Exercise 1

min_sup = 0.3

Items	support	Type
a	0.5	C
b	0.7	C
c	0.5	C
d	0.9	C
e	0.6	F

Items	support	Type
ab	0.3	M
ac	0.2	I
ad	0.4	F
ae	0.4	F
bc	0.3	M
bd	0.6	C
be	0.4	F
cd	0.4	C
ce	0.2	I
de	0.6	C

Items	support	Type
abc		I
abd	0.2	I
abe	0.2	I
acd		I
ace		I
ade	0.4	M
bcd	0.2	I
bce		I
bde	0.4	M
cde		I

Support for abc, acd and ace is not counted since itemset ac is infrequent and therefore any superset cannot be frequent which means that the superset cannot also be closed or maximal. The same counts for ace, bce and cde, because itemset ce was found infrequent.

Items	support	Type
abcd		I
abce		I
abde	0.2	I
acde		I
bcde		I

Support for abcd, abce, acde and bcde is not counted since they have subsets which are infrequent.

Itemsets ab,bc, ade and bde are found maximal since they are frequent but none of their supersets are frequent. Maximal itemsets are also closed!

Itemsets a,b,c,d,ab,bc, bd, cd,,de, ade, bde are closed since none of their immediate supersets have the same support as they have.

Itemset e is frequent but not closed since itemset de has the same support as it. Since the itemset b is not closed it cannot be maximal. The same is true for itemsets ad (s(ad)=s(ade)), ae (s(ae)=s(ade)) and be (s(be)=s(bde)).

The rest of itemsets are infrequent since they haven't got enough big support.

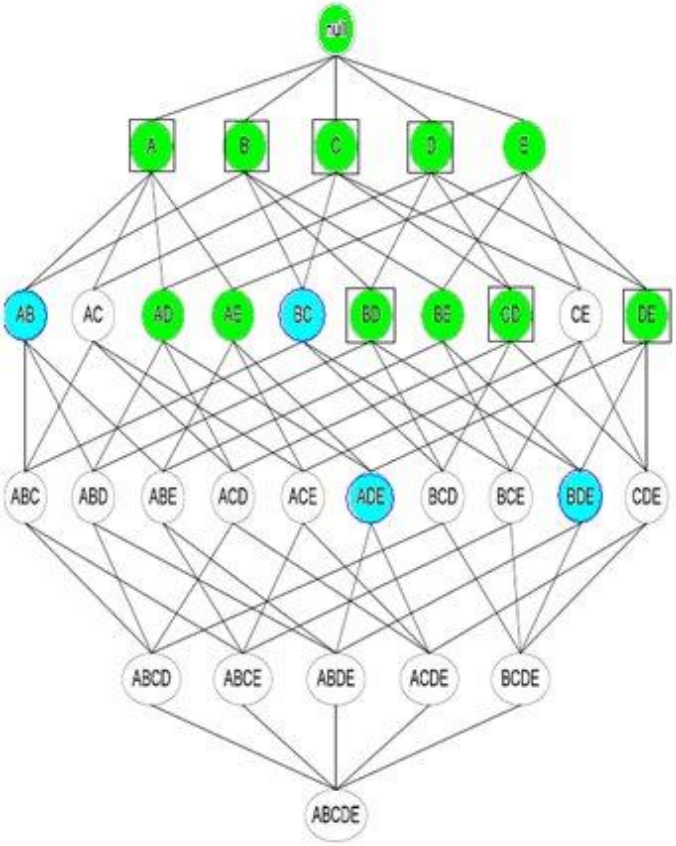


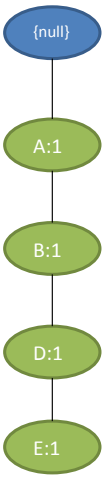
Figure 1. Green nodes are frequent (also null), white infrequent, blue maximal and boxed ones closed itemsets.

Exercise 2

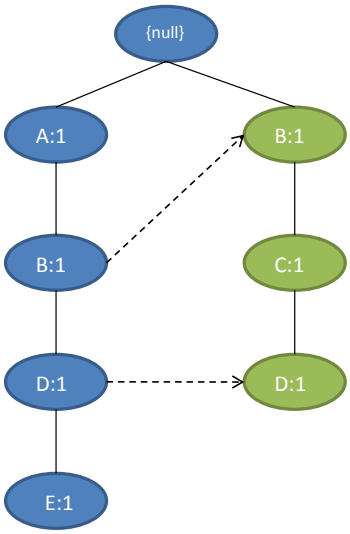
Added nodes are marked as green and updated nodes as orange.

a) alphabetical order

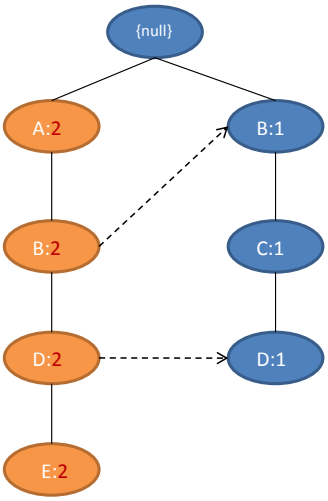
Adding {abde}



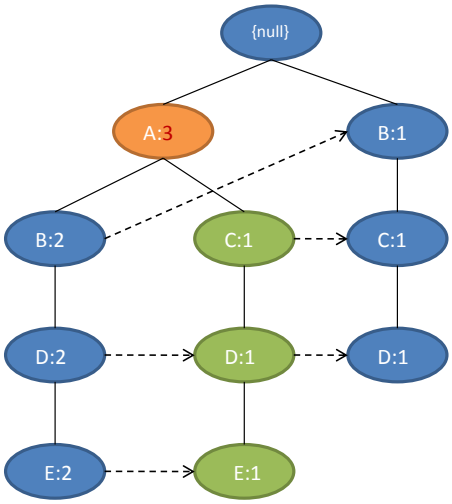
Adding {bcd}



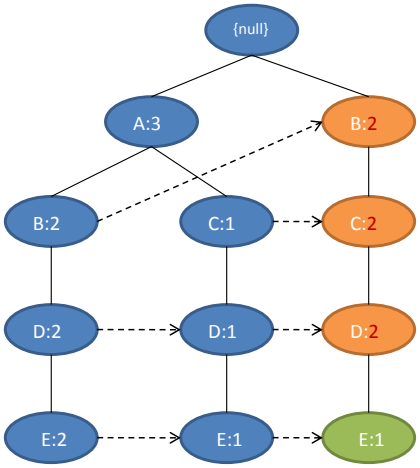
Adding {abde}



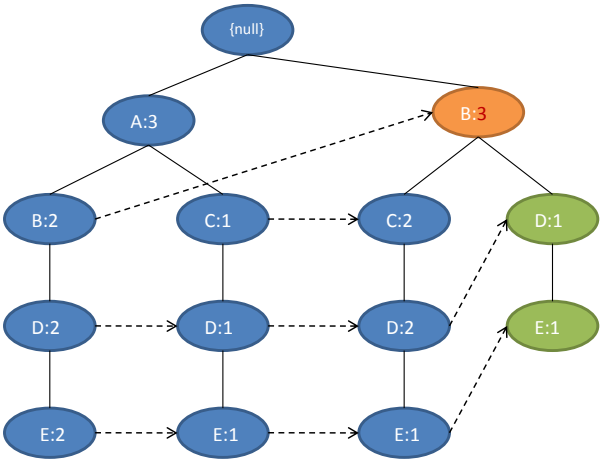
Adding {acde}



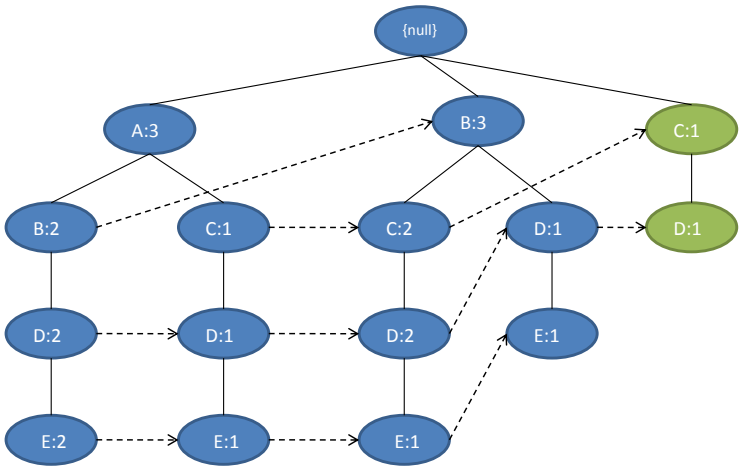
Adding {bcde}



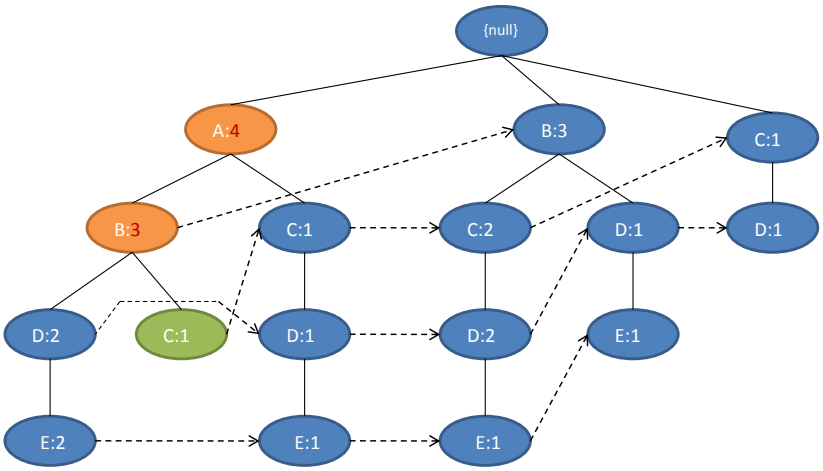
Adding {bde}



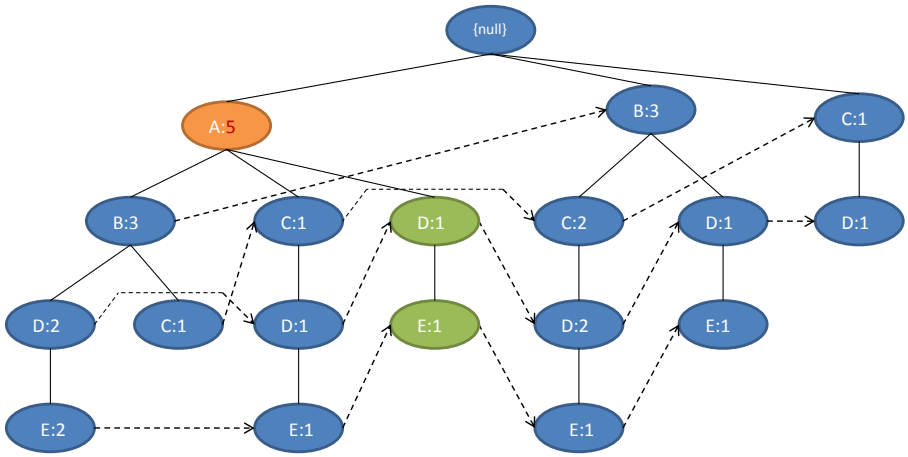
Adding {cd}



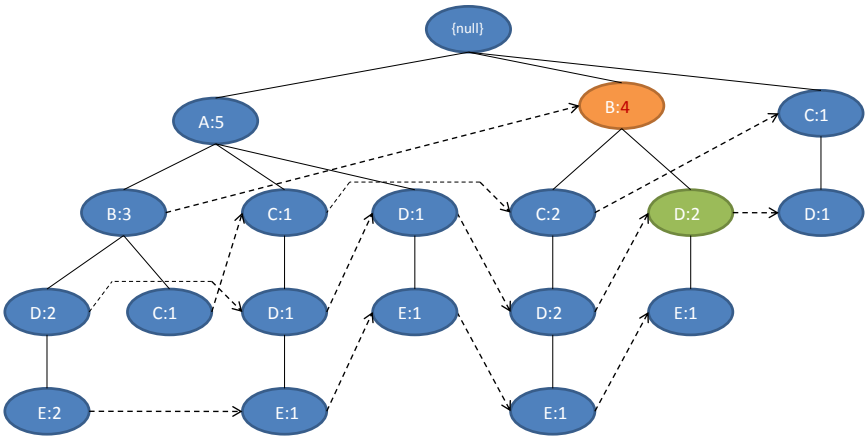
Adding {abc}



Adding {ade}



Adding {bd}

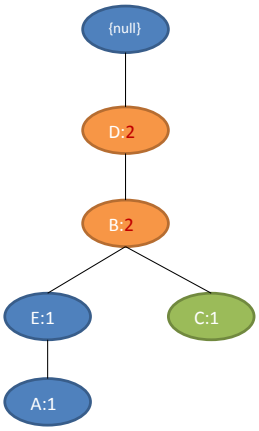


b) descending order of support

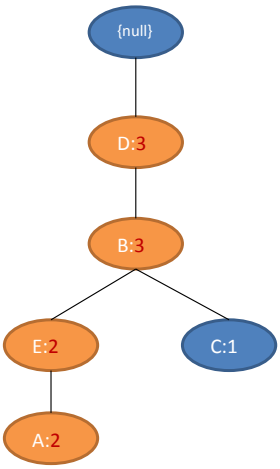
order: D:0.9, B:0.7, E:0.6, A:0.5, C:0.5

Adding {dbea}

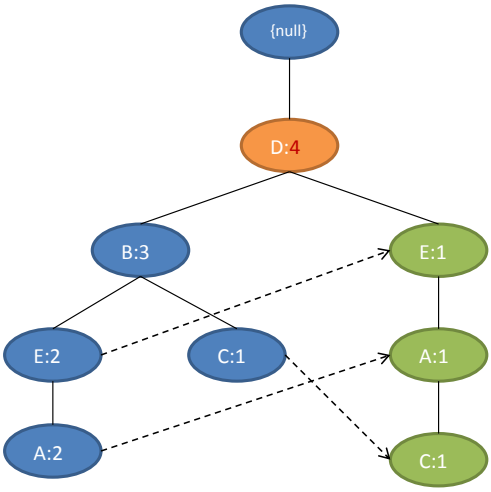
Adding {dbc}



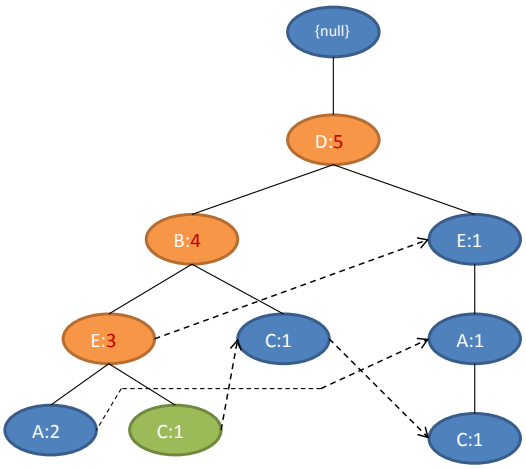
Adding {dbea}



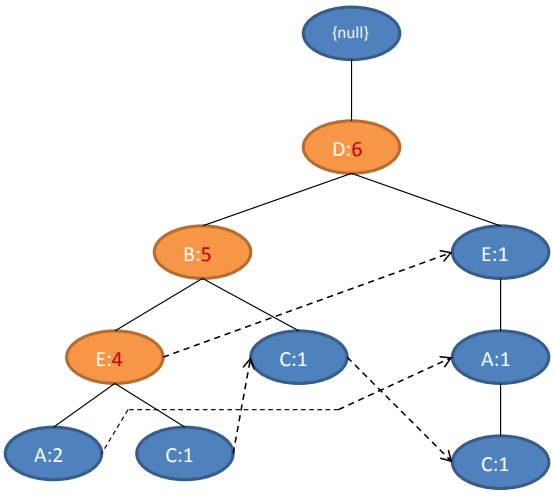
Adding {deac}



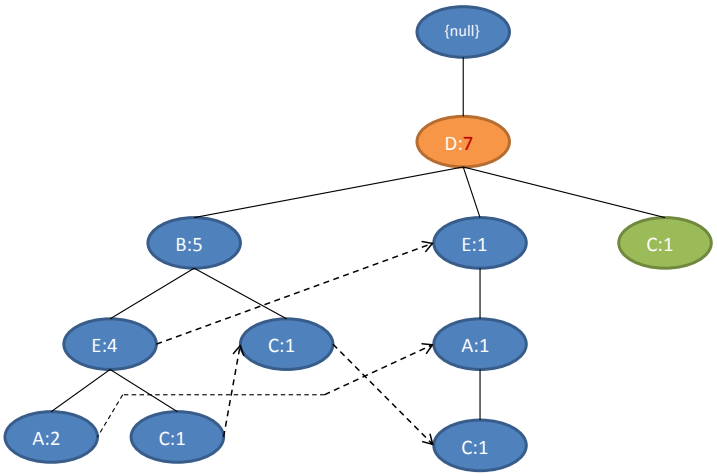
Adding {dbec}



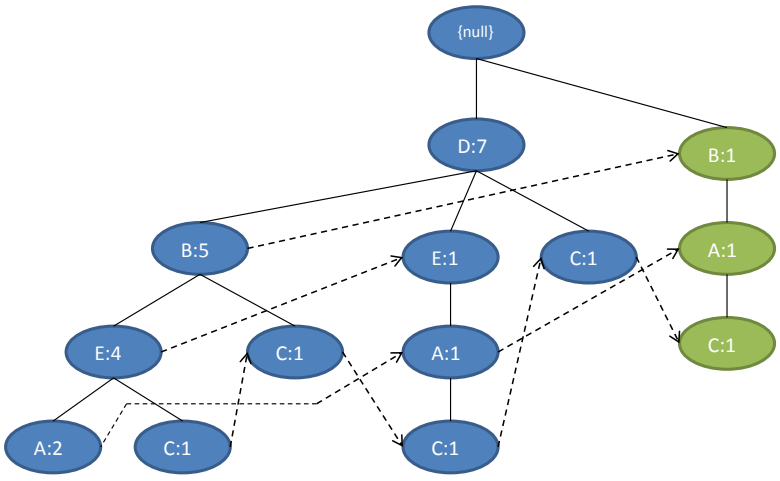
Adding {dbe}



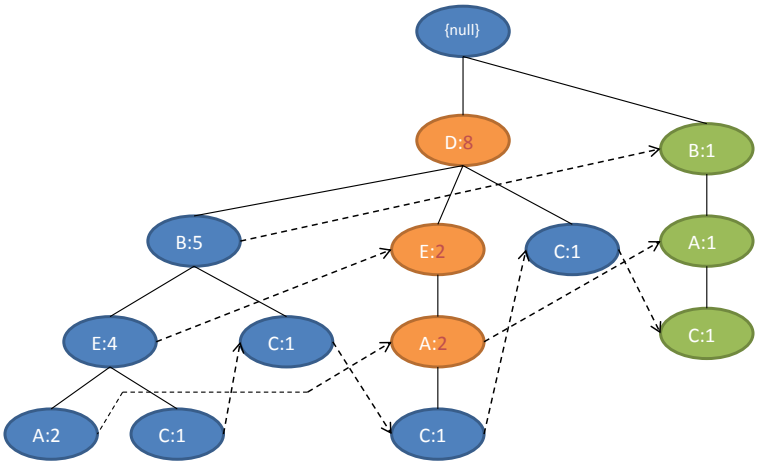
Adding {dc}



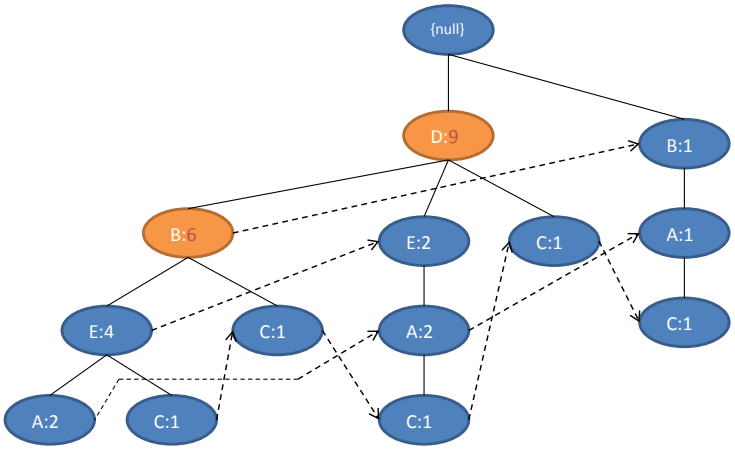
Adding {bac}



Adding {dea}



Adding {db}

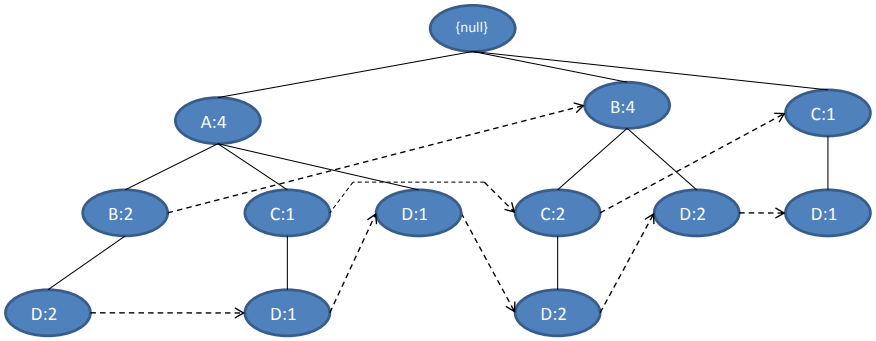


Exercise 3

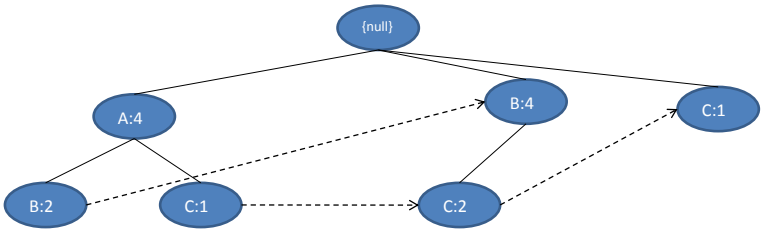
min_sup = 0.3

a) {d}-conditional tree

set of paths ending in d

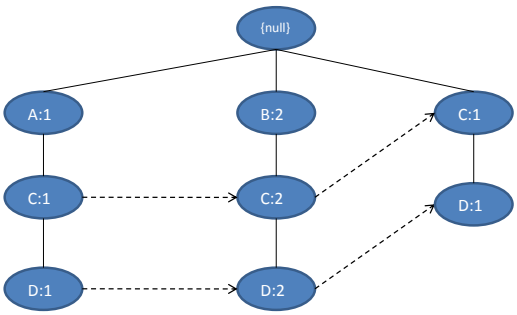


and {d}-conditional tree



b) {c,d}-conditional tree

set of paths ending in cd



and {c,d}-conditional tree

Exercise 4

a) b → c

	c	not c	
b	3	4	7
not b	2	1	3
	5	5	10

The first cell (b and c) has the support of itemset {bc}, cell b not c has count of transactions which include b but not c etc.

b) a →d

	d	not d	
d	4	1	5
not d	5	0	5
	9	1	10

c) b →d

	d	not d	
b	6	1	7
not b	3	0	3
	9	1	10

d) e → c

	c	not c	
e	2	4	6
not e	3	1	4
	5	5	10

e) e → a

	a	not a	
e	4	2	6
not e	1	3	4
	5	5	10

Exercise 5

a) confidence: c(A→B) = s(AB) / s(A)

c(b→c) = s(bc)/s(b) = 3/7 ≈ 0.429
c(a→d) = s(ad)/s(a) = 4/5 = 0.8
c(b→d) = s(bd)/s(b) = 6/7 ≈0.857
c(e→c) = s(ec)/s(e) = 2/6 ≈ 0.333
c(e→a) = s(ea)/s(e) = 4/6 ≈ 0.667
descending order by confidence: b→d, a→d, e→a, b→c, e→c

b) interest factor: I(AB) = s(AB)/(s(A)s(B))
I<1 denotes the pattern occurs less often than expected from independent events

I(b,c) = s(bc)/(s(b)s(c)) = 3/(7*5) ≈ 0.857
I(a,d) = s(ad)/(s(a)s(d)) = 4/(5*9) ≈ 0.889
I(b,d) = s(bd)/(s(b)s(d)) = 6/(7*9) ≈ 0.952
I(e,c) = s(ec)/(s(e)s(c)) = 2/(6*5) ≈ 0.667
I(e,a) = s(ea)/(s(e)s(a)) = 4/(6*5) ≈ 1.333

descending order by interest factor e→a, b→d, a→d, b→c, e→c

c) IS: IS(A,B) = s(AB)/sqrt(s(A)s(B))
IS(b,c) = s(bc)/sqrt(s(b)s(c)) = 3/sqrt(7*5) ≈ 0507
IS(a,d) = s(ad)/sqrt(s(a)s(d)) = 4/sqrt(5*9) ≈ 0.596
IS(b,d) = s(bd)/sqrt(s(b)s(d)) =6/sqrt(7*9) ≈ 0.756
IS(e,c) = s(ec)/sqrt(s(e)s(c)) = 2/sqrt(6*5) ≈ 0.365
IS(e,a) = s(ea)/sqrt(s(e)s(a)) = 4/sqrt(6*5) ≈ 0.730

descending order by IS measure:
b →d, e→a, a→d, b→c, e→c

Exercise 6

a) b→ c

Taking first randomly permuted column (4 6 2 10 3 5 7 8 9 1) and "swap" values of c accordingly.
New database:

tid	items
1	abcde
2	bd
3	abcde
4	ade
5	bde
6	bcde
7	cd
8	abc
9	ade
10	bd

s(b) is the same = 0.7
s(bc) = 0.4

c(b→c) = s(bc)/s(b) = 0.4/0.7 ≈ 0.5714

Taking second randomly permuted column (8 1 10 4 2 9 6 5 3 7) and "swap" values of c accordingly.
New database

tid	items
1	abcde
2	bd
3	abde
4	acde
5	bcde
6	bde
7	d
8	abc
9	ade
10	bcd

s(b) is the same = 0.7
s(bc) = 0.4
c(b→c) = s(bc)/s(b) = 0.4/0.75 ≈ 0.5714

Taking third randomly permuted column (2 6 8 7 10 3 9 4 5 1) and "swap" values of c accordingly.
New database

tid	**items**
1	abcde
2	bd
3	abcde
4	acde
5	bde
6	bde
7	d
8	abc
9	acde
10	bd

s(b) is the same = 0.7
s(bc) = 0.3
c(b→c) = s(bc)/s(b) = 0.3/0.7 ≈ 0.4286

Taking fourth randomly permuted column (5 9 4 2 3 8 7 10 6 1) and "swap" values of c accordingly.
New database

tid	**items**
1	abcde
2	bd
3	abcde
4	acde
5	bde
6	bcde
7	cd
8	ab
9	ade
10	bd

s(b) is the same = 0.7
s(bc) = 0.3
c(b→c) = s(bc)/s(b) = 0.3/0.7 ≈ 0.4286

confidences ordered in descending order
0.5714, 0.5714, 0.4286, 0.4286

confidence from original data
c(b → c) = 0.429

p-value: two of datasets has higher confidence for rule b → c
p = r/N = 2/4 = 0.5

Exercise 7

a) The original confidence from ex 5.a) : c(b → c) ≈ 0.429
p = r/N
Because the original confidence "hits" to the second slot we choose the worst possibility for us, which means that we get the biggest p-value.
Therefore we take the count of three bins having higher or equal confidence and use the sum as r.
r = 42+45+6 = 93
and p = 93/100 = 0.93

b) The original confidence from ex5.a): c(a → d) ≈ 0.8
The original confidence "hits" the first slot. If we count the sum of all bins having equal or higher confidence we get
p = r/N = 100/100 = 1

c) The original confidence from ex5.a): c(e → c) ≈ 0.333
The original confidence "hits" the border of the first and second bin. If we count the three bins having equal or higher confidence , we get
p = r/N = (24+47+26)/100 = 0.97

Based on these the rules are ranked based on p-value (worst case scenario):

b→c, e→c, a→d

Note about p-value: p-value is measuring the statistical significance: what is the propability that our value is obtained randomly.

Basic definition of p-value is that the p-value is the probability that a variable would assume a value greater than or equal to the observed value from random source.
This means that the smaller p-values we get, more sure we can be that our value is not just accidentally "good" and therefore we want to get as small p-values as we can.

To be sure that our evaluation is correct, we here try to give estimation of which we can be sure of and therefore we give the worst case scenario p-value (= every other observed confidence value in the bin our value hits is bigger than our value). This means that at least we're not "lying" or giving too good estimates. In reality it could well be that there's only one value higher than our confidence-value in the bin, but we cannot be sure of that and that's why we assume that all the values in the bin where our value hits are higher than our value.

Exercise 8

tid	a	b	c	d	e
1	1	1	0	1	1
2	0	1	1	1	0
3	1	1	0	1	1
4	1	0	1	1	1
5	0	1	1	1	1
6	0	1	0	1	1
7	0	0	1	1	0
8	1	1	1	0	0
9	1	0	0	1	1
10	0	1	0	1	0

conditions for swapping: xi=yj=1 and xj=yi=0

swapping (row,col), (row,col) = (8,2), (4,5)

tid	a	b	c	d	e
8	1	1	1	0	0
4	1	0	1	1	1

condition fulfilled (8,2)=(4,5) = 1 and (8,5)=(4,2) = 0

tid	a	b	c	d	e
8	1	0	1	0	1
4	1	1	1	1	0

swapping (row,col), (row, col) = (7,2), (1,5)

tid	a	b	c	d	e
7	0	0	1	1	0
1	1	1	0	1	1

condition not fulfilled since (7,2) != (1,5) and (7,2) = 0 and (7,5) != (1,2) and (1,2)=1

swapping (row,col), (row,col) = (2,4), (8,1)

tid	a	b	c	d	e
2	0	1	1	1	0
8	1	0	1	0	1

condition fulfilled (2,4)=(8,1) = 1 and (2,1)=(8,4) = 0 → swapping

tid	a	b	c	d	e
2	1	1	1	0	0
8	0	0	1	1	1

swapping (row,col), (row, col) = (4,2), (3,4)

tid	a	b	c	d	e
4	1	1	1	1	0
3	1	1	0	1	1

condition not fulfilled since (4,4) =(3,2)=1

swapping (row,col), (row,col) = (9,5), (2,2)

tid	a	b	c	d	e
9	1	0	0	1	1
2	1	1	1	1	0

condition fulfilled (9,5)=(2,2) = 1 and (9,2)=(2,5) = 0 → swapping

tid	a	b	c	d	e
9	1	1	0	1	0
2	1	0	1	1	1

new transaction database

tid	a	b	c	d	e
1	1	1	0	1	1
2	1	0	1	1	1
3	1	1	0	1	1
4	1	1	1	1	0
5	0	1	1	1	1
6	0	1	0	1	1
7	0	0	1	1	0
8	0	0	1	1	1
9	1	1	0	1	0
10	0	1	0	1	0

tid	items
1	abde
2	acde
3	abde
4	abcd
5	bcde
6	bde
7	cd
8	cde
9	abd
10	bd