Data Mining Summer Term 2020

Institute of Neural Information Processing
PD Dr. F. Schwenker

Assignment 9 (Submission until July 7, 2020)

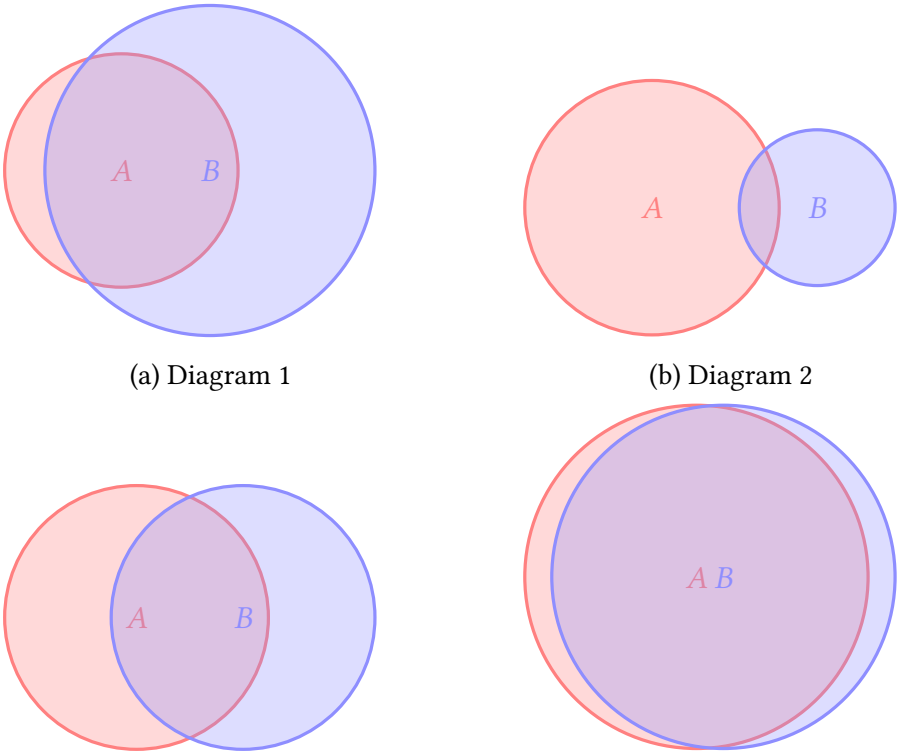
Exercise 1 (3 points): Association Rules [Pen and Paper]

With the help of association rules, we can discover relations between variables. Two important measures in this context are the support of item sets X and the confidence of item rules $X \longrightarrow Y$. In the shopping cart scenario, we can use support(X) to find relevant items (non-leftovers) and conf($X \longrightarrow Y$) tells us how appropriate the rule is in our scenario, i.e. how likely is it when purchasing X that also Y is purchased. Our goal in this exercise is to get used to these two basic measures.

1. Suppose we have one product A which is included in every transaction and one product B which is included in every second transaction. The total number of transactions is even.
   a) Calculate support(A), support(B) and support($A \cup B$).
   b) Calculate conf($A \longrightarrow B$) and conf($B \longrightarrow A$).
2. Explain based on two singleton item sets $A = \{I_1\}$ and $B = \{I_2\}$ with $I_1 \neq I_2$ why it does not make much sense to calculate support($A \cap B$).
3. Figure 1 shows four Venn diagrams for two item sets A and B. You can think of this as a graphical representation of all transactions which include the item A or B (or both). For each of the diagrams, Table 1 lists confidence values for the two rules $A \longrightarrow B$ and $B \longrightarrow A$. Map each diagram of Figure 1 to a row of Table 1.
4. Suppose we have two products A and B and already calculated the confidence values

$$\text{conf}(A \longrightarrow B) = 0.1$$
$$\text{conf}(B \longrightarrow A) = 0.3.$$

   What is higher, the support of A or the support of B, i.e. support(A) > support(B) or support(B) > support(A)?



(a) Diagram 1 (b) Diagram 2

(c) Diagram 3 (d) Diagram 4

Figure 1: Four different Venn diagrams each visualizing two item sets.

Table 1: Confidence values corresponding to the Venn diagrams in Figure 1.

Diagram	conf($A \longrightarrow B$)	conf($B \longrightarrow A$)
	0.5	0.5
	0.9	0.9
	0.0625	0.166667
	0.833333	0.416667

Exercise 2 (7 points): Apriori Algorithm [Pen and Paper]

With the help of the Apriori algorithm, we can analyse a set of transactions and find all popular item sets X which fulfil support(X) $\geq s_{\text{min}}$ as well as all rules $X \longrightarrow Y$ with conf($X \longrightarrow Y$) $\geq k_{\text{min}}$. In this exercise, we use the Apriori algorithm to extract these sets of items and rules from a small example dataset shown in Table 2.

We are interested in all sets with a minimum support of $s_{\text{min}} = 0.4$ and all rules with a minimum confidence of $k_{\text{min}} = 0.8$. Basically, the algorithm works in two steps. First, we find all relevant

item sets X which fulfil the support constraint $s_{\min}$ and second, we extract all rules from these items which fulfil the confidence constraint $k_{\min}$.

For the first step, we use the following notation:

- I_n: list of all relevant item sets with n elements, i.e. we need to make sure that

$$\forall X \in I_n \,:\, |X| = n \,\wedge\, \text{support}(X) \geq s_{\min}.$$

- I: list of all relevant item sets with arbitrary cardinality. In the end, it holds

$$\forall X \in I \,:\, \text{support}(X) \geq s_{\min}.$$

- H_n: list of all item sets with n elements where we need to check whether they fulfil the support constraint $s_{\min}$. It is based on the cross product of the previous list of relevant item sets I_n and the list of relevant singleton items I_1, i.e. $H_{n+1} = I_n \times I_1$.

Table 2: Example data with four transactions and four items.

Transaction	Beer (B)	Nappies (N)	Crisps (C)	Tomatoes (T)
$\boldsymbol{x}_1$	x	x		x
$\boldsymbol{x}_2$	x	x	x	
$\boldsymbol{x}_3$			x	
$\boldsymbol{x}_4$	x	x	x	

We can now apply the algorithm on our example transactions.

1. Let's start with the first step and search for relevant item sets. We initialize

$$H_1 = \{\{B\}, \{N\}, \{C\}, \{T\}\} \quad \text{and} \quad I = \{\}$$

since we need to check for each singleton item whether it fulfils the required support constraint. Please use Table 3 for the first step of the algorithm.

   a) Calculate the support for each $X \in H_1$.

   b) Decide for each $X \in H_1$ whether we need to add it to the list I_1. This is the case when $\text{support}(X) \geq s_{\min}$.

   c) Prepare the list H_2 for the next iteration. Do so by calculating the cross product $H_2 = I_1 \times I_1$.

   d) Repeat the process until $I_4 = \{\}$. Note that Table 3 is already prepared for the next iterations.

2. The next step is to find all rules which satisfy the required confidence level $k_{\min}$ based on the relevant item sets in $I = I_1 \cup I_2 \cup I_3$. However, instead of testing all possible rule combinations, we apply a scheme where we start with single-head rules and expand the head only if the confidence level is already high enough. We can apply this strategy in a clear way via a tree structure. This is depicted in Figure 2.

   a) The first level of the tree shows all item sets of $I \setminus I_1$. Add a node for the remaining item sets and write the calculated support value support(X) on the edge between $I \setminus I_1$ and X.

   b) The second level of the tree shows all single-head rules $X \longrightarrow Y$. Add nodes for the remaining rules and write the confidence value conf($X \longrightarrow Y$) on the edges.

   c) In the next levels of the tree, we split the existing rule nodes by expanding the head with items from the body. This is shown as an example for the rule $\{B, C\} \longrightarrow \{N\}$. We expand only rules which already fulfil the confidence constraint $k_{\min}$ and also write the confidence value on the edge. Complete the tree by adding the remaining nodes and label the empty edges. Note: it is possible that two nodes from one level point to the same node in the next level.

   d) Mark the final rules $X \longrightarrow Y$ with conf($X \longrightarrow Y$) $\geq k_{\min}$.

3. How can you tell solely from Table 2 what the minimum confidence value is for rules based on I?

4. Name a rule with conf($X \longrightarrow Y$) = 0.

Table 3: First step of the Apriori algorithm where we find relevant item sets which fulfil the support constraint $s_{\min}$.

$X \in H_n$	support(X)	Add X to I_n?
	$n = 1$	
$\{B\}$		
$\{N\}$		
$\{C\}$		
$\{T\}$		
	$n = 2$	
	$n = 3$	

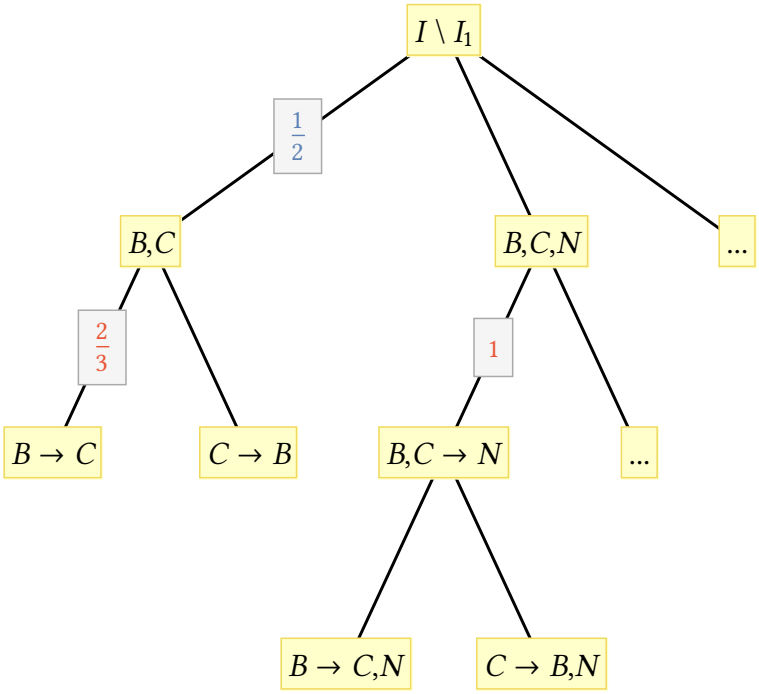


Figure 2: Tree structure used to summarize the information from the Apriori algorithm. The first level shows the relevant item sets with the corresponding support value in blue. In the other levels, rules derived from the item sets are shown with the corresponding confidence value in red. Set braces are omitted for simplicity.