אוניברסיטת אריאל בשומרון
המחלקה למדעי המחשב

מבחן במסדי נתונים

מירב שקרון

7028010

סמסטר קיץ מועד א' כט תשרי התשע"ח, 19.10.2017

הנחיות כלליות:

- משך הבחינה: 180 דקות.
- **יש לענות בגוף השאלון!** המחברת תשמש כטיוטא בלבד. על מענה במחברת יורדו נקודות!
- במבחן 11 שאלות, שימו לב שאתם עוברים על כולם
- אין להכניס שום חומר עזר.
- השימוש במחשבון **אסור**.
- בסיום הבחינה - נא למסור את השאלון ואת המחברת.

	1	2	3	4	5	6	7	8	9	10	11	Total
Max points	16	5	13	8	5	7	10	5	9	9	13	100
Grade												

ב ה צ ל ח ה!

03.9066640
03.9066692
אריאל, 40700
אוניברסיטת אריאל בשומרון ע"ר

1. SQL (16 נק')
נתון מבנה הנתונים הנ"ל, של אולימפיאדת מקסיקו 2016:

Participant (ParticipantId, name, gender, yearOfBirth, country) טבלת משתתפים

CompetitionTypes (competitionTypesId, name) טבלת תחרויות

Prizes (prizeId, name, points) טבלת סוגי פרסים וניקוד

CompetitionParticipant (competitionTypeId, ParticipantId, dayAsInt, prizeId)
טבלת משתתפים בתחרות והפרס (הפרס יכול להיות null)

ניתן להניח ש dayAsInt הוא מספר שלם המייצג את יום המשחק מאז שהחלה האולימפידה.

א. כיתבו שאילתא שמחזירה את מס' הנשים שזכו בפרסים ואת מס' הגברים שזכו בפרסים ומה סה"כ הניקוד של כל אחד מהם (של הגברים והנשים בנפרד)

```
SELECT gender, count (*) as price, sum (points) as points, sum (points) as points

FROM Participant p JOIN CompetitionParticipant cp

ON p.ParticipantId = cp.ParticipantId JOIN Prizes pr ON cp.prizeId = pr.prizeId

GROUP BY gender
```

ב. כיתבו שאילתא שמחזירה עבור כל יום את **גילו** של הצעיר ביותר שלא זכה באף תחרות

```
SELECT dayAsInt, MIN(2016 – yearOfBirth) as younger_looser,

FROM Participant p JOIN CompetitionParticipant cp

ON p.ParticipantId = cp.ParticipantId

WHERE genre = 1 AND prizeId = NULL

GROUP BY dayAsInt
```

2. Relational algebra (5 נק')

כתבו שאילתת SQL השקולה לביטוי הרלציוני הבא:

$$\pi_{(name,staffName,salary)}(\sigma_{salary\geq 2000})(employee \bowtie_{id=employeeId} staff))$$

```
SELECT name, staffName, salary
FROM employee e RIGHT JOIN staff s
ON e.id = s.employeeId
WHERE salary >= 2000
```

3. Normalization (13 נק')

נתונה הרלציה הבאה:

R (U, V, W, X, Y, Z)

נתונות התלויות שלה:

Z→W

Y→{X, Z}

{W, X}→Y

{U, Y, Z}→V

א. מצא את כל ה-candidate keys האפשריים (5 נק')

{U, Y}, {U, W, X} , {U, Z, X} are the two candidate-keys.

The "algorithm" to find them is the following:

First verify which attribute(s) can't be determined by other attribute(s): In our case U.

Second, verify which attribute(s) can we determined with the previous attribute(s) (U): In our case no one.

Finally verify all the possibilities and check if the possibilities are candidate-key or not..

Beginning by the easier, {U, Y}. Then, since {w, x}→ Y, we also have {U, W, X}. Also, Z→W, so we have {U, Z, X}.

ב. מהם תכונות ה- prime? (1 נק')

The prime attributes are U, W, X, Y, Z. Only V is a non prime since he doesn't appear in any candidate-key.

ג. מהי רמת הנירמול (NF) של הרלציה הנ"ל (1NF, 2NF, 3NF, BCNF, 4NF).
נמק למה הרלציה לא יכולה להתאים ל-NF גבוה יותר (7 נק')

Assuming 1NF, we now check the 2NF:

Since V is the only non-prime attribute, we only need to worried about him.

The only subset on which V depends is {U, Y, Z} which is not a PROPER SUBSET of any candidate-key. Then, the NF level is at least 2.

we're now checking 3NF:

As above, we're only checking for V. As we know, V depends on {U, Y, Z} which is actually a super-key (since {U, Y} is a candidate-key). Then, by definition the NF level is at least 3.

About 3.5NF, you can notice that W depends on Z, which isn't a key. We are done and the level is finally 3NF.

XML and XSD.4 (8 נק')

כתבו את **כל** הסיבות מדוע ה- XML הבא, לא בתוקף (not valid) לפי הגדרת ה-XSD להלן:

XML:

```
<pencilCase>
     <pencil>HB</pencil>
     <pen>BLUE</pen>
     <scissors>AAA</scissors>
     <eraser>STAEDTLER</eraser>
     <ruler>15 cm</ruler>
       <sharpener>KANEX</sharpener>
</pencilCase>
```

XSD:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
   <xs:element name="pencilCase">
   <xs:complexType>
     <xs:sequence>
       <xs:element name="pencil" minOccurs="2" maxOccurs="4"/>
       <xs:element name="pen" type="xs:int" />
       <xs:element name="eraser" />
       <xs:element name="scissors" />
       <xs:element name="ruler" />
       <xs:element name="colors" maxOccurs="10"/>
       <xs:element name="sharpener" />
     </xs:sequence>
     <xs:attribute name="forWhom" default="students"/>
   </xs:complexType>
 </xs:element>
</xs:schema>
```

1. pencil occurs only once but must occurs at least twice.
2. pen must be a number but it's a string "BLUE".
3. eraser and scissors must swap.
4. color must occur at least once since the "minOccurs" of the circle is not define, but he is not occurs.
5. pencilCase must have an attribute name of the name "forWhom" with a default: "students" but he hasn't.

NoSQL.5 (5 נק')

מה ההבדל העיקרי בין בסיס נתונים של Redis לבסיס נתונים של MongoDB?

א. Redis מאחסן נתונים בצורה Key-Value ו-MongoDB מבוסס על Wide column store.

ב. ב- Redis ניתן לאחסן רק ערכים פשוטים כמו string, int וכו' בעוד שב-MongoDB ניתן לשמור גם ערכים בצורה של xml או json

ג. ב- Redis ניתן לשלוף נתונים לפי ה-key שלהם בעוד שב-mongoDB ניתן לתשאל גם את הנתונים עצמם.

ד. MongoDB הרבה יותר מהיר מ-Redis.

Explanation:

From the sildes: about mongoDB: db.students.find({"FirstName": "Tal"}).

Then, as we can see, in mongoDB it's possible to ask for the own data. In contrary in Redis, it's not possible.

6. TF-IDF (7 נק')

דרגו את המשפטים הבאים לפי הניקוד של ה- TF-IDF שלהם בהינתן ה-query הבא. במקום הראשון- את התשובה הרלוונטית ביותר ובמקום האחרון את התשובה הכי פחות רלוונטית. נמק את תשובתך.

(אסור להשתמש במחשבון וגם אין צורך בו)

להזכירכם נוסחת ה- TF-IFD היא:

$$tfidf(d) = \sum_{k=0}^{|Q|} \frac{\#k\ in\ d}{|d|} \log\left(\frac{|D|}{\#D\ with\ k}\right)$$

Q: horses and cows

D1: **horses** gallop while **cows** moo
D2: zebras **and horses** are similar
D3: Milk all **cows and** goats
D4: Jessica **and** Daniel are a couple, married last summer
D5: Ariel **and** Dvora love to ride **horses**

First (**most** relevant): D1
Second: D3
Third: D2
Forth: D5
Fifth (**least** relevant): D4

Explanation:

You can provide two ways for the answer.

First way:

By |sentence| we mean the number of word in the sentence.By |word| we mean the number of time the the word (in the question Q) appear in the sentences

(D1, ... ,D5).

By D1 → word(s) we mean that in D1 there are the word(s).

We next count the data for each sentence:

|D1| = 5 |D2| = 5 |D3| = 5 |D4| = 9 |D5| = 7

|horses| = 3 |and| = 4 |cows| = 2

(about the punctuation, I have no idea how this is work, but I think there is nothing to worried

about cause the test question will not ask about).

Now sort the results as : (ascendant sorting)

|D1| = |D2| = |D3| < |D5| < |D2|

|cows| < |horses| < |and|

Then we already know which sentence is the forth (D5) and last (D2).

We need to class D1, D3 AND D4.

D1→ |horses| + |cows| = 5

D2 → |and| + |horses| = 7

D3 → |cows| + |and| = 6

sort the result as : (ascendant sorting)

D1 < D3 < D2

We're done.

For the second solution, please use the formula.

Cassandra.7 (10 נק')

נתונות 2 הטבלאות הבאות- T1, T2.

T1 מוגדרת כך:

CREATE TABLE T1 (W INT, X INT, Y INT, Z INT, PRIMARY KEY ((W, X), Y, Z));

T2 מוגדרת כך:

CREATE TABLE T2 (W INT, X INT, Y INT, Z INT, PRIMARY KEY (W, X, Y));

סמן עבור כל אחד מהשאילתות ה-CQL האם היא legal, illegal או syntax error

a. SELECT * FROM T1 WHERE Y<412 AND W=152 AND X=741 — legal
b. SELECT X FROM T1 WHERE W=174 AND X<890 — illegal
c. SELECT X FROM T2 WHERE W=174 AND X<890 — legal
d. SELECT W FROM T2 JOIN T1 ON T2. Z=T1.Z WHERE W=154 — syntax
e. SELECT * FROM T2 WHERE W>59 — illegal

Explanation:

a) Since both of attribute in the partition key are mentioned with an equal statement, adding the FIRST clustering key (it's possible either an = or an <,> statement) the query is legal.

b) In T1, the partition key are W and X and THEY MUST APPEAR in each query, with an equal "=" statement. Trivially, in the query the X key is not mentioned.

c) Since attribute in the partition key is mentioned with an equal statement, adding the FIRST clustering key (it's possible either an = or an <,> statement) the query is legal.

d) The is no JOIN in CASSANDRA as well as UNION.

e) In T2, the partition key id W and IT MUST APPEAR in each query, with an equal "=" statement. Trivially, in the query the W key is not mentioned with an "=".

8.Logistic regression (5 נק')

איזה מהבעיות הבאות מתאימה ביותר לפתור בעזרת Logistic regression (לעומת Linear regression ו-Naïve bayes)

א. ניבוי ציוניו של סטודנט לתואר שני על סמך נוכחותו בשיעורים והגשת מטלות בית
ב. ניבוי הרווחים של חנות כלי בית על סמך מיקום החנות וגודל האוכלוסייה
ג. ניבוי מיקום חנות כלי בית על סמך רווחי החנות וגודל האוכלוסייה
ד. ניבוי האם חנות כלי בית תהיה רווחית או הפסדית על סמך מיקום החנות וגודל האוכלוסייה

Explanation:

The logistic regression is used for binary choice: either 0 or 1.

The only answer which is a binary choice is the d: either the student will win or loose.

9.Mongo DB (9 נק')

בהינתן רשימת פרטי תלמידים הבאה:

```
db.student.insert([
{student_id:"111", name:"Dani", course_name:"Databases", grade:42},
{student_id:"222", name:"Dafna", course_name:"Operating systems", grade:81},
{student_id:"333", name:"Miri", course_name:"Software structure", grade:78},
{student_id:"444", name:"Nati", course_name:"Databases", grade:52},
{student_id:"555", name:"Yaffa", course_name:"Software structure", grade:62},
{student_id:"666", name:"Zohar", course_name:"Operating systems", grade:95}
{student_id:"777", name:"Ari", course_name:"Operating systems", grade:48}
{student_id:"888",name:"Miki",course_name:"Databases",grade:65})
 ] )
```

ופונקציית ה-mapReduce הבאה:

```
db.student.mapReduce(
        function (){
                if ( this.grade > 59 ) {
                        emit(this.course_name, this.grade);
                }
        },
        function(key, val1) {
                var total =0;
                for (var i = 0; i < val1.length; i++) {
                        total += val1[i];
                }
                return total / val1.length;
        },
        {out: "my_out"});
```

מה תכיל my_out לאחר הרצת הקוד לעיל (כלומר, מה נקבל כשנקליד **db.my_out.find()**):

"Algorithm" for the answer:

First from all the item inserted, delete all the item in which the grade is < 59. (in red).

Then, since the key is course, reunite the item with the same course.

Since the value is the grade, and the function is calculate the average, do the same:

Operating systems → 88

Software structure → 70

Databases → 65

The last step is to properly write the answer:

```
“my_out” = {
{ _id:”Operating systems “ , value:88},
{ _id:”Software structure “ , value:70},
{ _id:”Databases“ , value:65},
}
```

01. Linear regression + JavaStreams (9 נק')

יש לממש את פונקציית ה-loss שלמדנו ב-Linear regression $J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(h(x_i) - y_i)^2$

בעזרת JavaStreams.

לצורך כך, הוגדר מבנה הנתונים הבא:

```
class Data {
   double x;
   double y;

   Data(double x, double y){
      this.x = x;
      this.y = y;
   }
}
```

הניחו כי w=0.1 ו-b=5

Assuming we initialize a list as follow:

```
List<Data> myList = Arrays.asList{/**...some data object...**/};
```

We can now use a stream:

```
myList.stream().map(s -> Math.pow(0.1*s.x + 5 - s.y, 2)).reduce((x, y) -> x + y) / 2 *
myList.size();
```

11. Linear regression (13 נק')

נזכיר שעבור linear regression הגדרנו את פונקציית ה-loss הבאה:

$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(h(x_i) - y_i)^2$$

נזכיר שקו הניבוי ב- linear regression הוא: h(x) = xw + b

בהינתן הנתונים הבאים:

X = [2, 4, 0, -2]

Y = [3, 1, 1, -3]

יש להניח שאנו משתמשים ב- Gradient Descent

א. מה ערכי ה- **gradient** (לפי w ולפי b) כאשר w=0 ו-b=0 ? (יש לתת ערך מספרי) (6 נק')

Derived by W :

sum (from I = 1 to m) [wxi + b – yi)*xi / m

Derived by B:

sum (from I = 1 to m) [wxi + b – yi) / m

By setting the value we get:

for W : (- 6 - 4 - 6) / 4 = -4

for B : (-3 -1 -1 +3) / 4 = - 1 / 2

Then, as a final answer: the gradient for W is -4 and for B is -0.5

ב. הניחו ש- α=0.1 , מה יהיו הערכים של w ו-b באיטרציה הבאה?
(השתמשו בגרדיאנט שחישבתם בסעיף א') (3 נק')

By the formulas :

W = w - α*grad = 0 – 0.1 * (-4) = 2 / 5

B = b- α*grad = 0 – 0.1 (- 1 *2) = 1*/20

where W and B are the new values and w and b are the old values (0 and 0).

ג. לפי הערכים שמצאתם ל-w ו-b בסעיף הקודם, מה יהיה הניבוי ל-x=2?
מתי הניבוי יותר מדויק- עם w=0, b=0 או עם הערכים שמצאתם בסעיף הקודם?
(יש להראות חישוב) (4 נק')

By the new W and B's value, we have: h(2) = 2 * (2 / 5) + (1 / 20) = 17 / 20

With the old value we have : h(2) = 0.

When x = 2, we have y = 3 (by the index). Then, trivially, 17/20 is a better prediction than 0 since 3 – 0.85 < 3 – 0.