Data Mining Assignment-1

Mudit Pandey 2014A7PS017H
Vedic Sharma 2014B4A7424H

April 29, 2017

Contents

1 Languages used

The entire code was written in C++.

2 Pre-processing

We process the data as follows:

1. Treat each attribute answer (yes / no) as an item
2. Treat each candidate class (republican / democrat) as an item
3. For each candidate, consider a transaction
   (a) If the attribute value is yes, then an item corresponding to that 'yes' is inserted in the transaction.
   (b) Follow a similar approach if the attribute value is 'no'.
   (c) If the attribute value is unanswered (?), then no item is inserted.

Heres the key for each of the possible answers / class of each candidate

Item Number	**Corresponding attribute**
1	handicapped-infants=n
2	handicapped-infants=y
3	water-project-cost-sharing=n
4	water-project-cost-sharing=y
5	adoption-of-the-budget-resolution=n
6	adoption-of-the-budget-resolution=y
7	physician-fee-freeze=n
8	physician-fee-freeze=y
9	el-salvador-aid=n
10	el-salvador-aid=y
11	religious-groups-in-schools=n
12	religious-groups-in-schools=y
13	anti-satellite-test-ban=n
14	anti-satellite-test-ban=y
15	aid-to-nicaraguan-contras=n
16	aid-to-nicaraguan-contras=y
17	mx-missile=n
18	mx-missile=y

19	immigration=n
20	immigration=y
21	synfuels-corporation-cutback=n
22	synfuels-corporation-cutback=y
23	education-spending=n
24	education-spending=y
25	superfund-right-to-sue=n
26	superfund-right-to-sue=y
27	crime=n
28	crime=y
29	duty-free-exports=n
30	duty-free-exports=y
31	export-administration-act-south-africa=n
32	export-administration-act-south-africa=y
33	Class=republican
34	Class=democrat

A republican candidate who only answered to spending for education will have a transaction basket like

{24, 33}

From here we can apply Apriori algorithm to find interesting rules.

3 Compilation and Execution

Run the following commands in the terminal:-

1. g++ -std=c++11 apriori.cpp
2. ./a.out

4 Support and Confidence Values

We have generated rules for the following:-

1. Support=0.30 Confidence value=0.98
   No.of rules=465

2. Support=0.35 Confidence value=0.95
   No.of rules=287
3. Support=0.40 Confidence value=0.90
   No.of rules=193
4. Support=0.40 Confidence value=0.98
   No.of rules=37
5. Support=0.40 Confidence value=1.00
   No.of rules=7
6. Support=0.45 Confidence value=0.95
   No.of rules=11
7. Support=0.50 Confidence value=0.95
   No.of rules=2

The output is generated in a file which list all frequent item sets with their support values and the set of rules with their confidence values.