COMP1942 Exploring and Visualizing Data (Spring Semester 2016)
Midterm Examination (Answer Sheet)
Date: 17 March, 2016 (Thu)
Time: 9:00-10:15
Duration: 1 hour 15 minutes

Student ID:_____ Student Name:_____

Seat No. :_____

Instructions:
(1) Please answer **all** questions in **Part A** in this paper.
(2) You can **optionally** answer the bonus question in **Part B** in this paper. You can obtain additional marks for the bonus question if you answer it correctly.
(3) The total marks in Part A are 100.
(4) The total marks in Part B are 10.
(5) The total marks you can obtain in this exam are 100 only.
If you answer the bonus question in Part B correctly, you can obtain additional marks.
But, if the total marks you obtain from Part A and Part B are over 100, your marks will be truncated to 100 only.
(6) You can use a calculator.

Answer Sheet

Part	Question	Full Mark	Mark
A	Q1	20	
	Q2	20	
	Q3	20	
	Q4	20	
	Q5	20	
Total (Part A)		100	
B	Q6 (OPTIONAL)	10	
Total (Parts A and B)		100	

Part A (Compulsory Short Questions)

Q1 (20 Marks)

(a) (i) support = 2

(ii) confidence = 2/3 = 66.7%

(iii) expected confidence of the consequent of the rule = 3/5
lift ratio of the rule = 66.7/60 = 1.11

(iv) freq. itemsets
= { {A}, {C}, {D}, {E},
{A, C}, {A, D}, {A, E}, {C, E}, {D, E},
{A, C, E}, {A, D, E} }

(b)

A	B	C	D	E
1	1	1	0	0
1	1	1	1	0
1	1	0	1	0

Q2 (20 Marks)

The given tree does not contain a single path.

Conditional FP-tree on "a" *(Count(a) = 7)*

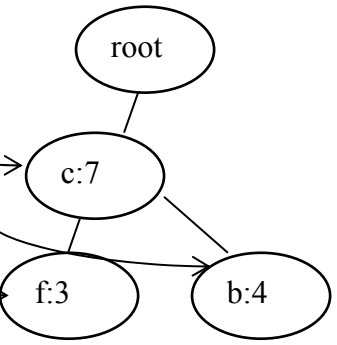
(f:3, c:3, a:3) => (c:3, f:3, a:3)
(c:4, b:4, a:4) => (c:4, b:4, a:4)

Item	
f	3
c	7
b	4

⟹

Item	
c	7
b	4
f	3

Item	
c	7
b	4
f	3



Conditional FP-tree on "af" *(Count(af) = 3)*

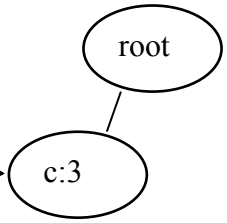
(c:3, f:3) => (c:3, f:3)

Item	
c	3

⟹

Item	
c	3

Item	
c	3



afc : 3

Conditional FP-tree on "ab" *(Count(ab) = 4)*

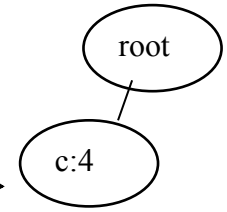
(c:4, b:4) => (c:4, b:4)

Item	
c	4

⟹

Item	
c	4

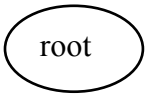
Item	
c	4



abc : 4

Conditional FP-tree on "ac" *(Count(ac) = 7)*

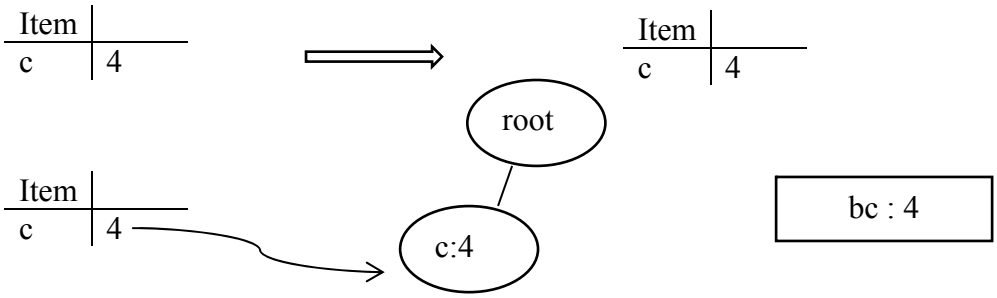
(c:7) => (c:7)



Q2 (Continued)

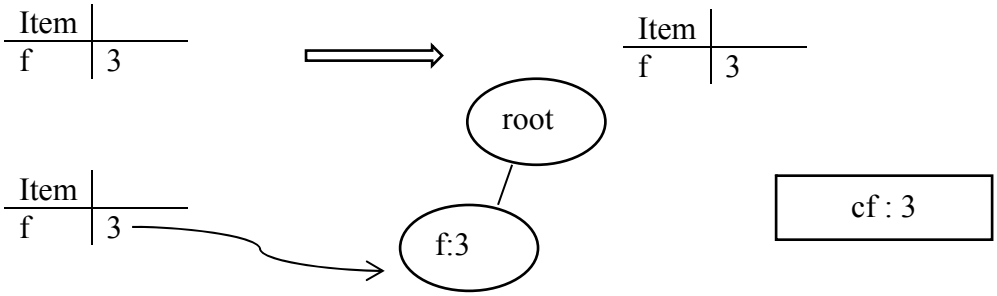
Conditional FP-tree on "b" *(Count(b) = 8)*

(c:4, b:4) => (c:4, b:4)
(b:4) => (b:4)



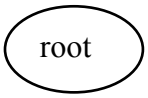
Conditional FP-tree on "c" *(Count(c) = 9)*

(f:3, c:3) => (f:3, c:3)
(c:6) => (c:6)



Conditional FP-tree on "f" *(Count(f) = 10)*

(f:10) => (f:10)



Frequent Patterns = a, af, afc, ab, abc, ac, b, bc, c, cf, f.

Q2 (Continued)

Q3 (20 Marks)
(a)

No.
We know that the whole dataset can be split into two clusters, {a, b} and {c, d, e}.
Consider cluster {c, d, e}.
We do not know the hierarchy for points c, d, and e.
We need two kinds of additional information, D({c}, {e}) and D({d}, {e}) to draw the dendrogram.

(b)(i) (1)

center = (1.5, 3)

(2)

center = (7, 8)

(ii) (1)

center = (4.8, 6)

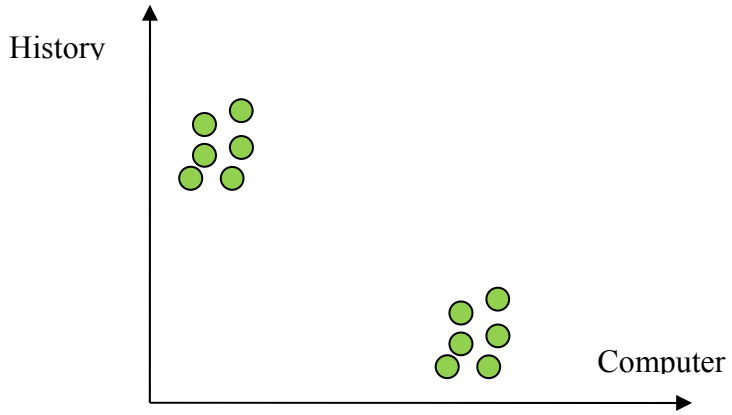
(2)

center = (4.25, 5.25)
(Note: center = (4.25, 5.5) is not correct.)

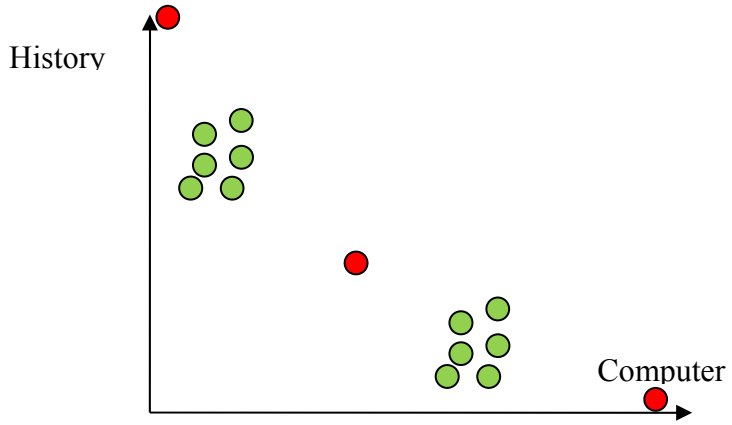
Q4 (20 Marks)
(a)

No.

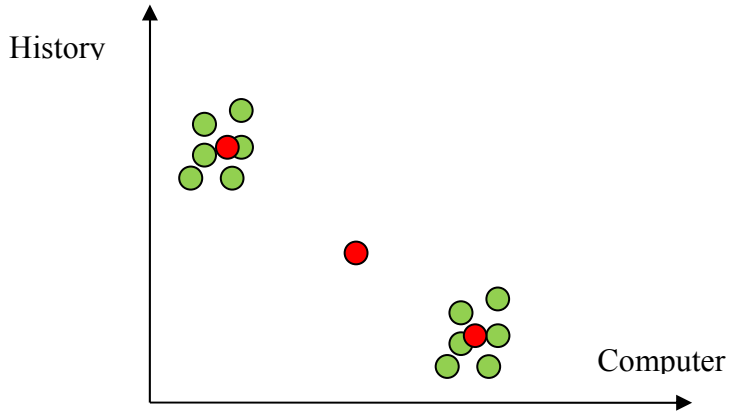
Consider the following example.



Suppose that k is set to 3 and the 3 starting means (denoted by 3 red dots) are shown as follows.



At the beginning, the middle "red" dot has some points associated with it. But, after a lot of iterations, the 3 red means become the following where the "middle" red dot has no point associated with it.



Q4 (Continued)

(b)(i)

- Make initial guesses for the means $m_1, m_2, \ldots, m_k$
- Until Interrupted
  - Acquire the next example x
  - If m_i is closest to x,
    - replace mi by $m_i + a(x - m_i)$

(ii)

$$
\begin{aligned}
m_n &= m_{n-1} + a(x_n - m_{n-1}) \\
&= (1-a)m_{n-1} + ax_n \\
&= (1-a)[(1-a)m_{n-2} + ax_{n-1}] + ax_n \\
&= (1-a)^2 m_{n-2} + (1-a)ax_{n-1} + ax_n \\
&= (1-a)^2[(1-a)m_{n-3} + ax_{n-2}] + (1-a)ax_{n-1} + ax_n \\
&= (1-a)^3 m_{n-3} + (1-a)^2 ax_{n-2} + (1-a)ax_{n-1} + ax_n \\
&= \ldots \\
&= (1-a)^n m_0 + \sum_{p=1}^{n} (1-a)^{n-p} a x_p
\end{aligned}
$$

$X = (1-a)^n$

$Y = (1-a)^{n-p}a$

Q4 (Continued)

Q5 (20 Marks)

(a)(i)

Info(T) = $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

For attribute Has_Bicycle,

Info(T_{no})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(T_{yes})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(Has_Bicycle, T)= $\frac{1}{2}Info(T_{no})+\frac{1}{2}Info(T_{yes})=1$

SplitInfo(Has_Bicycle)= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Gain(Has_Bicycle, T)= $\frac{1-1}{1}=0$

For attribute Age,

Info(T_{young})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(T_{middle})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(T_{old})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(Age, T)= $\frac{1}{2}Info(T_{young})+\frac{1}{4}Info(T_{middle})+\frac{1}{4}Info(T_{old})=1$

SplitInfo(Age)= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{4}\log\frac{1}{4}-\frac{1}{4}\log\frac{1}{4}=1.5$

Gain(Age, T)= $\frac{1-1}{1.5}=0$

For attribute Income,

Info(T_{high})= $-1\log 1-0\log 0=0$

Info(T_{fair})= $-\frac{3}{4}\log\frac{3}{4}-\frac{1}{4}\log\frac{1}{4}=0.8113$

Info(T_{low})= $-0\log 0-1\log 1=0$

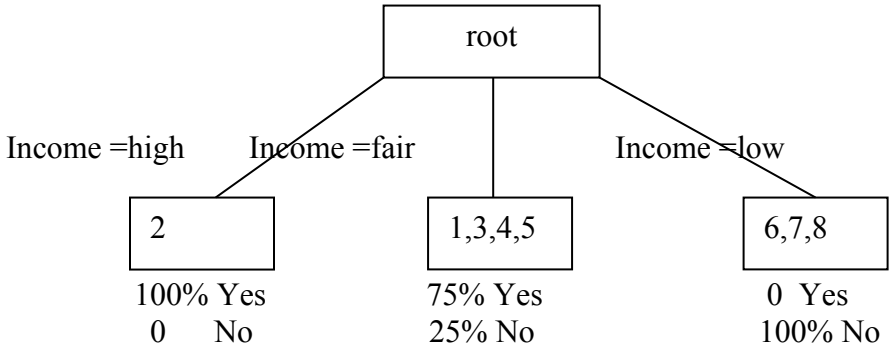
Info(Income)= $\frac{1}{8}Info(T_{high})+\frac{1}{2}Info(T_{fair})+\frac{3}{8}Info(T_{low})=0.405$

SplitInfo(Income)= $-\frac{1}{8}\log\frac{1}{8}-\frac{1}{2}\log\frac{1}{2}-\frac{3}{8}\log\frac{3}{8}=1.4056$

Gain(Income, T)= $\frac{1-0.405}{1.4056}=0.4233$

Q5 (Continued)

We choose attribute Income for Splitting:



Consider the node for "Income =fair"

Info(T)= $-\frac{3}{4}\log\frac{3}{4}-\frac{1}{4}\log\frac{1}{4}=0.8113$

For attribute Has_Bicycle,

Info(T_{no})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(T_{yes})= $-1\log 1-0\log 0=0$

Info(Has_Bicycle, T)= $\frac{1}{2}Info(T_{no})+\frac{1}{2}Info(T_{yes})=0.5$

SplitInfo(Has_Bicycle)= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Gain(Has_Bicycle, T)= $\frac{0.8113-0.5}{1}=0.3113$

For attribute Age,

Info(T_{young})= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{2}\log\frac{1}{2}=1$

Info(T_{middle})= $-1\log 1-0\log 0=0$

Info(T_{old})= $-1\log 1-0\log 0=0$

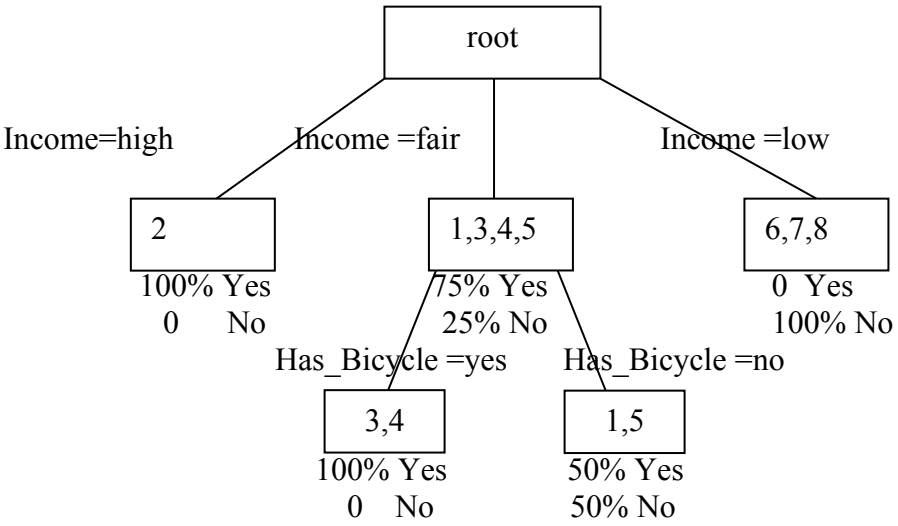
Info(Age)= $\frac{1}{2}Info(T_{young})+\frac{1}{4}Info(T_{middle})+\frac{1}{4}Info(T_{old})=0.5$

SplitInfo(Age)= $-\frac{1}{2}\log\frac{1}{2}-\frac{1}{4}\log\frac{1}{4}-\frac{1}{4}\log\frac{1}{4}=1.5$

Gain(Age, T)= $\frac{0.8113-0.5}{1.5}=0.2075$

Q5 (Continued)

We choose attribute Has_Bicycle for Splitting:



(a) (ii) It is very likely that this user will buy a hoverboard.

Q5 (Continued)
(b)

Differences:
The definition of the gain used in C4.5 is different from that used in ID3.
The gain used in C4.5 is equal to the gain used in ID3 divided by SplitInfo.

The reason why there is a difference is described as follows.
In ID3, there is a higher tendency to choose an attribute containing more values (e.g., attribute identifier and attribute HKID). Thus, splitInfo in C4.5 is used to penalize an attribute containing more values. If this value is larger, the penalty is larger.

Part B (Bonus Question)

Note: The following bonus question is an **OPTIONAL** question. You can decide whether you will answer it or not.

Q6 (10 Additional Marks)

(a)

Yes. The Apriori Algorithm can be adapted.
The idea is similar to the original Apriori Algorithm learnt in class.
However, we use the profit of an itemset in the adapted algorithm instead of the total number of rows containing the itemset in the dataset.
The reason why we can follow the framework if the Apriori Algorithm is that we have he following property.

- if an itemset S has profit >= 50, then any proper subset of S has profit >=50. (Or if an itemset S does not have profit >= 50, then any proper superset of S must not have profit >=50.)

Algorithm:

1. $L_1 \leftarrow$ All items with profit >= 50
2. $k \leftarrow 2$
3. while $L_{k-1} \neq \varnothing$
   - $C_k \leftarrow$ Generate candidates from L_1 by Join Step and Prune Step discussed in class
   - Perform a counting step on C_k and obtain L_k
4. Output $\bigcup_i L_i$

For example,

	frequency	profit
A	0+3+0+1+6=10	10*10=100
B	0+4+0+0+0=4	4*10=40
C	3+0+1+3+0=7	7*10=70
D	2+0+3+5+0=10	10*10=100

So, L_1={A,C,D}, C_2={AC,AD,CD}

	frequency	profit
AC	0+0+0+1+0=1	1*10=10
AD	0+0+0+1+0=1	1*10=10
CD	2+0+1+3+0=6	6*10=60

L_2 = {CD}
Output = {A, C, D, CD}

Q6 (Continued)

(b)

No. The Apriori Algorithm cannot be adapted. In this problem, the following Apriori property cannot be satisfied.

- If an itemset S has profit >=50, then any proper subset of S has profit >=50. (Or if an itemset S does not have profit >=50, then any proper superset of S must not have profit >=50.)

The following shows an example that this property cannot be satisfied.

	frequency	profit
CD	2+0+1+3+0=6	6*(6+4)=60
C	3+0+1+3+0=7	7*6=42

In the above example, CD has profit >=50, but a proper subset of CD (e.g., C) has profit < 42.

Here, we have another algorithm for reference.

1. $O \leftarrow \varnothing$
2. For each possible itemset S with frequency >=1,
   - Find the profit of S
   - If the profit of S >=50,

     $O \leftarrow O \cup \{S\}$
3. Return O.

Q6(Continued)

End of Answer Sheet