Yuan Wang

440 Data Mining

Prof. Jiebo Luo

Homework 4

**7.5**

**Step 1:**

For each frequent item, construct its conditional pattern-base and conditional FP-Tree

Recursively create conditional FP-Tree until the resulting FP-Tree is empty

**Step 2:**

For each frequent itemset X, scan infrequent item sets Z, which contains X.

Find Y contains Z which are not in X

If Y is a frequent item set, then calculate

(P(X | Y )+P(Y | X))/2 = (sup(Z)/sup(Y )+sup(Z)/sup(X))/2 to determine whether X and Y are negatively correlated

**7.9**

The distance measure Pat Dist is a valid distance metric.

When , *Pat\_Dist () = 0*

When , *Pat\_Dist () > 0*

It also has

*Pat\_Dist () = Pat\_Dist (),*

*Pat\_Dist () + Pat\_Dist () Pat\_Dist ()*

**7.10**

We can use clustering method based on δ-clusters. First, partitioning the dataset to N subsets, find the local representatives which δ-cover the most other patterns locally. Then find the global representative which δ-cover based on local combined.

**8.3**

Method (a) prune the rule, it may remove any precondition of the rule.

Method (b) prune the subtree, it may remove the whole subtree.

Method (a) is less restrictive.

**8.5**

Use the Rainforest algorithm for this case. Assume there are C class labels. Scan the whole databased to build up AVC-list for each of the 50 attributes. The size of each AVC-list is 100\*C. the total size of the AVC-set is 100\*C\*50, which fits 512MB memory. The other AVC-set takes less computation because there is less attributes available. We can compute the AVS-set for nodes at the same level of the tree in parallel to reduce the scan times.

**8.7**

**(a)**

The attribute selection measure should count each of tuples.

Determine the most common class among tuples

**(b)**

**(c)**

;

Naïve Bayesian predicts “junior”

**8.12**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Tuple | Class | Prob | TP | FP | TN | FN | TPR | RPR |
| 1 | P | 0.95 | 1 | 0 | 5 | 4 | 0.2 | 0 |
| 2 | N | 0.85 | 1 | 1 | 4 | 4 | 0.2 | 0.2 |
| 3 | P | 0.78 | 2 | 1 | 4 | 3 | 0.4 | 0.2 |
| 4 | P | 0.66 | 3 | 1 | 4 | 2 | 0.6 | 0.2 |
| 5 | N | 0.6 | 3 | 2 | 3 | 2 | 0.6 | 0.4 |
| 6 | P | 0.55 | 4 | 2 | 3 | 1 | 0.8 | 0.4 |
| 7 | N | 0.53 | 4 | 3 | 2 | 1 | 0.8 | 0.6 |
| 8 | N | 0.52 | 4 | 4 | 1 | 1 | 0.8 | 0.8 |
| 9 | N | 0.51 | 4 | 5 | 0 | 1 | 0.8 | 1 |
| 10 | P | 0.4 | 5 | 5 | 0 | 0 | 1 | 1 |

**Graphical user interface

Description automatically generated**

**8.14**

Given that Alpha is 1%, we accept the null hypothesis because p\_value = 2.3% > 1%

I also use the scipy to test but it got slightly different value (though, it does not impact the conclusion), but I wonder why they are different.

Graphical user interface, text, application, email

Description automatically generated