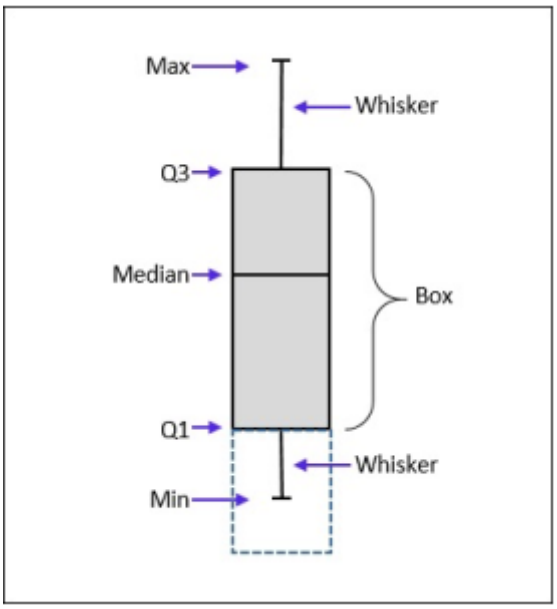
CS412 Midterm Exam

1.

    1. The five-number summary of a distribution consists of the smallest observation, the first quartile, the median, the third quartile, and the largest observation, written in order from smallest to largest.
    2. Boxplot is a graphic display of five-number summary. Data is represented with a box



        - Q1, Q3, IQR: The ends of the box are at the first and third quartiles, i.e., the height of the box is IQR (the middle 50% of values when ordered from lowest to highest).
        - Median (Q2) is marked by a line within the box.
        - Whiskers: two lines outside the box extended to Minimum and Maximum as shown in the picture "Min" and "Max".

          Q1, Q3, Median, Min and Max explained above are the five-number summary incorporated by boxplots.

    3. Yes, two different distributions can have the exact same boxplot, but this doesn't mean the two distribution are of the same shape. Five-number summary is identical doesn't mean that the distribution is identical. Some information is lost when we present data graphically in a box plot.

       This is because five number summary doesn't tell us about the distribution of the values between the minimum and lower quartile, or between the lower quartile and the median, and so on. We do know that the frequency between minimum and lower quartile must match the frequency between lower quartile and median, but we don't know to which values of the variable those frequencies are allocated. As long as two distribution have the same five number summary, their box plots are identical. But that can't guarantee the two distribution have the same shape.

For example,

distribution x contain data {0,0,1,1,2,2,3,3,4,4,5,5}

min_x = 0

max_x = 5

Q1_x = 1

Q3_x = 4

Median_x = 2.5

distribution y contain data {0, 0.75, 0.75, 1.25, 2.5, 2.5, 2.5, 2.5, 3.5, 4.5, 4.75, 5}

min_y = 0

max_y = 5

Q1 = 1

Q3 = 4

Median = 2.5

Distribution x and y apparently are not identical, but since they have the same five-number summary, they share same boxplots.

4. Quantile-Quantile (Q-Q) Plot is a scatterplot, and it graphs the quantiles of one univariate distribution against the corresponding quantiles of another. It is a graphical tool to help us assess data distribution.
5.

   i. Yes, it's possible. It indicates the prices of Store B tends to be higher than the prices of Store A. For example, consider prices of Store A following distribution X_A ~ P while prices of Store B following X_B = min(X_A + 1, a). In this example the QQ plot will be above y = x line.

   ii. It's wrong. (m_A, m_B) lies below y=x only means the median m_A > m_B, while average may not.

   Consider a counter example, if prices of Store-A is [0, 0.75a, 0.75a] and prices of Store-B is [0 , 0.6a , a]. Then m_A = 0.75a > m_B = 0.5a thus (m_A, m_B) is under line y = x. But the average of A is 0.5a, which is less than average of B(8/15 a).

2.

   1. 2.

   Contingency table showing the observed values and calculated row totals, column totals, and grand total:

	Buy Diaper	**Not Buy Diaper**	**row totals**
Buy Beer	100	400	500
Not Buy Beer	300	200	500
column totals	400	600	1000

   Contingency table showing the equations for calculating expected values. These equations represent the mathematical calculations to be performed:

	Buy Diaper	Not Buy Diaper	row number
Buy Beer	(R1*C1)/GT	(R1*C2)/GT	R1
Not Buy Beer	(R2*C1)/GT	(R2*C2)/GT	R2
column number	C1	C2	GT

Contingency table showing the values entered into the equations for calculating expected values:

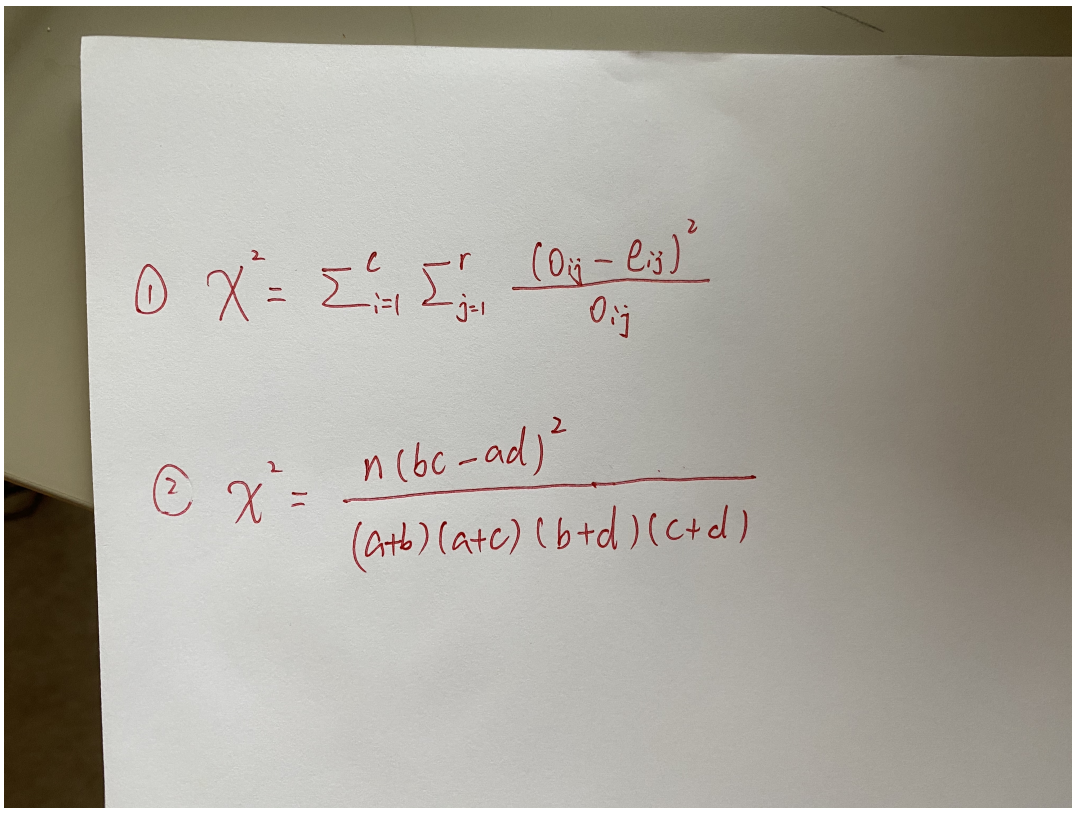
	Buy Diaper	Not Buy Diaper	row total
Buy Beer	(500*400)/1000	(500*600)/1000	500
Not Buy Beer	(500*400)/1000	(500*600)/1000	500
column number	400	600	1000

Contingency table showing the calculated expected values:

	Buy Diaper	Not Buy Diaper
Buy Beer	200	300
Not Buy Beer	200	300

Under the null hypothesis, i.e., 'Buy Beer' and 'Buy Diaper' are independent, the expected number for 'Buy Beer' and 'Buy Diaper' is 200, and the expected number for 'Buy Beer' and 'Not Buy Diaper' is 300.

3.



We can get χ2 either use equation 1 or equation 2.

In equation 1, oij represents the actual value, eij represents the expected value. In this case. the (actual, expected) pairs should be (100, 200), (400, 300), (300, 200), (200, 300). The χ2 statistic for the contingency table is thus 166.666666667.

In equation 2, n = 1000, a = 100, b = 400, c = 300, d = 200. We get the χ2 statistic 166.666666667.

4.

   The value needed to reject null hypothesis at a significance level of α = 0.05 is 3.841. This can be computed using the appropriate function of Scipy.

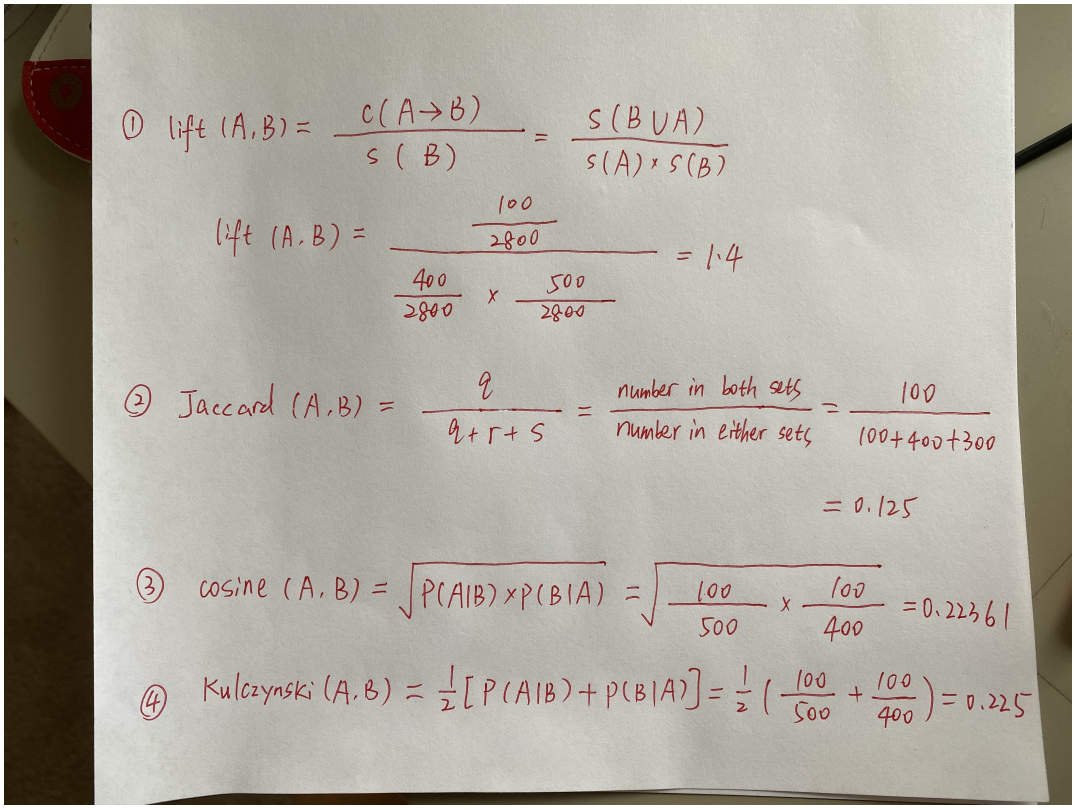
   Since χ2 statistic 166.666666667>3.841, we reject the null hypothesis. The degrees of freedom used for the hypothesis test is 1 ((2 - 1)^2) = 1.

   Therefore, the two variables 'Buy Beer' and 'Buy Diaper' are not independent.

5.

   1.

	Buy Diaper	Not Buy Diaper	row totals
Buy Beer	100	400	500
Not Buy Beer	300	2000	2300
column totals	400	2400	2800



   Lift is a simple correlation measure.

   Jaccard coefficient is similarity measure for asymmetric binary variables.

   Values of Kulczynski and cosine measures are only influenced by the supports of A, B, and A ∪ B, but not by the total number of transactions. These measures range from 0 to 1, and the higher the value, the closer the relationship between A and B.

3.

1.

support = number of occurence/total number of transaction

All the frequent 1-item set

Items	Frequency	Support
A	4	4/5 = 0.8
B	3	3/5 = 0.6
C	4	4/5 = 0.8
D	4	4/5 = 0.8
E	2	2/5 = 0.4
F	1	1/5 = 0.2
G	1	1/5 = 0.2
H	2	2/5 = 0.4

Filter on min-sup = 0.6, and we get:

Items	Frequency	Support
A	4	4/5 = 0.8
B	3	3/5 = 0.6
C	4	4/5 = 0.8
D	4	4/5 = 0.8

All the frequent 2-item set:

Items	Frequency	Support
A, B	1	1/5 = 0.2
A, C	2	2/5 = 0.4
A, D	2	2/5 = 0.4
B, C	3	3/5 = 0.6
B, D	3	3/5 = 0.6
C, D	3	3/5 = 0.6

Filter on min-sup = 0.6, and we get:

Items	Frequency	Support
A, C	3	3/5 = 0.6
A, D	3	3/5 = 0.6
B, C	3	3/5 = 0.6
B, D	3	3/5 = 0.6
C, D	3	3/5 = 0.6

All the frequent 3-item set:

Items	Frequency	Support
A, C, D	2	2/5 = 0.4
A, B, C	2	2/5 = 0.4
A, B, D	2	2/5 = 0.4
B, C, D	3	3/5 = 0.6

Filter on min-sup = 0.6, and we get:

Items	Frequency	Support
B, C, D	3	3/5 = 0.6

So the frequent k-itemset for the largest k is the 3-itemset {B, C, D} with k = 3

2.

To compute the strength of an association rule X -> Y, we need to compute s and c and see if they meet min_sup and min_conf respectively.

s = support of {x, y} =occurence of {x, y}/total transaction

and

c = The conditional probability that a transaction containing X also contains Y

= sup(X,Y)/sup(X)

The listed type of rules contain 3 items, so we only consider the 3-itemset with minsup=0.6, {B, C, D}

X	Y	s	c
{B, C}	{D}	3/5 = 0.6	3/3 = 1
{B, D}	{C}	3/5 = 0.6	3/3 = 1
{C, D}	{B}	3/5 = 0.6	3/3 = 1

Since confidence of the above three association rules are all above min_conf = 0.6, the answer is:

{B, C} -> {D}, {B, D} -> {C}, {C, D} -> {B}

3. (Q2 b)
    1.

        The type of constraint for avg(S.price) > 50 is **Convertible** constraints. It can convert tough constraints into (anti-)monotone by proper ordering of items in transactions.

        If we Order items in (price) value-descending order:

        <a, g, h, b, f, d, c, e> and if an itemset {ag} violates the constrant (avg(ag) = 30) , so does ag* (i.e., ag-projected DB).

        As we can see, the constrant is anti-monotone if patterns grow in the right order.

        In other words, the mining strategy for convertible constraints is to converted the constraint to monotonic or anti-monotonic if items can be properly ordered in processing.

    2. There are two constrants here.

        C1 : sum(S.Price with profit>5) >= 200, and C1 is **monotone**. As we know that a constraint c is monotone if an itemset S satisfies the constraint c, so does any of its superset.

        Here if itemset S satisfies C1, then no matter what item is added to S, it still satisfies C1.

        Mining strategy for monotone is that if the constraint c is satisfied, no need to check c again.

        C2 : sum up only if item.profit>5, and C2 is **data anti-monotone**. We know that:

        A constraint c is data anti-monotone: In the mining process, if a data entry t cannot contribute to a pattern p satisfying c, t cannot contribute to p's superset either.

        Here if a data entry t has t.profit<=5, t will not contribute to the sum of prices, and t can be removed or added to any itemset since its price will not be sum up and will not influence C1.

        Mining strategy for data anti-monotone is that if a transaction t does not satisfy c, then t can be pruned to reduce data processing effort.

4.
    1.

Cand.	**sup**
< a >	3
< b >	5
< c >	4
< d >	3
< e >	3
< f >	2
< g >	1
< h >	1

Since min_sup = 2, we get the following length-1 frequent sequential patterns in SDB1 and their supports:

patterns	sup
< a >	3
< b >	5
< c >	4
< d >	3
< e >	3
< f >	2

2.

Yes, < bb > is a length-2 frequent pattern in SDB1 and its support is 4

patterns	sup
< bb >	4

To find the projected database for prefix < bb >, we first find the projected database for prefix < b >

< b >-projected DB
< (_d)cb(ac) >
< (_f)(ce)b(fg) >
< (_f)abf >
< (_e)(ce)d >
< (_d)bcb(ade) >

Based on prefix < b > projected DB, we get prefix < bb > projected DB below:

< bb >-projected DB
< (ac) >
< (fg) >
< f >
< cb(ade) >

3.

Yes, < (bd) > is a length-2 frequent pattern in SDB1 and its support is 3

patterns	sup
< (bd) >	3

To find the projected database for prefix < (bd) >, we first find the projected database for prefix < b >

< b >-projected DB
< (_d)cb(ac) >
< (_f)(ce)b(fg) >
< (_f)abf >
< (_e)(ce)d >
< (_d)bcb(ade) >

Based on prefix < b > projected DB, we get prefix < (bd) > projected DB below:

< (bd) >-projected DB
< cb(ac) >
< (fg) >
< f >
< bcb(ade) >

4.

I have explained in previous question.

< b >-projected DB
< (_d)cb(ac) >
< (_f)(ce)b(fg) >
< (_f)abf >
< (_e)(ce)d >
< (_d)bcb(ade) >

5.

Find subsets of patterns: < b >-projected db, frequent:

< a >: 3

< b >: 4

< c >: 4

< d >: 2

< _d>: 2

< e >: 3

< _e >: 3

< f >: 2

< _f >: 2

All sequential patterns:

< a >,

< aa >, < ab >, < ac >, < ad >, < ae >, < af >

6.

The projected database for prefix < c >:

< c >-projected DB
< b(ac) >
< (_e)b(fg) >
< (_e)d >
< b(ade) >

7.

Find subsets of patterns: < c >-projected db, frequent:

< a >: 2

< b >: 3

< d >: 2

< e >: 3

< _e >: 3

All sequential patterns:

< c >,

< ca >, < cb >, < cd >, < ce >