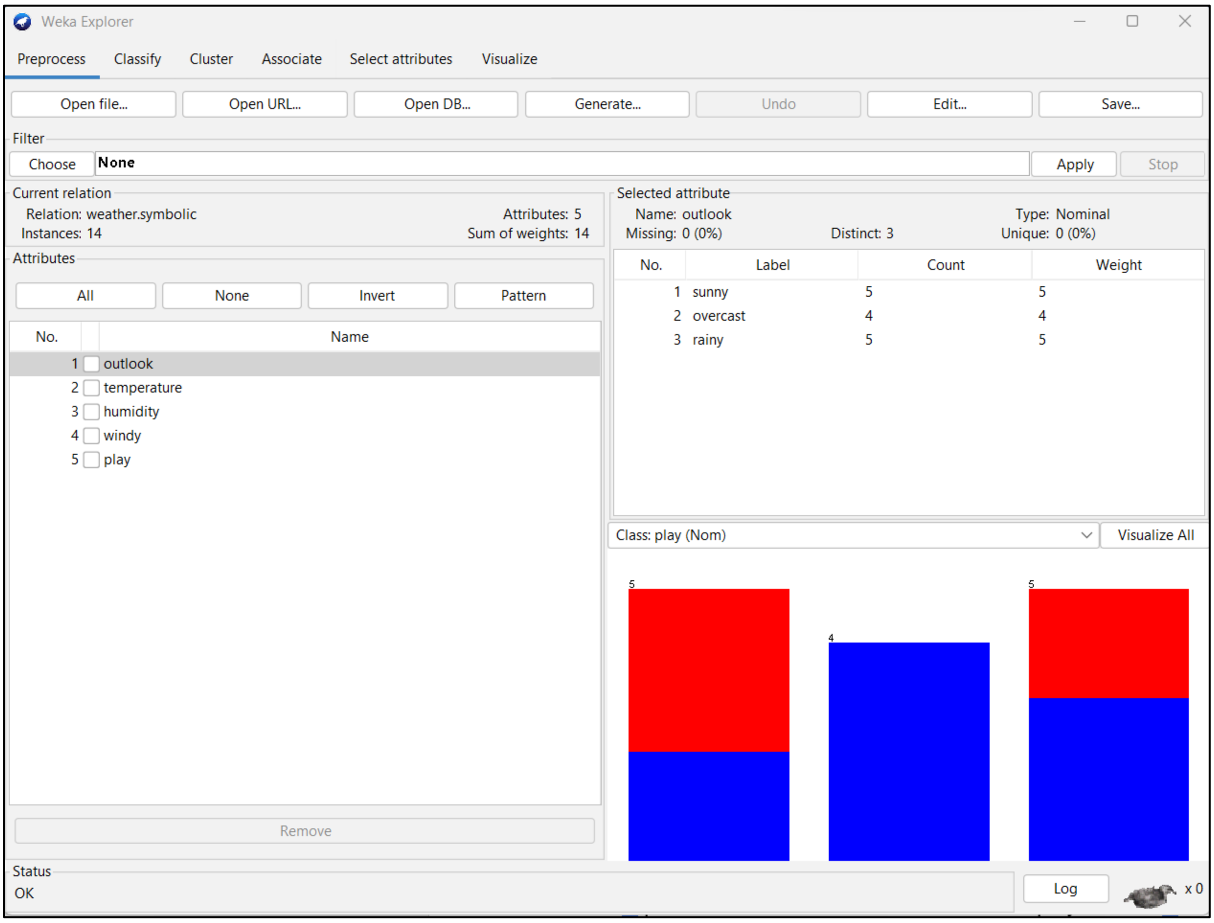
```
supermarket.arff - Notepad
File  Edit  Format  View  Help
@relation supermarket

@attribute A {1, 0}
@attribute B {1, 0}
@attribute C {1, 0}
@attribute D {1, 0}
@attribute E {1, 0}
@attribute K {1, 0}

@data
1, 1, 0, 1, 0, 1
1, 1, 1, 1, 1, 0
1, 1, 1, 0, 1, 0
1, 1, 0, 1, 0, 0
```

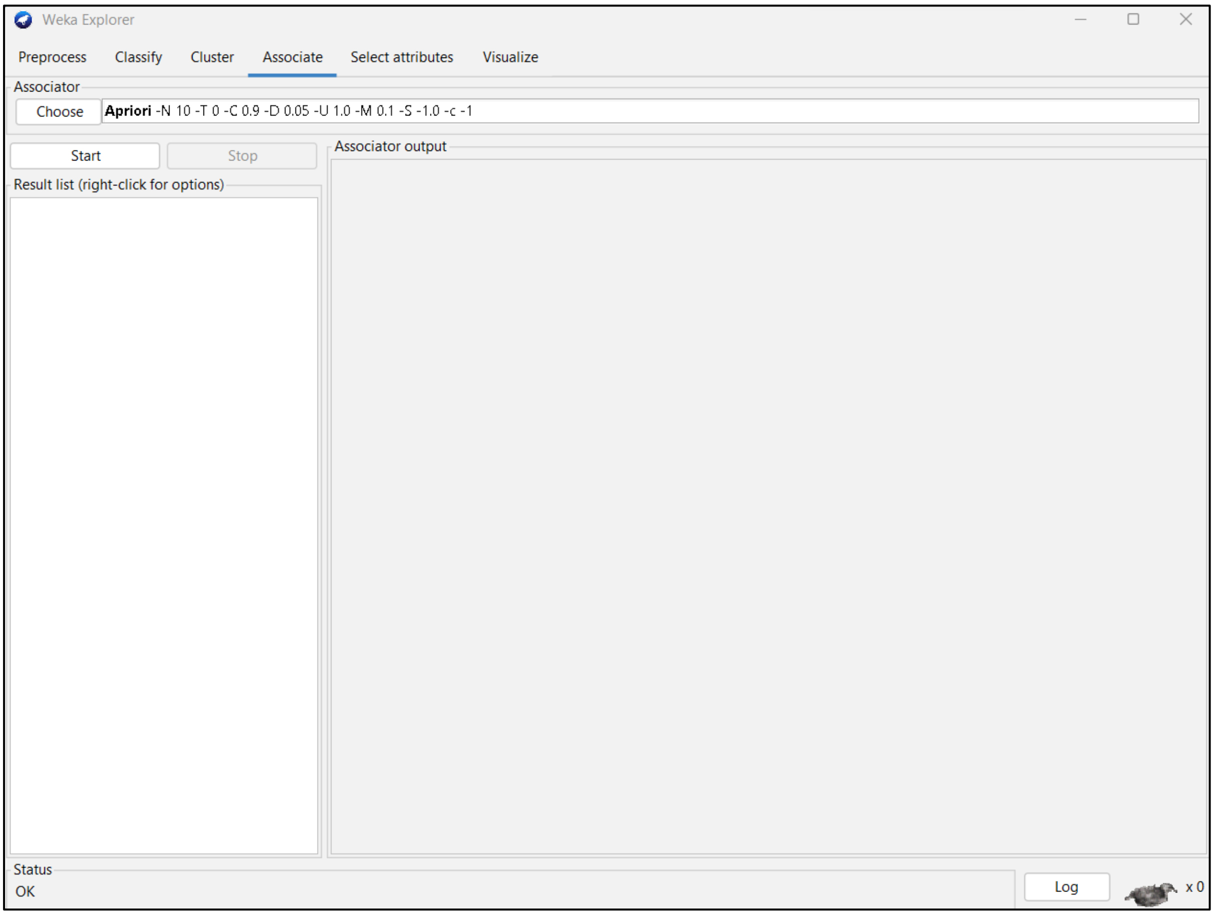
2. Load into WEKA and perform association rule mining.

```
1. B=1 4 ==> A=1 4    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
2. A=1 4 ==> B=1 4    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
3. D=1 3 ==> A=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
4. K=0 3 ==> A=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
5. D=1 3 ==> B=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
6. K=0 3 ==> B=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
7. B=1 D=1 3 ==> A=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
8. A=1 D=1 3 ==> B=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
9. D=1 3 ==> A=1 B=1 3    <conf:(1)> lift:(1) lev:(0) [0] conv:(0)
```

We observe that association rules that we determined using manual method, are exactly same as that of given by WEKA. I created and useda .arff file (See point 1)

Exercise 3:


Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Open file...
Open URL...
Open DB...
Generate...
Undo
Edit...
Save...
Filter
Choose
None
Apply
Stop
Current relation
Relation: weather.symbolic
Instances: 14
Attributes: 5
Sum of weights: 14
Selected attribute
Name: outlook
Missing: 0 (0%)
Distinct: 3
Type: Nominal
Unique: 0 (0%)
No. Label Count Weight
1 sunny 5 5
2 overcast 4 4
3 rainy 5 5
Attributes
All
None
Invert
Pattern
No. Name
1 outlook
2 temperature
3 humidity
4 windy
5 play
Class: play (Nom)
Visualize All
Remove
Status
OK
Log
x 0

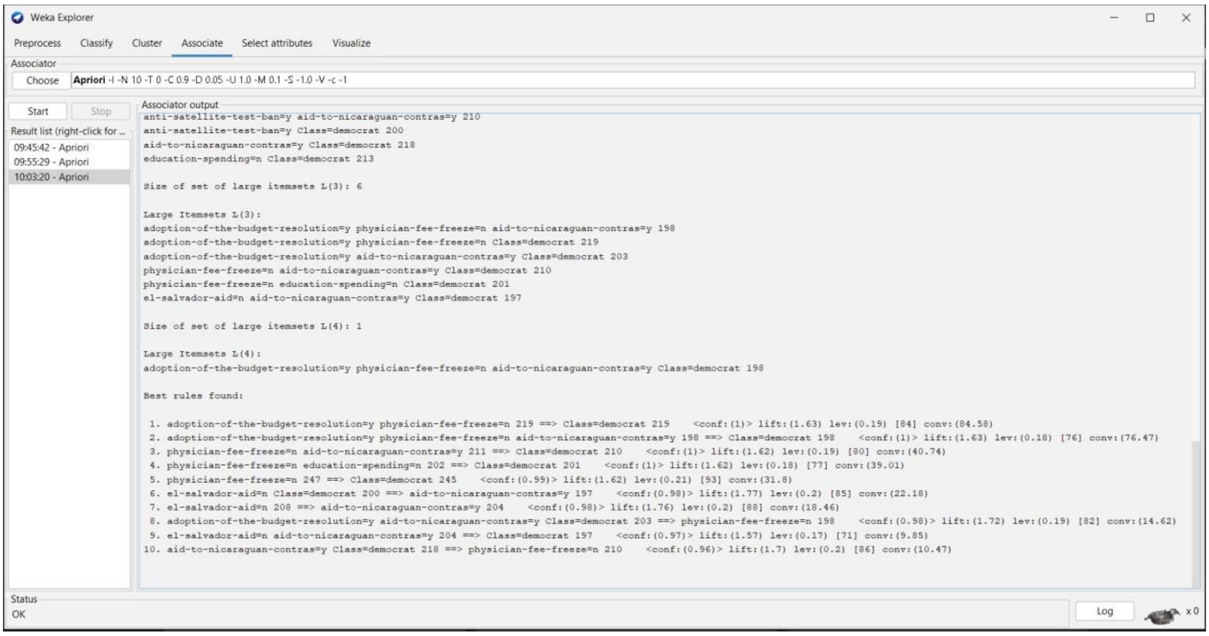

Weka Explorer
Preprocess
Classify
Cluster
Associate
Select attributes
Visualize
Associator
Choose
Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Start
Stop
Associator output
Result list (right-click for options)
Status
OK
Log
x 0

```
Apriori
=======

Minimum support: 0.15 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12

Size of set of large itemsets L(2): 47

Size of set of large itemsets L(3): 39

Size of set of large itemsets L(4): 6

Best rules found:

 1. outlook=overcast 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 2. temperature=cool 4 ==> humidity=normal 4    <conf:(1)> lift:(2) lev:(0.14) [2] conv:(2)
 3. humidity=normal windy=FALSE 4 ==> play=yes 4    <conf:(1)> lift:(1.56) lev:(0.1) [1] conv:(1.43)
 4. outlook=sunny play=no 3 ==> humidity=high 3    <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
 5. outlook=sunny humidity=high 3 ==> play=no 3    <conf:(1)> lift:(2.8) lev:(0.14) [1] conv:(1.93)
 6. outlook=rainy play=yes 3 ==> windy=FALSE 3    <conf:(1)> lift:(1.75) lev:(0.09) [1] conv:(1.29)
 7. outlook=rainy windy=FALSE 3 ==> play=yes 3    <conf:(1)> lift:(1.56) lev:(0.08) [1] conv:(1.07)
 8. temperature=cool play=yes 3 ==> humidity=normal 3    <conf:(1)> lift:(2) lev:(0.11) [1] conv:(1.5)
 9. outlook=sunny temperature=hot 2 ==> humidity=high 2    <conf:(1)> lift:(2) lev:(0.07) [1] conv:(1)
10. temperature=hot play=no 2 ==> outlook=sunny 2    <conf:(1)> lift:(2.8) lev:(0.09) [1] conv:(1.29)
```

Exercise 4:

Here, number of members of Democratic party are more in numberas compared to members of Republic party which ultimately increases the probability of their appearance in the most frequent item sets.

Hence, we see no member of republic party in the rules. Probably if weincrease the number of members of Republic party, we may find few entries in rules.



Exercise 5:

1. minConfidence 0.9:

```
Best rules found:
```

At minConf = 0.9 with minsupport 0.3, no rule is generated.

2. minConfidence 0.6:

```
Best rules found:

 1. biscuits=t 2605 ==> bread and cake=t 2083    <conf:(0.8)> lift:(1.11) lev:(0.04) [208] conv:(1.4)
 2. milk-cream=t 2939 ==> bread and cake=t 2337    <conf:(0.8)> lift:(1.1) lev:(0.05) [221] conv:(1.37)
 3. fruit=t 2962 ==> bread and cake=t 2325    <conf:(0.78)> lift:(1.09) lev:(0.04) [193] conv:(1.3)
 4. baking needs=t 2795 ==> bread and cake=t 2191    <conf:(0.78)> lift:(1.09) lev:(0.04) [179] conv:(1.29)
 5. frozen foods=t 2717 ==> bread and cake=t 2129    <conf:(0.78)> lift:(1.09) lev:(0.04) [173] conv:(1.29)
 6. vegetables=t 2961 ==> bread and cake=t 2298    <conf:(0.78)> lift:(1.08) lev:(0.04) [167] conv:(1.25)
 7. vegetables=t 2961 ==> fruit=t 2207    <conf:(0.75)> lift:(1.16) lev:(0.07) [311] conv:(1.41)
 8. fruit=t 2962 ==> vegetables=t 2207    <conf:(0.75)> lift:(1.16) lev:(0.07) [311] conv:(1.41)
 9. bread and cake=t 3330 ==> milk-cream=t 2337    <conf:(0.7)> lift:(1.1) lev:(0.05) [221] conv:(1.22)
10. bread and cake=t 3330 ==> fruit=t 2325    <conf:(0.7)> lift:(1.09) lev:(0.04) [193] conv:(1.19)
```

At minConf = 0.6 with minSupport 0.3, we can see generatedrules

Conclusion:

Association rule mining finds interesting associations and relationships among large sets of data items. This rule shows how frequently a itemset occurs in a transaction. Given a set of transactions, we can find rules that will predict the occurrence of an item based on the occurrences of other items in the transaction. Apriori algorithm allows us to mine the frequent itemset in order togenerate association rule between them. The main limitation is time required to hold a vast number of candidate sets with much frequent item sets, low minimum support or large item sets i.e. it is not an efficientapproach for large number of datasets.