2018 Scheme

1	N	T							

NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY

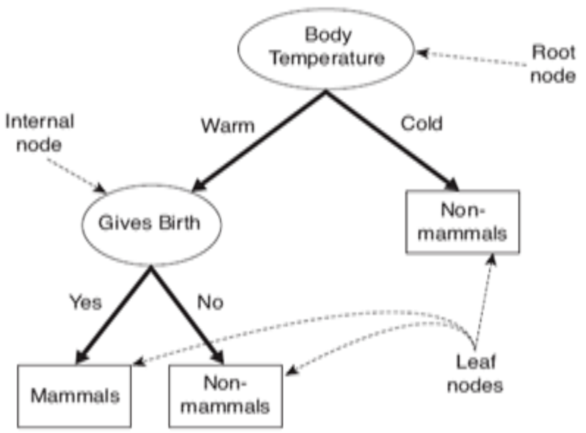
An Autonomous Institution Approved by UGC/AICTE/Govt. of Karnataka
Accredited by NBA (Tier – I) and NAAC 'A+' Grade
Affiliated to Visvesvaraya Technological University, Belagavi
Post Box No. 6429, Yelahanka, Bengaluru – 560 064, Karnataka, India

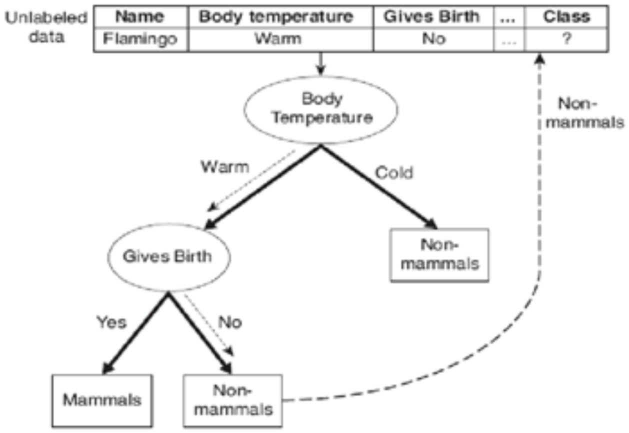
KNOWLEDGE • CHARACTER • UNITY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MID SEMESTER EXAMINATION-II

Course Title with code	Data Mining, 18CS54	**Maximum Marks**	**30 Marks**
Date and Time	30/12/2021, 9.30am to 10.30am		
Course Instructor(s)	Dr. Vijaya Shetty S, Dr. Sujata Joshi, Dr. Vani V		
	SCHEME AND SOLUTION		

Q. No	**Question**	**MAX MARKS**
1. a	**Solution** ● We can solve a classification problem by asking a series of carefully crafted questions about the attributes of the test record. ● Each time we receive an answer? a follow-up question is asked until we reach a conclusion about the class label of the record. ● The series of questions and their possible answers can be organized in the form of a decision tree, which is a hierarchical structure consisting of nodes and directed edges. ● The tree has three types of nodes: o A root node that has no incoming edges and zero or more outgoing edges. o Internal nodes, each of which has exactly one incoming edge and two or more outgoing edges. o Leaf or terminal nodes, each of which has exactly one incoming edge and no outgoing edges. ● In a decision tree, each leaf node is assigned a class label. The non-terminal nodes, which include the root and other internal nodes, contain attribute test conditions to separate records that have different characteristics.  A decision tree for the mammal classification problem. ● **Classifying a test record** is straightforward once a decision tree has been constructed. Starting from the root node, we apply the test condition to the record and follow the appropriate branch based on the outcome of the test.	4M



Classifying an unlabeled vertebrate. The dashed lines represent the outcomes of applying various attribute test conditions on the unlabeled vertebrate. The vertebrate is eventually assigned to the Non-mammal class.

2M

1. b

Scheme:
Coverage:2m
Accuracy:2m
Solution:
rule: **(Give Birth = yes) ∧ (Blood Type = warm) → Mammals** 2M

$$\text{Coverage}(r) = \frac{|A|}{|D|}$$

=6*100/20 =30%

$$\text{Accuracy}(r) = \frac{|A \cap y|}{|A|},$$

=6*100/6=100% 2M

1. c 5

Normalization: 2m
Euclidian distance:2m
Prediction with result:1m
Solution :

After standardizing the input attributes Height and weight between 0 and 1 range using min-max normalization.

$$v_i' = \frac{v_i - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A.$$

Query Eample: Height=0.2 weight=0.3 k=3					
Height	Weight	Euclidian distance from query example $d(p,q) = \sqrt{\Sigma(p_i - q_i)^2}$	Rank min distance	Distance included	Class
0	0	0.36	4	no	
1	0.6	0.85	7	no	
0.6	1	0.81	6	no	
0.4	0.7	0.45	5	no	
0.3	0.2	0.14	2	no	Normal
0.5	0.3	0.3	3	yes	Underweight
0.1	0.4	0.14	1	yes	Normal

- Use simple majority of the category of nearest neighbors as the prediction value of the query instance

	• We have 2 Normal and 1 Underweight, since 2>1, we conclude that the person in the query example with height = 170cm and weight=57kg is included in Normal category	
2. a	Consider the following 5 transactions and 6 items. Shown in Table 2.1 With the help of **Apriori algorithm find the association rules** with **50% support and 75% confidence**	8

Table 2.a Dataset

TID	Items Bought
1	A,B,C,D
2	A,B,D
3	A,E,F
4	A,D,E
5	B,D,E

Solution

First find L_1. 50% support requires that each frequent item appear in at least three transactions. Therefore, L_1 is given by:

A	4
B	3
D	4
E	3

The candidate 2-itemsets or C_2 therefore has six pairs. These pairs and their frequencies are:

A,B	2
A,D	3
A,E	2
B,D	3
B,E	1
D,E	2

L_2 has only two frequent item pairs {A, D} and {B, D}. After these two frequent pairs, there are no candidate 3-itemsets (since we do not have two 2-itemsets that have the same first item.

The two frequent pairs lead to the following possible rules:

A→ D
D→ A
B → D
D→ B

The confidence of these rules is obtained by dividing the support for both items in the rule by the support of the item on the left-hand side of the rule.

The confidence of the four rules therefore are

3/4 = 75% (A → D)
3/4 = 75% (D → A)
3/3 = 100% (B → D)
3/4 = 75% (D → B)

Since all of them have a minimum 75% confidence, they all qualify as association rules.

Marking Scheme:
L1 : 2 Marks, frequency and support calculation
C2 : 1 Mark,
L2 : 2 Marks, frequency and support calculation
Confidence calculation: 2 Marks
Conclusion: 1 Mark

2. b Write the pseudocode for the frequent itemset generation part of the Apriori algorithm. (4)

Solution:

Algorithm 6.1 Frequent itemset generation of the *Apriori* algorithm.

```
k = 1.
F_k = { i | i ∈ I ∧ σ({i}) ≥ N × minsup}.   {Find all frequent 1-itemsets}
repeat
   k = k + 1.
   C_k = apriori-gen(F_{k−1}).   {Generate candidate itemsets}
   for each transaction t ∈ T do
      C_t = subset(C_k, t).   {Identify all candidates that belong to t}
      for each candidate itemset c ∈ C_t do
         σ(c) = σ(c) + 1.   {Increment support count}
      end for
   end for
   F_k = { c | c ∈ C_k ∧ σ(c) ≥ N × minsup}.   {Extract the frequent k-itemsets}
until F_k = ∅
Result = ⋃ F_k.
```

Marking Scheme:
4 marks for writing down all the steps and partial marks can be given based on the writeup.

2. c (3)

Transaction ID	Items Bought
1	{Milk, Beer, Diapers}
2	{Bread, Butter, Milk}
3	{Milk, Diapers, Cookies}
4	{Bread, Butter, Cookies}
5	{Beer, Cookies, Diapers}
6	{Milk, Diapers, Bread, Butter}
7	{Bread, Butter, Diapers}
8	{Beer, Diapers}
9	{Milk, Diapers, Bread, Butter}
10	{Beer, Cookies}

Consider the market basket transactions shown in Table 2.c to answer (a) and (b)
(a) What is the maximum number of association rules that can be extracted from this data (including rules that have zero support)?
(b) What is the maximum size of frequent itemsets that can be extracted (assuming minsup > 0)?

Solution:

(a) Milk, Beer, Diaper, Butter, Cookies, Bread => 6 items
d = 6, $3^d - 2^{d+1} + 1$ = **602 rules**

(b) Maximum size frequent itemset extracted from the given with minsup>0: **4** {Milk, Diapers, Bread , Butter}.

Marking Scheme

(a) 2 Marks
(b) 1 Mark

3. a (6)

Scheme
i. Overall Entropy : 1 mark
Info Gain(Temperature)=2 marks
Info Gain(Wind)=2 marks

ii. Best attribute - 0.5 mark
iii. Attribute used as the first split – 0.5 mark

Solution

i. Entropy at a given node t:

$$Entropy(t) = -\sum_j p(j\mid t)\log p(j\mid t)$$

Where $p(j \mid t)$ is the relative frequency of class j at node

- Overall Entropy Entropy(Temperature)

No. of positive examples = 9, No. of negative examples= 5
Overall Entropy= -9/14 log 9/14 – 5/14 log 5/14 = 0.940

- Entropy(Temperature)

  The counts for this attribute are:

	Hot	Mild	Cool
No	2	2	1
Yes	2	4	3

Entropy(Temperature=Hot) = -2/4 log 2/4 – 2/4 log 2/4 = 1.00
Entropy(Temperature=Mild) = -2/6 log 2/6 – 4/6 log 4/6 =0.918
Entropy(Temperature=Cool) = -1/4 log 1/4 – 3/4 log 3/4 = 0.811
Entropy(Temperature) = 4/14 * 1.00 + 6/14 *0.918 4/14 * 0.811 = 0.911
Info gain(Temperature)= Overall Entropy- Entropy(Temperature) = 0.940-0.911= 0.029

- Entropy(Wind)

  The counts for this attribute are:

	Weak	Strong
No	2	3
Yes	6	3

Entropy(Wind= Weak) = -2/8 log 2/8 – 6/8 log 6/8 = 0.811
Entropy(Wind=Strong)) = -3/6 log 3/6 – 3/6 log 3/6 = 1.00
Entropy(Wind) = 8/14 * 0.811 + 6/14 * 1.00 = 0.892
Info gain(Wind)= Overall Entropy- Entropy(Wind) = 0.940-0.892= 0.048

ii. According to the information gain measure , the best split among temperature and wind is Wind as te info gain of Wind is 0.048 which is greater than that of the temperature which is 0.029

iii. The decision tree algorithm uses the attribute " Wind" as the first split

3. b

a) 2.5m

$$\text{FOIL's information gain} = p_1 \times \left(\log_2 \frac{p_1}{p_1+n_1} - \log_2 \frac{p_0}{p_0+n_0}\right)$$

Assume the initial rule is $\emptyset \longrightarrow +$. This rule covers $p_0 = 100$ positive examples and $n_0 = 400$ negative examples.

The rule R_1 covers $p_1 = 4$ positive examples and $n_1 = 1$ negative example. Therefore, the FOIL's information gain for this rule is

P0=100,n0=400
P1=4,n1=1

$$4 \times \left(\log_2 \frac{4}{5} - \log_2 \frac{100}{500}\right) = 8.$$

The rule R_2 covers $p_1 = 30$ positive examples and $n_1 = 10$ negative example. Therefore, the FOIL's information gain for this rule is

$$30 \times \left(\log_2 \frac{30}{40} - \log_2 \frac{100}{500}\right) = 57.2.$$

According to Foil's information gain, R2 is the best rule and R1 is the worst rule.

b)

The Laplace measure. $\text{Laplace} = \frac{f_+ + 1}{n + k}$

Answer:

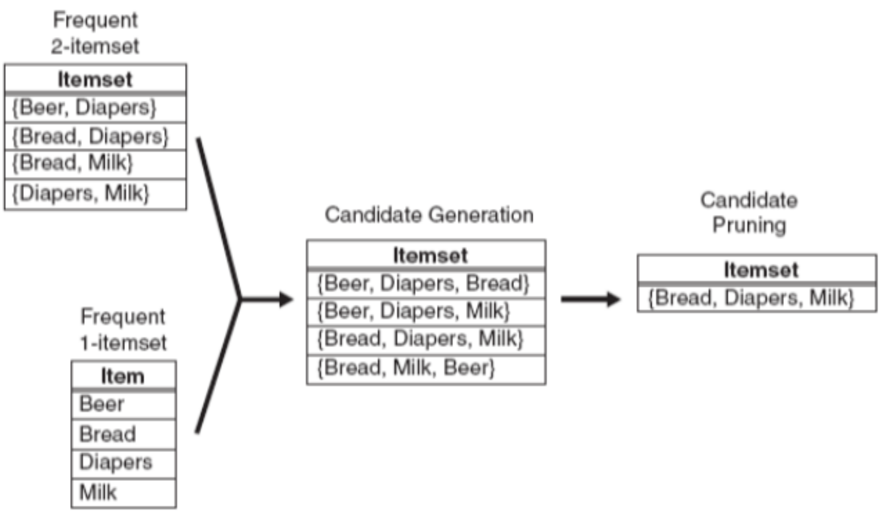
The Laplace measure of the rules are 71.43% (for R_1) and 73.81% (for R_2) Therefore R_2 is the best candidate and R_1 is the worst candidate according to the Laplace measure.

2.5m

3. c

$F_{k-1} \times F_1$ Method of candidate generation and pruning.

This is a method for candidate generation. The idea here is to extend each frequent $(k-1)$-itemset with other frequent items. As shown in the figure , we can see that a frequent 2-itemset such as {Beer, Diapers} can be augmented with a frequent item such as Bread to produce a candidate 3-itemset {Beer, Diapers, Bread}. This method will produce $O(|F_{k-1}| \times |F_1|)$ candidate k-itemsets, where $|F_j|$ is the number of frequent j-itemsets. 2m



Generating and pruning candidate k-itemsets by merging a frequent $(k-1)$-itemset with a frequent item. Note that some of the candidates are unnecessary because their subsets are infrequent.

2m

- The procedure is complete because every frequent k-itemset is composed of a frequent (k - 1)-itemset and a frequent 1-itemset.
- Avoid generating duplicate candidates by ensuring that the items in each frequent itemset are kept sorted in their lexicographic order. Each frequent (k-1)-itemset X is then extended with frequent items that are lexicographically larger than the items in X.
- For example, the itemset {Bread, Diapers} can be augmented with {Milk} since Milk is lexicographically larger than Bread and Diapers. However, we should not augment {Diapers, Milk} with {Bread} nor {Bread, Milk} with {Diapers} because they violate the

	lexicographic ordering condition. • For every candidate k-itemset that survives the pruning step, every item in the candidate must be contained in at least k - 1 of the frequent (k - 1)-itemsets. Otherwise, the candidate is guaranteed to be infrequent.		