Midterm Solution

1. Fill in your information:

Full Name: ________________________________

NetID: ________________________________

1/1. (10 points) Prove that the co-variance matrix is semi-definite.

Solution. Consult slide 31,32 in lecture 3.

1/1. (5 points) Given the following three methods: *multiway array cubing* (Zhao, et al. SIGMOD,1997),*BUC* (bottom-up computation) (Beyer and Ramakrishnan, SIGMOD,2001),and *shell-fragment approach* (Li et al, VLDB,2004), list one method which is the best and another which is the worst (or not working) to implement one of the following:

A. computing a dense iceberg cube of low dimensionality (e.g., less than 6 dimensions)
Best: Multiway Worst: BUC

B. computing a large iceberg cube of around 8 dimensions with highly skewed data distribution.
Best: BUC Worst: Multiway

C. assisting query answering in high-dimensional cube space.
Best: Shell Fragments Worst: BUC, Multiway

2/1. (5 points) Suppose *MinN* function computes N minimum values in a given set.What kind of measure does *MinN* belongs to distributive, algebraic or holistic? Justify your answer.
Algebraic Function. There are N arguments where each is distributive.

3/1. (5 points) Suppose we use Bottom-Up Computation (BUC) to materialize cubes. We have a 3-D data array containing three dimensions A, B, C. The data contained in the array is as follows:

$(a_0; b_0; c_0) : 2$	$(a_0; b_0; c_1) : 1$	$(a_0; b_0; c_2) : 2$
$(a_0; b_1; c_0) : 2$	$(a_0; b_1; c_1) : 3$	$(a_0; b_1; c_2) : 4$
$(a_0; b_2; c_0) : 2$	$(a_0; b_2; c_1) : 3$	$(a_0; b_2; c_2) : 1$
$(a_0; b_3; c_0) : 2$	$(a_0; b_3; c_1) : 1$	$(a_0; b_3; c_2) : 2$

Now suppose we construct an iceberg cube for dimension A, B, C with different orders of exploration.

A. Draw the trace trees of expansion with regard to exploration order: A, B, C.
Same as quiz

B. Suppose the minimum support = 5 with the exploration order of B, A, C, how many cells would be computed? Please give detailed explanation.
40 cells

2/1. (5 points) Consider a students A. He had taken 2 courses last semester - CS 241 and CS 225. The class size was same for both courses i.e 100. A's total score at the end of semester was 80 in CS 241 while 60 in CS 225. Maximum possible score for both courses was 100

In which class did A do better? Explain.
Here are some statistics on the 2 courses :

Course	Mean Score	Standard Deviation
CS 241	70	10
CS 225	40	10

Solution. $z_{CS241} = \frac{80-70}{10} = 1.0$
$z_{CS225} = \frac{60-40}{10} = 2.0$
So A did better in CS 225.

2. Suppose that a data warehouse contains 20 dimensions, each with about five levels of granularity. Users are mainly interested in four particular dimensions, each having three frequently accessed levels for rolling up and drilling down. How would you design a data cube structure to support this preference efficiently?

Answer:

An efficient data cube structure to support this preference would be to use partial materialization, or selected computation of cuboids. By computing only the proper subset of the whole set of possible cuboids, the total amount of storage space required would be minimized while maintaining a fast response time and avoiding redundant computation.

Rubrics:

If the answer mentions partial materialization or selected computation or similar ideas, give full points.

3/1. (5 points) Stats-R-Us laboratories has conducted a survey to determine how many strawberries are eaten by 100,000 people during a one year period. The data indicate that the number of strawberries eaten is approximately normally distributed with a mean of 29 strawberries, and a standard deviation of 4 strawberries eaten by each person. According to this data, approximately how many of the surveyed people ate more than 25 strawberries during the course of the year?

Solution. 29 -4 = 25 . So 68% + ((100% - 68%)/2) = 84% = 840000

1/1. (5 points) Below are some transaction patterns with their supports from a transaction dataset. Identify all the core patterns ($\boldsymbol{\tau} = \mathbf{0.4}$) for each given transaction pattern. Please show your calculation steps briefly.

Transaction pattern	Support (number of transactions)
ab	100
abc	70
abe	200
bcde	80

Solution.

Transaction pattern	Support (number of transactions)
ab	
abc	c, ac, bc
abe	e, ae, be
bcde	c, d, bc, bd, cd, ce, de, bcd, bce, bde, cde

[9] Suppose the base cuboid of a data cube contains only two cells

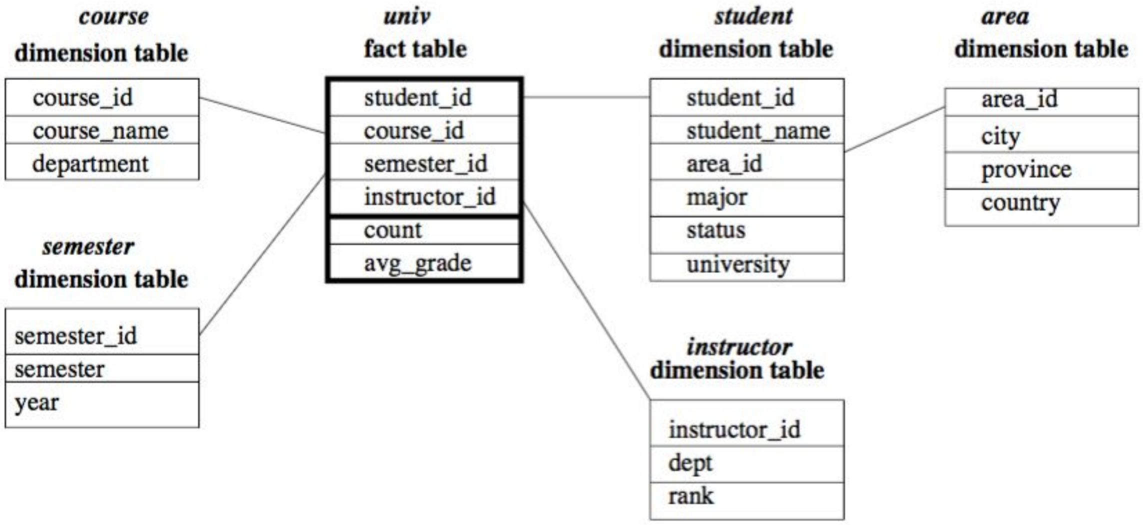
$$(a_1, a_2, a_3, \ldots, a_{20}), (b_1, b_2, b_3, \ldots, b_{20}),$$

where $a_i = b_i$ if i is an odd number; otherwise $a_i \neq b_i$.

i. How many nonempty cuboids are there in this data cube?
Answer: $2^{20} = 1,048,576$

ii. How many nonempty aggregate cells are there in this data cube?
Answer: Each cell generates $2^{20} - 1$ nonempty aggregate cells. Thus in total we should have $2 \times 2^{20} - 2$ with overlapped cells not considering. We have 2^{10} overlapped cells (cells with a_i, or b_i, fixed where i is odd number). Therefore, we have $2^{20} - 2 - 2^{10} = 2,096,126$.

1. Suppose that a data warehouse for a university consists of the following four dimensions: student, course, semester, and instructor, and two measures count and avg grade. When at the lowest conceptual level (e.g., for a given student, course, semester, and instructor combination), the avg grade measure stores the actual course grade of the student. At higher conceptual levels, avg grade stores the average grade for the given combination.



Starting with the base cuboid [student, course, semester, instructor], what specific OLAP operations (e.g., roll-up from semester to year) should one perform in order to list the average grade of CS courses for each university student.

Answer:

The specific OLAP operations to be performed are:
(i) Roll-up on course from course id to department.
(ii) Roll-up on semester from semester id to all.
(iii) Slice for course=“CS”

Rubrics:

2 points for (i)
2 points for (ii)
1 point for (iii)

2/1. (5 points) Below is a table of different pattern constraints. S is a variable set and v is a constant. Please answer whether the constrain is antimonotonic, monotonic or succinct and whether it is convertible?

Constrain	Antimonotonic	Monotonic	Succinct
$sum(S) \leq v$			
$range(S) \geq v$			
$min(S) \leq v$			
$average(S) \geq v$			

Solution.

Constrain	Antimonotonic	Monotonic	Succinct
$sum(S) \leq v$	yes	no	no
$range(S) \geq v$	no	yes	no
$min(S) \leq v$	no	yes	yes
$average(S) \geq v$	convertible	convertible	no

3/1. (10 points) Below is a sequence dataset containing transaction sequences. Note that (bc) means items b and c are purchased at the same time. Let **min_sup = 2**,

SID	Sequence
S1	abcdef
S2	a(bc)bd
S3	(bc)(bd)
S4	c(de)
S5	a(bd)e

(a) Use the Generalized Sequential Patterns (GSP) mining algorithm to find all the frequent sequential patterns. You need to list all the steps and results from each step.

(b) Use SPADE to find all the frequent sequential patterns. You need to list all the steps and results from each step.

Solution.

(a) GSP:

C1	sup
$<a>$	3
$<b>$	4
$<c>$	4
$<d>$	5
$<e>$	3
$<f>$	1

L1	sup
$<a>$	3
$<b>$	4
$<c>$	4
$<d>$	5
$<e>$	3

C1	sup
$<ab>$	3
$<ac>$	2
$<ad>$	3
$<ae>$	2
$<bc>$	1
$<bd>$	3
$<be>$	2
$<cd>$	4
$<ce>$	2
$<de>$	2
$<(ab)>$	0
$<(ac)>$	0
$<(ad)>$	0
$<(ae)>$	0
$<(bc)>$	2
$<(bd)>$	2
$<(be)>$	0
$<(cd)>$	0
$<(ce)>$	0
$<(de)>$	1

L2	sup
$<ab>$	3
$<ac>$	2
$<ad>$	3
$<ae>$	2
$<bd>$	3
$<be>$	2
$<cd>$	4
$<ce>$	2
$<de>$	2
$<(bc)>$	2
$<(bd)>$	2

C3	sup
$< abc >$	1
$< abd >$	2
$< abe >$	2
$< acd >$	2
$< ace >$	1
$< ade >$	2
$< bcd >$	1
$< bce >$	1
$< bde >$	1
$< cde >$	1
$< a(bc) >$	1
$< b(bc) >$	0
$< c(bc) >$	0
$< d(bc) >$	0
$< e(bc) >$	0
$< a(bd) >$	1
$< b(bd) >$	1
$< c(bd) >$	1
$< d(bd) >$	0
$< e(bd) >$	0
$< (bc)a >$	0
$< (bc)b >$	2
$< (bc)c >$	0
$< (bc)d >$	2
$< (bc)e >$	0
$< (bd)a >$	0
$< (bd)b >$	0
$< (bd)c >$	0
$< (bd)d >$	0
$< (bd)e >$	1

L3	sup
$< abd >$	2
$< abe >$	2
$< acd >$	2
$< ade >$	2
$< (bc)b >$	2
$< (bc)d >$	2

C4	sup
$< abde >$	1
$< abcd >$	1
$< acde >$	1
$< (bc)bd >$	1
$< (bc)be >$	0
$< (bc)de >$	0
$< (bc)(bd) >$	1

L4	sup

(b) SPADE:

SID	EID	Item
1	1	a
1	2	b
1	3	c
1	4	d
1	5	e
1	6	f
2	1	a
2	2	bc
2	3	b
2	4	d
3	1	bc
3	2	bd
4	1	c
4	2	de
5	1	a
5	2	bd
5	3	e

a

SID	EID
1	1
2	1
5	1

b

SID	EID
1	2
2	2
2	3
3	1
3	2
5	2

c

SID	EID
1	3
2	2
4	1

d

SID	EID
1	4
2	4
3	2
4	2
5	2

e

SID	EID
1	5
4	2
5	3

ab

SID	EID (a)	EID (b)
1	1	2
2	1	2
5	1	2

ac

SID	EID (a)	EID (c)
1	1	3
2	1	2

ad

SID	EID (a)	EID (b)
1	1	4
2	1	4
5	1	2

ae

SID	EID (a)	EID (e)
1	1	5
5	1	3

bd

SID	EID (b)	EID (d)
1	2	4
2	2	4
2	3	4
3	1	2

be

SID	EID (b)	EID (e)
1	2	5
5	2	3

cd

SID	EID (c)	EID (d)
1	3	4
2	2	4

ce

SID	EID (c)	EID (e)
1	3	5
4	1	2

de

SID	EID (d)	EID (e)
1	4	5
5	2	3

(bc)

SID	EID
2	2
3	1

(bd)

SID	EID
3	2
5	2

abd

SID	EID (a)	EID (b)	EID (d)
1	1	2	4
2	1	2	4

abe

SID	EID (a)	EID (b)	EID (e)
1	1	2	5
5	1	2	3

acd

SID	EID (a)	EID (c)	EID (d)
1	1	3	4
2	1	2	4

ade

SID	EID (a)	EID (d)	EID (e)
1	1	4	5
5	1	2	3

$(bc)b$

SID	EID (bc)	EID (b)
2	2	3
3	1	2

$(bc)d$

SID	EID (bc)	EID (b)
2	2	4
3	1	2
