

Day3-Data Mining Experiments

Experiment 1: Apriori and FP-Growth Algorithm

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

```

Associator
Choose FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1

Start Stop
Result list (right-click for ...)
09:48:25 - Apriori
09:48:34 - FPGrowth

Associator output
Instances: 6
Attributes: 5
Milk
Bread
Butter
Jam
Cheese

=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.4 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 12

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10
Size of set of large itemsets L(2): 21
Size of set of large itemsets L(3): 9

Best rules found:

1. Jam=no 3 ==> Milk=yes 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
2. Cheese=no 3 ==> Bread=yes 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
3. Butter=no 2 ==> Milk=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0] conv:(0.6)
4. Milk=no 2 ==> Butter=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0] conv:(0.6)
5. Milk=no 2 ==> Jam=yes 2 <conf:(1)> lift:(2) lev:(0.17) [1] conv:(1)
6. Butter=no 2 ==> Bread=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0] conv:(0.6)
7. Bread=no 2 ==> Butter=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0] conv:(0.6)
8. Bread=no 2 ==> Cheese=yes 2 <conf:(1)> lift:(2) lev:(0.17) [1] conv:(1)
9. Bread=yes Butter=no 2 ==> Milk=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0]
10. Milk=yes Butter=no 2 ==> Bread=yes 2 <conf:(1)> lift:(1.5) lev:(0.11) [0]

=== Run information ===

Scheme: weka.associations.FPGrowth -P 2 -I -1 -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1
Relation: market_basket
Instances: 6
Attributes: 5
Milk
Bread
Butter
Jam
Cheese

=== Associator model (full training set) ===

```

Experiment 2: Association Rule Analysis

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

The screenshot displays the Weka Classifier window. The 'Choose' dropdown is set to 'J48 -C 0.25 -M 2'. Under 'Test options', 'Cross-validation' is selected with 'Folds' set to 10. The 'Result list' on the left shows two entries: '09:51:22 - rules.ZeroR' and '09:52:10 - trees.J48', with the latter being selected. The 'Classifier output' pane on the right contains the following text:

```
=== Run information ===

Scheme:      weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:    student_performance
Instances:   8
Attributes:  4
              attendance
              study_hours
              internal_marks
              result
Test mode:   10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
-----

attendance = high: pass (3.0)
attendance = medium: pass (2.0)
attendance = low: fail (3.0)

Number of Leaves :    3
Size of the tree :    4

Time taken to build model: 0.01 seconds
```

Experiment 3: Bayes Classification and Decision Tree

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

- Step 1: Load the dataset.
- Step 2: Preprocess the data.
- Step 3: Apply the specified algorithm.
- Step 4: Analyze the results.
- Step 5: Interpret the outcome.

Output:

The screenshot displays the Weka Classifier window. The 'Choose' button is set to 'PART -C 0.25 -M 2'. Under 'Test options', 'Cross-validation' is selected with 'Folds' set to 10. The target attribute is '(Nom) loan_status'. The 'Result list' on the left shows three entries: '09:51:22 - rules.ZeroR', '09:52:10 - trees.J48', and '10:00:25 - rules.PART', with the last one selected. The 'Classifier output' pane on the right contains the following text:

```
=== Run information ===

Scheme:      weka.classifiers.rules.PART -C 0.25 -M 2
Relation:    loan_approval
Instances:   7
Attributes:  4
              income
              credit_score
              employment
              loan_status
Test mode:   10-fold cross-validation

=== Classifier model (full training set) ===

PART decision list
-----

employment = no: rejected (4.0/1.0)

: approved (3.0)

Number of Rules :      2

Time taken to build model: 0.01 seconds
```

Experiment 4: Diabetes Dataset Analysis using Regression

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

The screenshot displays the Weka Classifier window. The 'Classifier' dropdown is set to 'J48 -C 0.25 -M 2'. Under 'Test options', 'Cross-validation' is selected with 'Folds' set to 10. The 'Result list' on the left shows four entries, with '10:02:38 - trees.J48' selected. The 'Classifier output' pane on the right shows the following information:

```
=== Run information ===

Scheme:      weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:    diabetes
Instances:    7
Attributes:   4
              age
              bmi
              glucose
              diabetes
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
-----

age <= 30: no (3.0)
age > 30: yes (4.0)

Number of Leaves :    2

Size of the tree :    3

Time taken to build model: 0 seconds
```

Experiment 5: WEKA Implementation (Apriori & FP-Growth)

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

```
=== Run information ===

Scheme:      weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0
Relation:    apriori_lab
Instances:    5
Attributes:   6
              M
              O
              N
              K
              E
              Y

=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.5 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 10

Generated sets of large itemsets:

Size of set of large itemsets L(1): 10
Size of set of large itemsets L(2): 24
Size of set of large itemsets L(3): 19
Size of set of large itemsets L(4): 7
Size of set of large itemsets L(5): 1

Best rules found:

1. E=1 4 ==> K=1 4    <conf: (1)> lift: (1.25) lev: (0.16) [0] conv: (0.8)
2. K=1 4 ==> E=1 4    <conf: (1)> lift: (1.25) lev: (0.16) [0] conv: (0.8)
3. O=1 3 ==> K=1 3    <conf: (1)> lift: (1.25) lev: (0.12) [0] conv: (0.6)
```

Experiment 6: Decision Tree Prediction using WEKA

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

The screenshot displays the WEKA Classifier window. The 'Choose' dropdown is set to 'J48 -C 0.25 -M 2'. Under 'Test options', 'Cross-validation' is selected with 'Folds' set to 10. The 'Result list' shows two entries: '09:51:22 - rules.ZeroR' and '09:52:10 - trees.J48', with the latter being selected. The 'Classifier output' pane shows the following information:

```
=== Run information ===

Scheme:      weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:    student_performance
Instances:   8
Attributes:  4
              attendance
              study_hours
              internal_marks
              result
Test mode:   10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
-----

attendance = high: pass (3.0)
attendance = medium: pass (2.0)
attendance = low: fail (3.0)

Number of Leaves :    3

Size of the tree :    4

Time taken to build model: 0.01 seconds
```

Experiment 7: ARFF Dataset Creation and Rule Generation

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

```
Associator
Choose Apriori -N 10 -T 0 -C 0.6 -D 0.05 -U 1.0 -M 0.333 -S -1.0 -c -1

Start Stop
Result list (right-click for ...)
13:25:45 - Apriori
13:25:57 - FPGrowth
17:34:44 - Apriori

Associator output
Buns
Ketchup
Coke
Chips
=== Associator model (full training set) ===

Apriori
=====

Minimum support: 0.55 (3 instances)
Minimum metric <confidence>: 0.6
Number of cycles performed: 9

Generated sets of large itemsets:

Size of set of large itemsets L(1): 6
Size of set of large itemsets L(2): 7
Size of set of large itemsets L(3): 4
Size of set of large itemsets L(4): 1

Best rules found:

1. Chips=yes 4 ==> Buns=no 4 <conf:(1)> lift:(1.5) lev:(0.22) [1] conv:(1.33)
2. Buns=no 4 ==> Chips=yes 4 <conf:(1)> lift:(1.5) lev:(0.22) [1] conv:(1.33)
3. Coke=yes 3 ==> Buns=no 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
4. Coke=yes 3 ==> Ketchup=no 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
5. Coke=yes 3 ==> Chips=yes 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
6. Ketchup=no Coke=yes 3 ==> Buns=no 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
7. Buns=no Coke=yes 3 ==> Ketchup=no 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
8. Buns=no Ketchup=no 3 ==> Coke=yes 3 <conf:(1)> lift:(2) lev:(0.25) [1] conv:(1.5)
9. Coke=yes 3 ==> Buns=no Ketchup=no 3 <conf:(1)> lift:(2) lev:(0.25) [1] conv:(1.5)
10. Ketchup=no Chips=yes 3 ==> Buns=no 3 <conf:(1)> lift:(1.5) lev:(0.17) [1] conv:(1)
```

```
Associator
Choose FPGrowth -P 2 -I -1 -N 20 -T 0 -C 0.6 -D 0.05 -U 1.0 -M 0.333

Start Stop
Result list (right-click for ...)
13:25:45 - Apriori
13:25:57 - FPGrowth
17:34:44 - Apriori
17:36:18 - FPGrowth

Associator output
=== Run information ===

Scheme: weka.associations.FPGrowth -P 2 -I -1 -N 20 -T 0 -C 0.6 -D 0.05 -U 1.0 -M 0.333
Relation: market_basket
Instances: 6
Attributes: 5
HotDogs
Buns
Ketchup
Coke
Chips
=== Associator model (full training set) ===

FPGrowth found 5 rules (displaying top 5)

1. [HotDogs=no]: 2 ==> [Buns=no]: 2 <conf:(1)> lift:(1.5) lev:(0.11) conv:(0.67)
2. [Chips=no]: 2 ==> [Coke=no]: 2 <conf:(1)> lift:(2) lev:(0.17) conv:(1)
3. [Ketchup=no]: 4 ==> [Buns=no]: 3 <conf:(0.75)> lift:(1.13) lev:(0.06) conv:(0.67)
4. [Buns=no]: 4 ==> [Ketchup=no]: 3 <conf:(0.75)> lift:(1.13) lev:(0.06) conv:(0.67)
5. [Coke=no]: 3 ==> [Chips=no]: 2 <conf:(0.67)> lift:(2) lev:(0.17) conv:(1)
```


Experiment 8: Rule-based vs Decision Tree Classification

Aim:

To implement and analyze the given data mining technique using appropriate datasets.

Algorithm:

Step 1: Load the dataset.

Step 2: Preprocess the data.

Step 3: Apply the specified algorithm.

Step 4: Analyze the results.

Step 5: Interpret the outcome.

Output:

The screenshot displays the Weka Classifier window. The 'Classifier' dropdown is set to 'J48 -C 0.25 -M 2'. Under 'Test options', 'Cross-validation' is selected with 'Folds' set to 10. The 'Result list' shows two entries: '09:51:22 - rules.ZeroR' and '09:52:10 - trees.J48', with the latter selected. The 'Classifier output' pane shows the following information:

```
=== Run information ===

Scheme:      weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:    student_performance
Instances:   8
Attributes:  4
              attendance
              study_hours
              internal_marks
              result
Test mode:   10-fold cross-validation

=== Classifier model (full training set) ===

J48 pruned tree
-----

attendance = high: pass (3.0)
attendance = medium: pass (2.0)
attendance = low: fail (3.0)

Number of Leaves :    3

Size of the tree :    4

Time taken to build model: 0.01 seconds
```