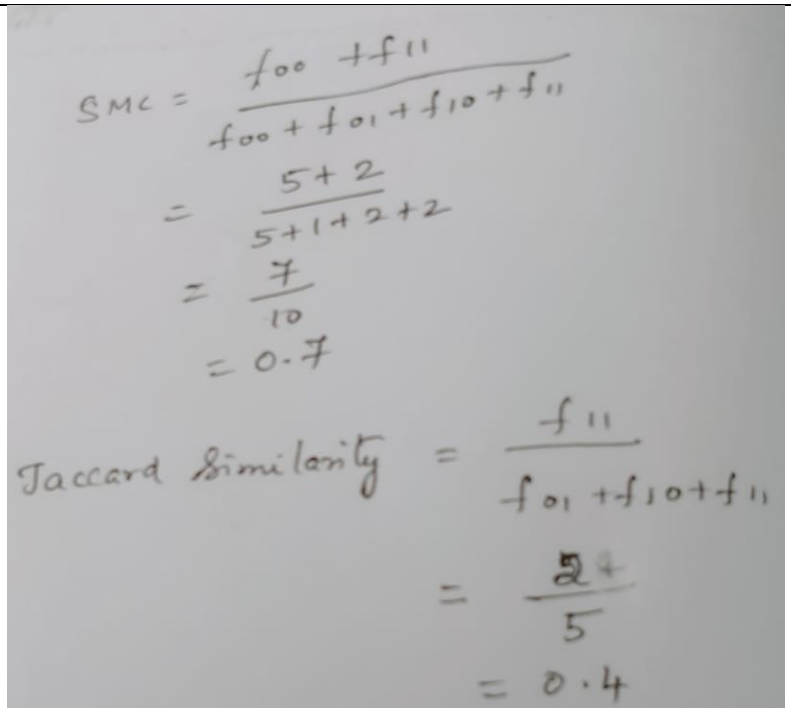
Scheme and Solution of Internal Assessment Test 2 – May. 2023

Sub:	**DATA MINING AND DATA WAREHOUSING (Professional Elective)**					Sub Code:	**18CS641**	Branch:	**CSE**	
Date:	24/05/2023	Duration:	90 mins	Max Marks:	50	Sem / Sec:	6 /A,B,C		OBE	

Answer any FIVE FULL Questions	**MARKS**	**CO**	**RBT**

1	**a) Explain the type of the attributes such as Nominal, Ordinal, Interval, Ratio.(1 mark for each type)** **Answer:** **Nominal Attributes:** Nominal attributes are categorical variables without any inherent order or hierarchy. They represent different categories or labels. Examples include gender (male/female), color (red/blue/green), or country of origin (USA/Canada/UK). Nominal attributes can only be compared for equality or inequality. **Ordinal Attributes:** Ordinal attributes also represent categorical variables, but they have a natural order or ranking. The categories have a relative position or preference, but the differences between them might not be equal. Examples include educational levels (high school/diploma/undergraduate/graduate), ratings (poor/fair/good/excellent), or survey responses with Likert scales (strongly disagree/disagree/neutral/agree/strongly agree). Ordinal attributes allow for comparison based on relative ranking or ordering. **Interval Attributes:** Interval attributes represent continuous or numerical variables where the differences between values are meaningful and consistent. However, there is no true zero point or absence of the attribute being measured. Common examples include temperature measured in Celsius or Fahrenheit, time of the day (24-hour clock), or calendar years. Interval attributes allow for comparison of differences and addition/subtraction operations, but multiplication or division by a constant might not be meaningful. **Ratio Attributes:** Ratio attributes are similar to interval attributes but have a true zero point, indicating the absence of the attribute being measured. They represent continuous or numerical variables with meaningful and consistent differences between values. Examples include height, weight, time duration, or counts. Ratio attributes allow for all arithmetic operations, including comparison, addition, subtraction, multiplication, and division. **b) Identify the type of the attributes such as Nominal, Ordinal, Interval, Ratio attributes. (1 mark for correct attribute type)** **i. Bronze, Silver and Gold medals as awarded at Sports** **ii. Gender (e.g., male, female, other)** **iii. Weight in kilograms (e.g., 60 kg, 70 kg)** **iv. Temperature on the Celsius or Fahrenheit scale (e.g., 20°C, 68°F)** **Answer:** i) Ordinal ii) Nominal iii)Ratio iv) Interval c) Find SMC and Jaccard Similarity measures for the following vector X=0101010001 Y=0100011000 **Answer:** **SMC= 7/10= 0.7 Jaccard = 2/ 5 = 0.4**	[4+4+2]	CO1	L2


SMC = $\frac{f_{00}+f_{11}}{f_{00}+f_{01}+f_{10}+f_{11}}$
$= \frac{5+2}{5+1+2+2}$
$= \frac{7}{10}$
$= 0.7$
Jaccard Similarity $= \frac{f_{11}}{f_{01}+f_{10}+f_{11}}$
$= \frac{2}{5}$
$= 0.4$

2	**a) Explain how similarity and dissimilarity measures are used in datamining.** **Answer:** Similarity and dissimilarity measures provide quantitative measures of the relationships between data objects, enabling various data mining tasks to operate on the notion of similarity. These measures help uncover patterns, relationships, or anomalies in the data, leading to valuable insights and actionable results. Clustering: Clustering algorithms group similar data objects together based on their similarity or dissimilarity. Similarity measures help determine how closely related or similar two data objects are. Common similarity measures include Euclidean distance, cosine similarity, or Jaccard similarity. These measures enable clustering algorithms to partition data into cohesive groups where objects within the same group are more similar to each other than those in different groups. **Classification:** Similarity or dissimilarity measures are used in classification tasks to compare instances and make predictions based on their similarity to known labeled examples. By measuring the similarity between an unlabeled instance and labeled instances in a training set, classification algorithms can assign a class label to the new instance. Measures such as nearest neighbor distance or similarity-based kernel functions aid in determining the most similar instances. **Recommendation Systems:** Similarity measures play a crucial role in recommendation systems. They help identify similar users or items based on their preferences, behaviors, or characteristics. By measuring the similarity between users or items, collaborative filtering techniques can generate personalized recommendations. Similarity measures like cosine similarity or Pearson correlation coefficient are commonly used to compare user preferences or item characteristics. **Anomaly Detection:** Dissimilarity measures are used in anomaly detection to identify unusual or abnormal patterns in data. By measuring the dissimilarity between a data object and a reference set or normal behavior, anomalies can be detected as objects that deviate significantly from the norm. Measures such as Mahalanobis distance or dissimilarity based on density estimation aid in identifying outliers or anomalous instances. **Data Integration:** Similarity measures are employed in data integration tasks to align or match similar records from different data sources. These measures compare attribute values or patterns to identify potential matches, duplicates, or similar entities. Similarity measures like edit distance, Jaccard index, or string similarity functions help determine the similarity between records and facilitate data integration and deduplication. **b) Calculate Cosine, Correlation and Jaccard Similarity measures for the two vectors given. X=(1,1,0,1,0,1) Y=(1,1,1,0,0,1)** **Answer:** Cosine x ● y = 1*1 + 1*1 + 0*1 + 1*0 + 0*0 + 1*1 = 3 ||x|| = sqrt(1*1 + 1*1 + 0*0 + 1*1 + 0*0 + 1*1) = 2 ||y|| = sqrt(1*1 + 1*1 + 1*1 + 0*0 + 0*0 + 1*1) = 2 cos(x,y) = (x ● y) / (||x||*||y||) = (3)/ (2 * 2) cos(x,y) = ¾ = 0.75 Correlation corr(x, y) = [covariance(x,y)] / [standard deviation(x) * standard deviation(y)] Mean of x = (1+1+0+1+0+1) / 6 = 4/6	[4+6]	CO1	L2

	Mean of y = (1+1+1+0+0+1) / 6 = 4/6 Covariance(x,y) = 1/(6-1) * [(1-4/6)(1-4/6) + (1-4/6)(1-4/6) +(0-4/6)(1-4/6) +(1-4/6)(0-4/6) + (0-4/6)(0-4/6) + (1-4/6)(1-4/6)] = (1/5)(1/3)=1/15 Standard_deviation (x) = sqrt[((1/(6-1))) * {(1-4/6)^2 + (1-4/6)^2 + (0-4/6)^2 + (1-4/6)^2 + (0-4/6)^2 + (1-4/6)^2}] = sqrt[(1/5) * (4/3)] = 0.5164 Standard_deviation (y) = sqrt[((1/(6-1))) * {(1-4/6)^2 + (1-4/6)^2 + (1-4/6)^2 + (0-4/6)^2 + (0-4/6)^2 + (1-4/6)^2}] = sqrt[(1/5) * (4/3)] = 0.5164 Corr(x,y) = (1/15) / (0.5164 * 0.5164) Corr(x,y) = 0.25 Jaccard J= (numbar of matching presences) / (number of attributes not involved in 00 matches) J=(*f*11)/(*f*01 + *f*10 + *f*11) *f*01 = 1 the number of attributes where x was 0 and y was 1 *f*10 = 1 the number of attributes where x was 1 and y was 0 *f*00 = 1 the number of attributes where x ws 0 and y was 0 *f*11 = 3 the number of attributes where x was 1 and y was 1 J= (3) / (1 + 1 + 3) J = 3/5 = 0.6			
3	a) Write the importance of data pre-processing in datamining. **Answer: (4 marks for any 4 points written)** Data pre-processing is a crucial step in data mining as it plays a significant role in ensuring the quality, usability, and effectiveness of the mining process. Here are some important reasons highlighting the significance of data pre-processing in data mining: Data Quality Improvement: Pre-processing helps identify and handle missing values, noisy data, outliers, and inconsistencies in the dataset. By addressing these issues, data quality is improved, leading to more accurate and reliable results during the mining process. Feature Selection and Extraction: Pre-processing enables the identification and selection of relevant features or attributes that have the most impact on the mining task. It helps remove redundant or irrelevant features, reducing dimensionality and enhancing computational efficiency while maintaining the discriminatory power of the data. Normalization and Scaling: Pre-processing techniques such as normalization and scaling ensure that different attributes or features are on a comparable scale. This is important for algorithms that rely on distance or similarity measures, as it prevents certain features from dominating others and ensures fair comparisons. Handling Categorical Data: Many data mining algorithms operate on numerical data, so pre-processing is required to handle categorical variables. Techniques such as one-hot encoding or ordinal encoding	[4+6]	CO1	L2

are applied to convert categorical attributes into numerical representations that can be effectively utilized by the algorithms.

Handling Imbalanced Data: In real-world datasets, class imbalance is a common issue where certain classes are underrepresented. Pre-processing techniques such as oversampling or undersampling can be employed to balance the class distribution, preventing bias and ensuring fair representation during mining.

Handling Text and Unstructured Data: Text and unstructured data require specialized pre-processing techniques such as tokenization, stemming, stop-word removal, and feature extraction (e.g., TF-IDF, word embeddings) to convert them into a structured format suitable for analysis by data mining algorithms.

Reducing Computational Complexity: Data pre-processing helps reduce computational complexity by eliminating unnecessary data, reducing noise, and optimizing the representation of the data. This results in faster and more efficient execution of data mining algorithms.

Enhancing Interpretability: Pre-processing techniques such as dimensionality reduction and feature selection can improve the interpretability of the mining results. By reducing the number of features or transforming the data into a more interpretable space, the extracted patterns and relationships become more understandable to domain experts.

b) Explain the following data-preprocessing techniques in detail.
   i) Aggregation
   ii) Sampling
   iii) Dimensionality reduction

Answer:

Aggregation:

Aggregation refers to the process of combining multiple data values or observations into a single representation or summary. It involves collecting and grouping data together to derive meaningful insights or to simplify data analysis. Aggregation is commonly used in data mining and data analysis to condense and summarize large datasets, enabling a more concise and informative representation of the data.

Sampling:

Sampling is a technique used in data analysis to select a subset of data from a larger population or dataset. It involves the process of choosing representative samples that can provide insights and draw accurate conclusions about the entire population. Sampling is widely used in various fields, including statistics, market research, surveys, and data mining.

Sampling Methods: There are different sampling methods, each with its own advantages and considerations. Some commonly used sampling methods include simple random sampling, stratified sampling, cluster sampling, and systematic sampling. The choice of sampling method depends on the research objectives, population size, available resources, and the desired level of precision.

	Dimensionality reduction: Dimensionality reduction is a technique used in data analysis and machine learning to reduce the number of variables or features in a dataset while preserving its essential information. It aims to overcome the curse of dimensionality, where datasets with a large number of features can suffer from increased computational complexity, reduced interpretability, and overfitting. Dimensionality reduction techniques extract a lower-dimensional representation of the data that captures the most important or relevant information.			
4	a) Explain the following terminology with respect to association rule mining: i) Frequent itemset ii) Support iii) Confidence **Answer:** **Frequent Itemset:** In association rule mining, a frequent itemset refers to a set of items that frequently co-occur together in a transactional dataset. An itemset can contain one or more items. **Support:** Support is a measure that indicates the frequency or occurrence of an itemset in a dataset. It represents the proportion of transactions in the dataset that contain the itemset. Support is typically expressed as a percentage or a decimal value between 0 and 1. High support values indicate that the itemset occurs frequently, suggesting a strong association between the items. Support is used to identify frequent itemsets, and a minimum support threshold is set to determine which itemsets are considered frequent. **Confidence:** Confidence is a measure of the strength of an association rule between two itemsets. It indicates the likelihood of one itemset (called the antecedent) being associated with another itemset (called the consequent). Confidence is calculated by dividing the support of the combined itemset (antecedent and consequent) by the support of the antecedent alone. Confidence is expressed as a percentage or a decimal value between 0 and 1. Higher confidence values indicate a stronger association between the itemsets. **b) What are the ways to reduce the computational complexity of frequent itemset generation** **Answer:** 1. Reducing the candidate itemsets generation using Apriori principle. 2. Reducing the number of comparisons by using advanced data structures and parallelising the operations.	[6+4]	CO2	L2
5	a) Write Apriori Principle and explain how this is used for frequent itemset generation and compare it with brute force approach. **Answer:** The Apriori principle is a fundamental concept in association rule mining that exploits the downward closure property to reduce the search space for frequent itemset generation. It is based on the observation that if an itemset is infrequent, all of its supersets	[4+6]	CO2	L2

(itemsets containing additional items) will also be infrequent. This principle forms the basis of the Apriori algorithm, which efficiently discovers frequent itemsets in a large dataset. Here's an explanation of how the Apriori principle is used for frequent itemset generation and a comparison with the brute force approach:

Apriori Algorithm: The Apriori algorithm follows a level-wise search strategy to generate frequent itemsets. It starts by finding frequent 1-itemsets (individual items) and then iteratively extends the itemsets to higher dimensions until no new frequent itemsets can be found. The algorithm uses the Apriori principle to prune the search space by eliminating candidate itemsets that are known to be infrequent.

Candidate Generation: The Apriori algorithm generates candidate itemsets by joining frequent (k-1)-itemsets with each other. For example, if {A, B} and {A, C} are frequent 2-itemsets, their join operation produces {A, B, C}. The algorithm then prunes the candidate itemsets by examining their subsets to ensure that all of their subsets are frequent. If any subset is found to be infrequent, the candidate itemset is discarded.

Comparison with Brute Force Approach: The brute force approach, also known as the "all itemsets" approach, involves considering all possible itemsets in the dataset and computing their support to identify the frequent ones. This approach has exponential computational complexity since the number of possible itemsets grows exponentially with the number of items. In contrast, the Apriori algorithm leverages the Apriori principle to significantly reduce the search space by pruning candidate itemsets that are known to be infrequent. This pruning step helps avoid the need to compute the support for all possible itemsets, resulting in improved efficiency.

Efficiency and Scalability: The Apriori algorithm's efficiency and scalability are achieved through the Apriori principle and the pruning steps. By eliminating infrequent itemsets and avoiding unnecessary support calculations, the algorithm reduces the computational complexity and enables efficient frequent itemset generation. This makes it suitable for mining large datasets and discovering interesting associations efficiently.

b) Consider the data set shown in Table below.

Customer ID	Transaction ID	Items Bought
1	0001	$\{a, d, e\}$
1	0024	$\{a, b, c, e\}$
2	0012	$\{a, b, d, e\}$
2	0031	$\{a, c, d, e\}$
3	0015	$\{b, c, e\}$
3	0022	$\{b, d, e\}$
4	0029	$\{c, d\}$
4	0040	$\{a, b, c\}$
5	0033	$\{a, d, e\}$
5	0038	$\{a, b, e\}$

Compute the support for itemsets {a}, {b, d}, and {b, d, e} by treating each transaction ID as a market basket.

Answer:

{a} = 7/10 =0.7

{b,d} = 2/10 =0.2

{b,d,e} =1/10=0.1

6 Consider the following Market basket transaction DB [10] CO2 L3

Transaction ID	Items Bought
1	{Milk, Beer, Diapers}
2	{Bread, Butter, Milk}
3	{Milk, Diapers, Cookies}
4	{Bread, Butter, Cookies}
5	{Beer, Cookies, Diapers}
6	{Milk, Diapers, Bread, Butter}
7	{Bread, Butter, Diapers}
8	{Beer, Diapers}
9	{Milk, Diapers, Bread, Butter}
10	{Beer, Cookies}

a) What is the maximum number of association rules that can be extracted from this data (including rules that have zero support)? (1 mark)

Answer:

There are six items in the data set. Therefore the total number of rules is 602.

$$\mathbf{R} = 3^d - 2^{d+1} + 1$$

d=6, hence R= 602

b) What is the maximum size of frequent itemsets that can be extracted (assuming minsup > 0)? (1 mark)

Answer:

Because the longest transaction contains 4 items, the maximum size of frequent itemset is 4.

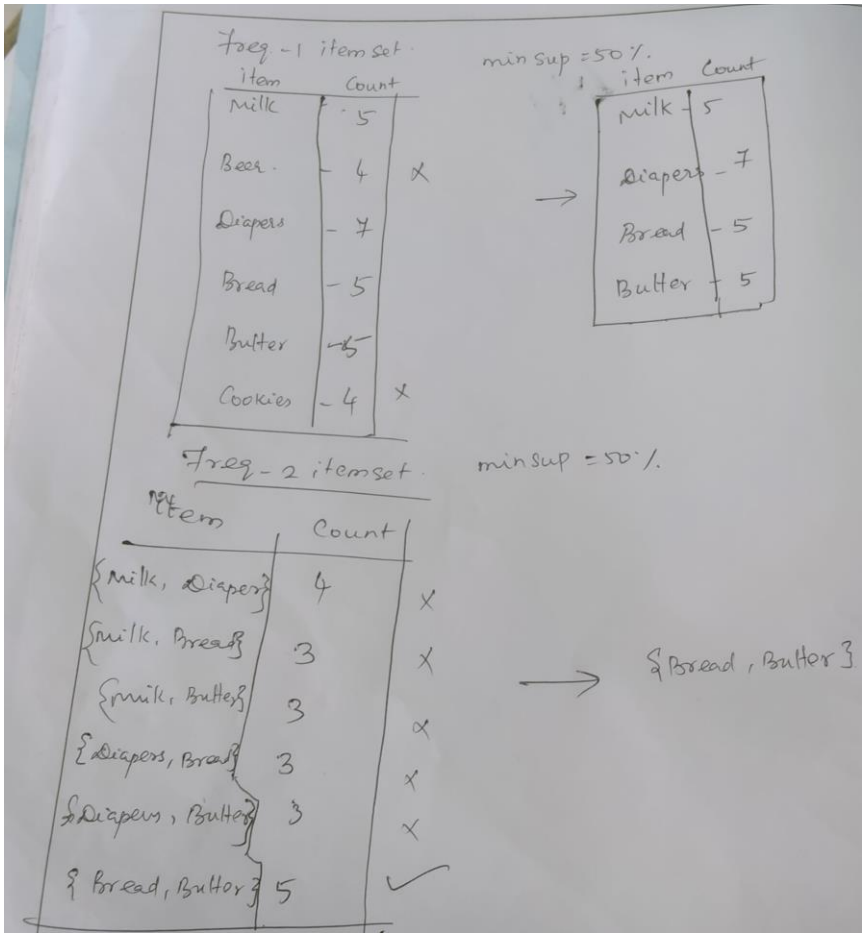
c) Write an expression for the maximum number of size-3 itemsets that can be derived from this data set. (1 mark)

Answer:

$$\binom{6}{3} = 20$$

d)Find all the frequent itemsets for the above transaction database by considering minsup=50% (4 marks)

Answer:



e) Find all strong association rules by considering minconf= 50% (3 marks)

Answer:

{Bread}-> { Butter} - 50% conf.

{Butter}->{ Bread}- 50% conf.