USN | | | | | | | | | | |

Internal Assessment Test 3 – December 2022

Sub:	BIG DATA AND ANALYTICS					Sub Code:	18CS72	Branch:	ISE
Date:	27/12/2022	Duration:	90 min's	Max Marks:	50	Sem/Sec:	VII / A, B & C		OBE

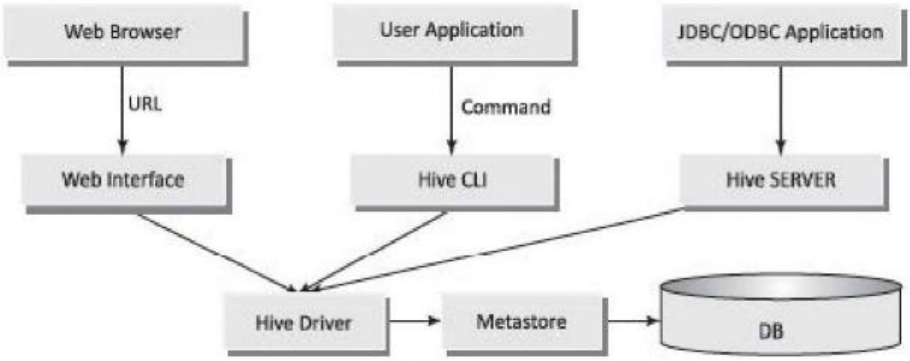
	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	**Illustrate main features and Architecture of Hive with neat diagram.** **Scheme: Architecture of HIVE with diagram = 10M** **Solution:** Components of Hive architecture are: • **Hive Server (Thrift)** – An optional service that allows a remote client to submit requests to Hive and retrieve results. Requests can use a variety of programming languages. Thrift Server exposes a very simple client API to execute HiveQL statements. • **Hive CLI (Command Line Interface)** – Popular interface to interact with Hive. Hive runs in local mode that uses local storage when running the CLI on a Hadoop cluster instead of HDFS. • **Web Interface** – Hive can be accessed using a web browser as well. This requires a HWI Server running on some designated code. The URL *http:// hadoop:<port no.> / hwi* command can be used to access Hive through the web. • **Metastore** – It is the system catalog. All other components of Hive interact with the Metastore. It stores the schema or metadata of tables, databases, columns in a table, their data types and HDFS mapping. • **Hive Driver** – It manages the life cycle of a HiveQL statement during compilation, optimization and execution. 	[10]	CO4	L2
2	**Using HiveQL for the following:** **Scheme: create table + Alter Table = 4+6 = 10M** **Solution:** **i) Create table with partition**	[10]	CO4	L2

Table Partitioning

Create a table with Partition using command:

```
CREATE [EXTERNAL] TABLE <table name> (<column name 1>
<data type 1>, …....)
PARTITIONED BY (<column name n> <data type n> [COMMENT
<column comment>], ...);
```

ii) Add, rename and drop a partition to a table

Rename a Partition in the existing Table using the following command:

```
ALTER TABLE <table name> PARTITION partition_spec
RENAME TO PARTITION partition_spec;
```

Add a Partition in the existing Table using the following command:

```
ALTER TABLE <table name> ADD [IF NOT EXISTS] PARTITION
partition_spec
[LOCATION 'location1'] partition_spec [LOCATION
'location2'] ...;
partition_spec:(p_column = p_col_value, p_column =
p_col_value, ...)
```

Drop a Partition in the existing Table using the following command:

```
ALTER TABLE <table name> DROP [IF EXISTS] PARTITION
partition_spec, PARTITION partition_spec;
```

3 | **Describe Pig data types and operator: Group, Filter, Limit, Split.** | [10] | CO4 | L2

Scheme: Data types+Group+Filter+Limit+Split = 2+2+2+2+2 = 10M

Solution:

Pig data types

Data type	Description	Example
bag	Collection of tuples	{(1,1), (2,4)}
tuple	Ordered set of fields	(1,1)
map (data map)	Set of key-value pairs	[Number#1]
int	Signed 32-bit integer	10
long	Signed 64-bit integer	10L or 10l
float	32-bit floating point	22.7F or 22.7f
double	64-bit floating point	3.4 or 3.4e2 or 3.4E2
chararray	Char [], Character array	data analytics
bytearray	BLOB (Byte array)	ff00

Operators: Group, Filter, Limit, Split.

Group GROUP statement collects records with the same key. There is no direct connection between group and aggregate functions in Pig Latin unlike SQL.

Collects all records with the same value for the provided key into a bag. Then it can pass to aggregate function, if required or do other things with that.	`A = load 'input' as (name: chararray, rollno:long, marks: float);`
	`grpd = group A by marks;`
	`B = foreach grpd generate name, COUNT(A);`

Filter FILTER gives a simple way to select tuples from a relation based on some specified conditions (predicate). It is Pig's *select* command.

Loads an entire record, then selects the tuples with marks more than 75 from each record	`A = load 'input' as (name:chararray, rollno:long, marks:float);` `B = filter A by marks > 75.0;`
Find name (chararray) that do not match a regular expression by preceding the text without a given character string. Output is all names that do not start with P.	`A = load 'input' as (name:chararray, rollno:long, marks:float);` `B = filter A by not name matches 'P.*';`

Limit LIMIT gets the limited number of results.

Outputs only first five tuples from the relation.	`A = load 'input' as (name: chararray, city: chararray);` `B = Limit A 5;`

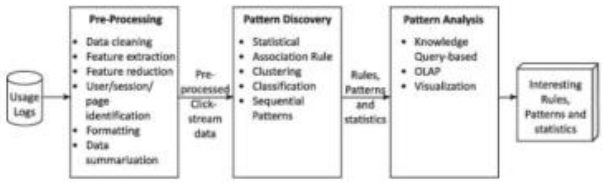
Split SPLIT partitions a relation into two or more relations

Outputs A relation `A` splits into two relations `P` and `Q`	`A = load 'input' as (name:chararray, rollno:long, marks:float);` `Split A into P if marks >50.0, Q if marks ≤ 50.0;`

4	**Describe the web content mining and three phases for web usage mining.** **Scheme: web content mining + web usage mining = 5+5 =10M** **Solution:** **Web Content Mining:** **Web Content Mining** is the process of information or resource discovery from the content of web documents across the World Wide Web. Web content mining can be (i) direct mining of the contents of documents or (ii) mining through search engines. They search fast compared to direct method. Web content mining relates to both, data mining as well as text mining. Following are the reasons: (i) The content from web is similar to the contents obtained from database, file system or through any other mean. Thus, available data mining techniques can be applied to the web. (ii) Content mining relates to text mining because much of the web content comprises texts. (iii) Web data are mainly semi-structured and/or unstructured, while data mining is structured and the text is unstructured. ***Applications*** Following are the applications of content mining from web documents: 1. Classifying the web documents into categories 2. Identifying topics of web documents 3. Finding similar web pages across the different web servers 4. Applications related to relevance: **Web Usage Mining:**	[10]	CO5	L2

Web usage mining discovers and analyses the patterns in click streams. Web usage mining also includes associated data generated and collected as a consequence of user interactions with web resources.

Figure 9.7 shows three phases for web usage mining.



The phases are:

1. Pre-processing – Converts the usage information collected from the various data sources into the data abstractions necessary for pattern discovery.
2. Pattern discovery – Exploits methods and algorithms developed from fields, such as statistics, data mining, ML and pattern recognition.
3. Pattern analysis – Filter outs uninteresting rules or patterns from the set found during the pattern discovery phase.

Usage data are collected at server, client and proxy levels. The usage data collected at the different sources represent the navigation patterns of the overall web traffic. This includes single-user, multi-user, single-site access and multi-site access patterns.

5 **In Machine learning, explain linear and non linear relationship with essential graphs.** [10] CO5 L2

Scheme: linear and nonlinear relationship with graph = 10M

Solution:

Correlation is a statistical technique that measures and describes the 'strength' and 'direction' of the relationship between two variables.

Correlation means analysis which lets us find the association or the absence of the relationship between two variables, x and y. Correlation gives the strength of the relationship between the model and the dependent variable on a convenient 0-100% scale.

R-Square R is a measure of correlation between the predicted values y and the observed values of x. *R-squared* (R^2) is a goodness-of-fit measure in linear-regression model. It is also known as the coefficient of determination. R^2 is the square of R, the coefficient of multiple correlations, and includes additional independent (explanatory) variables in regression equation.

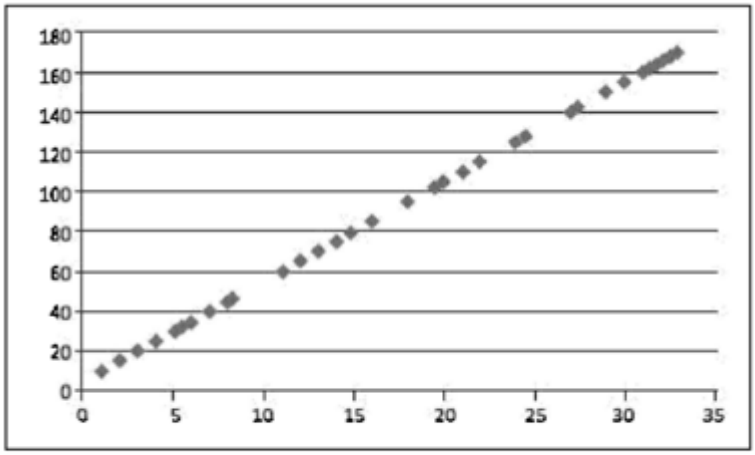
(i) Constrained Pearson correlation – It is a variation of Pearson correlation that uses midpoint instead of mean rate.

(ii) Spearman rank correlation – It is similar to Pearson correlation, except that the ratings are ranks.

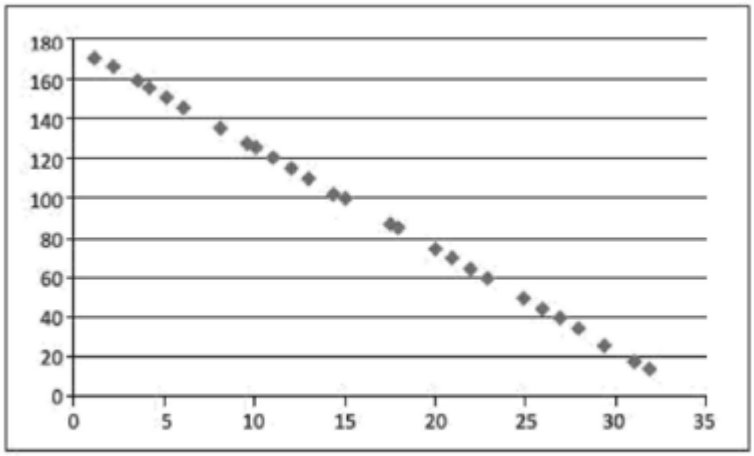
(iii) Kendall's G correlation – It is similar to the Spearman rank correlation, but instead of using ranks themselves, only the relative ranks are used to calculate the correlation.

Table 6.1 The strength of the relationship as a function of r

Value of r	Strength of relationship
-1.0 to -0.5 or 1.0 to 0.5	Strong
-0.5 to -0.3 or 0.3 to 0.5	Moderate
-0.3 to -0.1 or 0.1 to 0.3	Weak
-0.1 to 0.1	None or very weak



Perfect Positive Linear Relationship ($r = 1$)



Perfect Negative Linear Relationship ($r = -1$)

6	How does the Apriori algorithm work? For the following table, find frequent item set and describe the different steps of forming Association rules using Apriori algorithm. Consider minimum support count as 2. **Scheme: Apriori algorithm + Association rule = 2+8 = 10M** **Solution:**	[10]	CO5	L3

ID	List of Items
T1	I1, I2, I5
T2	I2, I4
T3	I2, I3
T4	I1, I2, I4
T5	I1, I3
T6	I2, I3
T7	I1, I3
T8	I1, I2, I3, I5
T9	I1, I2, I3

The Apriori principle can reduce the number of itemsets needed to be examined. Apriori principle suggests if an itemset is frequent, then all of its subsets must also be frequent. For example, if itemset {A, B, C} is a frequent itemset, then all of its subsets {A}, {B}, {C}, {A, B}, {B, C} and {A, C} must be frequent. On the contrary, if an itemset is not frequent, then none of its supersets can be frequent.

Assume X and Y are two itemsets. Apriori principle holds due to the following property of support measure:

$$\forall\ X, Y: (X \subseteq Y) \rightarrow s(X) \geq s(Y) \qquad (6.24)$$

Explanation: $\forall$ means for all, and $\subseteq$ means 'subset of' and can be 'equal to or included in'. Support of an itemset never exceeds the support of its subsets. This is known as the *anti-monotone property* of support.

Apriori algorithm evaluates candidates for association as follows:

C_k: Set of candidate-itemsets of size k

F_i: Set of frequent itemsets of size k

F_1 = {large items}

for (k=1; F_k != 0; k++) *do* {

C_{k+1} = New candidates generated from F_k

for each transaction t in the database *do*

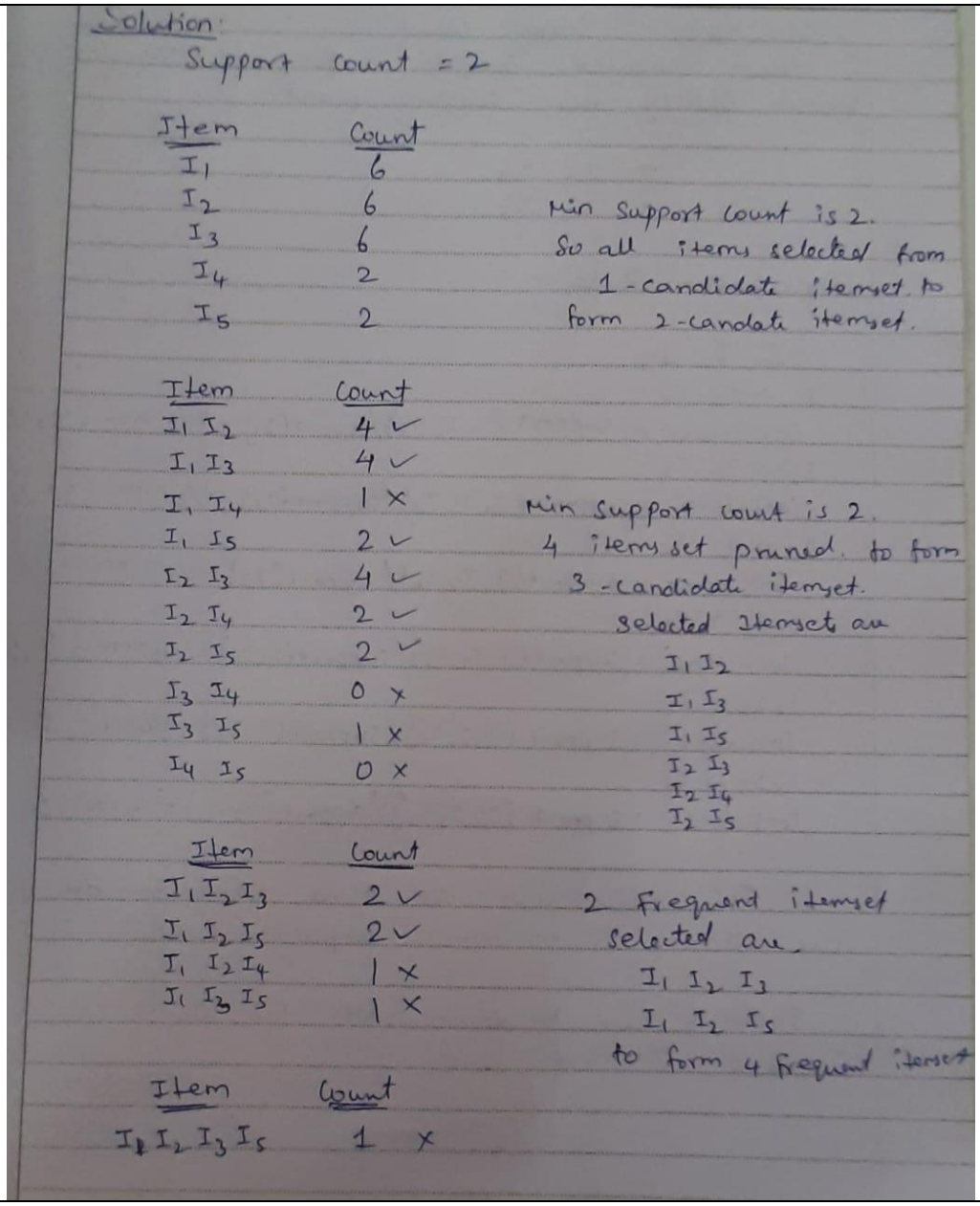
Increment the count of all candidates in C_{k+1} that are contained in t

F_{k+1} = Candidates in C_{k+1} with minimum support

}

Steps of the algorithm can be stated in the following manner:

1. Candidate itemsets are generated using only large itemsets of the previous iteration. The transactions in the database are not considered while generating candidate itemsets.
2. The large itemset of the previous iteration is joined with itself to generate all itemsets having size higher by 1.
3. Each generated itemset that does not have a large subset is discarded. The remaining itemsets are candidate itemsets.



Solution:

Support count = 2

Item	Count
I_1	6
I_2	6
I_3	6
I_4	2
I_5	2

Min Support count is 2.
So all items selected from 1-candidate itemset to form 2-candate itemset.

Item	Count
$I_1\ I_2$	4 ✓
$I_1\ I_3$	4 ✓
$I_1\ I_4$	1 ✗
$I_1\ I_5$	2 ✓
$I_2\ I_3$	4 ✓
$I_2\ I_4$	2 ✓
$I_2\ I_5$	2 ✓
$I_3\ I_4$	0 ✗
$I_3\ I_5$	1 ✗
$I_4\ I_5$	0 ✗

Min Support count is 2.
4 items set pruned to form 3-candidate itemset.
Selected Itemsets are
$I_1\ I_2$
$I_1\ I_3$
$I_1\ I_5$
$I_2\ I_3$
$I_2\ I_4$
$I_2\ I_5$

Item	Count
$I_1\ I_2\ I_3$	2 ✓
$I_1\ I_2\ I_5$	2 ✓
$I_1\ I_2\ I_4$	1 ✗
$I_1\ I_3\ I_5$	1 ✗

2 Frequent itemset selected are.
$I_1\ I_2\ I_3$
$I_1\ I_2\ I_5$
to form 4 frequent itemset

Item	Count
$I_1\ I_2\ I_3\ I_5$	1 ✗

$I_1 I_2 I_3$ I_5 Support count is 1 so it is not selected. Consider 3 frequent Itemset which satisfy min. support count are.

I_1 I_2 I_3
I_1 I_2 I_5

Association Rule :-

I_1 I_2 I_3

$I_1 \rightarrow I_2, I_3$
Confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_1\}$ = 2/6 × 100 = 33%.

$I_2 \rightarrow I_1, I_3$
Confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_2\}$ = 2/6 × 100 = 33%.

$I_3 \rightarrow I_1, I_2$
Confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_3\}$ = 2/6 × 100 = 33%.

$I_1, I_2 \rightarrow I_3$
confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_1, I_2\}$ = 2/4 × 100 = 50%.

$I_1, I_3 \rightarrow I_2$
Confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_1, I_3\}$ = 2/4 × 100 = 50%.

$I_2 I_3 \rightarrow I_1$
Confidence = support $\{I_1, I_2, I_3\}$ / support $\{I_2, I_3\}$ = 2/4 × 100 = 50%.

This shows that last three transactions ~~or~~ rules are strong if minimum confidence threshold is 50%.

Similarly need to derive rules for $I_1, I_2 I_5$ frequent Itemset also.