DATA MINING

Module 1: Introduction

1. Data Mart

A data mart is a data storage system that contains information specific to an organization’s business unit. It contains a small and selected part of the data which the company stores in a larger storage system. Companies use a data mart to analyze department-specific information more efficiently. E.g. the advertising department will have its own data mart, while the financial department will have its separate data mart.

Types ->

* Independent data marts: These are standalone entities which might or might not be attached to a central warehouse.
* Dependent data marts: These are populated from a central data repository.

1. Data Mining

Data mining is the process of sorting through large data sets to identify patterns and relationships that can help to solve business problems through data analysis. Data mining techniques and tools help enterprises to predict future trends and make more informed business decisions.

Data mining is a key part of data analytics and one of the core disciplines in data science, which uses advanced analytics techniques to find useful information in data sets. At a more granular level, data mining is a step in the knowledge discovery database (KDD) process.

Advantages of data mining over traditional approaches:

* **Cost-efficiency** – Data mining is more cost-efficient than other statistical data applications.
* **Uncovers hidden patterns** – Data mining can automatically discover patterns, trends, and relationships in large amounts of data that might otherwise go unnoticed.
* **Help make decisions** – Data mining can help organizations make data-driven decisions, identifying new business opportunities.
* **Can be used with any application.**
* **Can help with customer service** – by analyzing customer behavior and preferences.

1. Data Warehouse (and characteristics)

Data warehouse is a type of data management system that is designed to enable and support business intelligence (BI) activities, especially analytics. Data warehouses are solely intended to perform queries and analysis and often contain large amounts of historical data. The data within a data warehouse is usually derived from a wide range of sources such as application log files and transaction applications.

A data warehouse centralizes and consolidates large amounts of data from multiple sources. Its analytical capabilities allow organizations to derive valuable business insights from their data to improve decision making.

Characteristics of Data Warehouse:

* **Subject Oriented** – A data warehouse is always subject oriented as it delivers information about a theme instead of organization’s current operations. Themes can be sales, distributions, marketing, etc.
* **Integrated** – It is somewhere same as subject orientation which is made in a reliable format. Integration means founding a shared entity to scale all the similar data from the different databases.
* **Time-Variant** – Data is maintained via different time intervals such as weekly, monthly or annually. It finds various time limits which are structured between the large datasets and are held in online transaction processes (OLTP).
* **Non-Volatile** – Data resided in data warehouse is permanent. It also means that data is not erased or deleted when new data is inserted.

1. OLAP vs OLTP

|  |  |  |
| --- | --- | --- |
| Features | OLTP – Online Transaction Processing | OLAP – Online Analytical Processing |
| Characteristics | Operational processing | Information Processing |
| Orientation | Transaction analysis | Analysis of data |
| User | DB professionals | Knowledge workers (Data scientist, data analyst) |
| Functions | Day-to-day operations | Long term informational requirements (Decision support system)[DSS] |
| Data | Current transaction | Historical data (Accuracy maintained over time) |
| Unit of Work | Simple transaction queries | Complex queries |
| Response Time | Requires faster response time | Can have longer response time |
| Access | Read and write both | Mostly read |
| Focus | Data input | Information output |
| No of records accessed | 100s of data | Millions of data (Big Data) |
| No of users | Thousands | 100s (Specific requirement) |
| DB size | GB to higher order GB | >=TB |

1. Typical 3-tier data warehousing architecture

Three tiers of a 3-tier data warehousing architecture are ->

* Botton Tier (Data Warehouse Server)

Bottom Tier consists of the Data Warehouse Server which is almost always an RDBMS. It may include several specialized data marts and a metadata (data about data) repository.

Data from operational databases and external sources are extracted using application program interfaces called a gateway. A gateway is provided by the underlying DBMS and allows customer programs to generate SQL code to be executed at a server.

Examples of gateway: ODBC (Open Database Connection), OLE-DB (Open-Linking and Embedding for Databases), JDBC (Java Database Connection), etc.

* Middle Tier (OLAP Server)

Middle-tier consists of an OLAP server for fast querying of the data warehouse. The OLAP server is implemented using any of the following:

1. A Relational OLAP [ROLAP] model – An extended relational DBMS that maps functions on multidimensional data to standard relational operations.
2. A Multi-dimensional OLAP [MOLAP] model – A particular purpose server that directly implements multidimensional information and operations.

* Top Tier (Front end Tools)

The top-tier consists of front-end tools for displaying results provided by OLAP, as well as additional tools for data mining of the OLAP-generated data.

1. Design steps for a typical data warehouse
2. **Extracting the transactional data from the data sources into a staging area**

This step covers the data extraction from the source system and makes it accessible for further processing.

The main objective of the extract step is to retrieve all the required data from the source system with as little resources as possible.

The extract step should be designed in such a way that it does not negatively affect the source system in terms of performance, response time or any kind of locking.

1. **Transforming the transactional data**

The transform step applies a set of rules to transform the data from the source to the target. This includes converting any measured data to the same dimension using the same units so that they can be joined later.

The transform step also requires joining data from several sources, generating aggregates, generating surrogate keys, sorting, deriving new calculated values, and applying validation rules.

1. **Loading the transformed data into a dimensional database**

During the load step, it is essential to ensure that the load is performed correctly and with as little resources as possible.

In order to make the load process efficient, it is helpful to disable all constraints and indexes before the load and enable them only after the load is completed. The referential integrity needs to be maintained by ETL tool to ensure consistency.

1. Warehousing schema

At the heart of every data warehouse, lies a schema. But data warehouse schemas are more than just technical blueprints.

A data warehouse schema describe how data is organized, stored, and related. The schema serves as the template for constructing and populating a data warehouse, dictating the structure of data tables, their relationships, and the rules governing data integrity and consistency.

Three popular data warehousing schemas are: Star, Snowflake and Galaxy.

1. Association Rules in Data Mining

Association rules are if-then statements that identify the relationships or dependencies between the data. With the characteristic property of suiting numeric and non-numeric categorical data, it’s often applied in market basket analysis and other applications.

Give Market Basket example with Support, Confidence and lift formulas.

1. Difference between Data Mart and Data Warehouse

|  |  |  |
| --- | --- | --- |
| Sl No | Data Warehouse | Data Mart |
| 1 | Centralized System | Decentralized System |
| 2 | Light denormalization takes place | High denormalization takes place |
| 3 | Top-down model | Bottom-up model |
| 4 | Its difficult to make a data warehouse | It’s easier to make a data mart |
| 5 | In data warehouse, fact constellation schema is used. | Star schema and snowflake schema are used. |
| 6 | Data warehouse is flexible | Data mart is not flexible |
| 7 | Data warehouse has a long life | Data mart has a shorter life than warehouse |
| 8 | Vast in size | Smaller in size |

1. Sequence Mining

Sequence mining is a technique in data mining that focuses on identifying and extracting patterns or regularities within sequences of events or items. It’s useful in scenarios where the order of events is important, such as analyzing customer purchasing behavior, web clickstreams, biological sequences or sensor data.

Key concepts in sequence mining ->

* Sequences – A sequence is an ordered list of events or items.
* Frequent Patterns – Subsequences that appear often within the dataset.
* Sequential Rules – These are implications in ‘if-then’ form.
* Support – A measure of how often a sequence or pattern appears in the dataset.
* Confidence – A measure of likelihood that a sequential rule is true.

Applications of sequence mining ->

* Market Basket Analysis (Example done in class)
* Web Usage Mining
* Biological data analysis
* Sensor Data analysis

Common Algos of sequence mining:

* Apriori Algorithm
* PrefixSpan
* SPADE

1. Enterprise Warehouse, Data Mart, Virtual Warehouse

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Enterprise Warehouse | Data Mart | Virtual Warehouse |
| Definition | Centralized repository consolidating data from across the entire organization. | Subset of a data warehouse focused on a specific department or business area. | Logical or virtual view of data aggregated from multiple sources without physical consolidation. |
| Scope | Organization wide | Departmental or business unit specific | Organization wide but virtual |
| Integration | High level of integration and consistency | Moderate, focused on specific needs of a department | Integration achieved virtually, generally via middleware |
| Size | Very large | Smaller | Variable (Depends on underlying sources) |
| Usage | * Extensive analysis * Reporting * Decision making | * Specific departmental analysis * Reporting | * Real-time queries * Quick access to up-to-date data |

1. Data warehouse vs database

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Data-base | Data lake | Data Warehouse |
| Workloads | Operational and transactional | Analytical | Analytical |
| Data type | Structured or semi-structured | Structured, semi structured and/or unstructured | Structured and/or semi structured |
| Schema Flexibility | Rigid or flexible depending on DB type | No schema definition is required | Pre-defined and fixed schema |
| Data Freshness | Real time | May not be up-to-date based on frequency of ETL processes | May not be up-to-date based on frequency of ETL processes |
| Users | Application developers | Business analysts, application developers, data scientists | Business analysts and data scientists |
| Example | Relational DB – Oracle, MySQL, PostgreSQL  Document DBs – MongoDB, CouchDB  Graph DBs – Neo4j, Amazon Neptune | Databricks SQL Analytics, AWS S3, Google Cloud Storage, etc. | Google BigQuery, IBM Db2 Warehouse, Microsoft Azure Synapse, Snowflake, etc. |

1. Knowledge Discovery Database (With Steps)

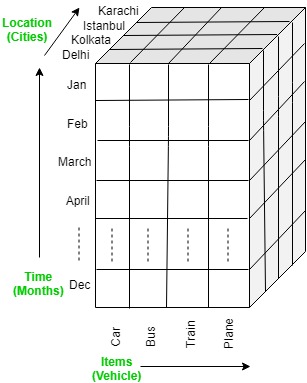
Steps to the KDD process are ->

1. Data Selection – Data relevant for analysis task is retrieved from DB.
2. Data integration – Multiple data sources may be combined
3. Data Cleaning – To remove noise and inconsistent data
4. Data Transformation – Where data are transformed or consolidated into forms
5. Data mining – An essential process where intelligent methods are applied in order to extract data patterns.
6. Pattern Evaluation – To identify the truly interesting patterns representing knowledge based on some interesting measures.
7. Knowledge Representation – Where visualization and knowledge representation techniques are used to present the mined knowledge to the user.
8. Roll up and Drill down processes

These are operations of OLAP.

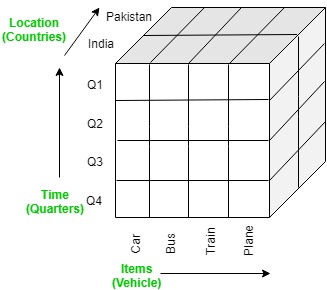
1. Drill down – In drill-down operation, the less detailed data is converted into highly detailed data. It can be done by:

* Moving down in the concept hierarchy.
* Adding a new dimension.



1. Roll up – It’s just the opposite of drill down. It performs aggregation of the OLAP cube. It can be done by:

* Climbing up the concept hierarchy.
* Reducing the dimensions.



1. Issues and challenges in data mining

Some of the main challenges of data mining are:

1. Data Quality

The quality of data used in data mining is one of the most significant challenges. The accuracy, completeness, and consistency of the data affect the accuracy of the results obtained. The data may contain errors, omissions, duplications, or inconsistencies, which may lead to inaccurate results.

Data quality issues can arise due to a variety of reasons, including data entry errors, data storage issues, data integration problems, and data transmission errors.

1. Data Complexity

Data complexity refers to vast amounts of data generated by various sources, such as sensors, social media, and the internet of things (IoT). The complexity of the data makes it challenging to process, analyze and understand. In additions, the data may be in different formats, making it challenging to integrate into a single dataset.

1. Data Privacy and Security

Data security is another significant challenge in data mining. As more data is collected, stored and analyzed, the risk of data breaches and cyber-attacks increases. The data may contain personal, sensitive, or confidential information that must be protected. Moreover, data privacy regulations such as GDPR, CCPA, and HIPAA impose strict rules on how data can be collected, used, and shared.

1. Scalability

Data mining algorithms must be scalable to handle large datasets efficiently. As the size of the dataset increases, the time and computational resources required to perform data mining operations also increase. Moreover, the algorithms must be able to handle streaming data, which is generated continuously and must be processed in real-time.

1. Interpretability

Data mining algorithms can produce complex models which are difficult to interpret. This is because the algorithms use a combination of statistical and mathematical techniques to identify patterns and relationships in the data.

1. Ethics

Data mining also raises ethical concerns related to the collection, use and dissemination of data. The data may be used to discriminate against certain groups, violate piracy rights, or perpetuate existing biases.

1. Text Mining

Text mining is an automatic process that uses natural language processing (NLP) to extract valuable insights from unstructured text. By transforming data into information that machines can understand, text mining automates the process of classifying texts by sentiment, topic and intent. Text mining can be used as a preprocessing step for data mining or as a standalone process for specific tasks.

1. DBMS vis-à-vis data mining

DBMS and data mining answer difference queries. Data mining helps in predicting future whereas DBMS gives reports.

Examples of DBMS reports:

* Last month’s sales for each service type
* Sales per service grouped by customer sex or age bracket
* List of customers who lapsed their policy

Questions answered using Data Mining:

* What characteristics do customers who lapse their policies have in common and how do they differ from customers who renew their policies.
* Which motor insurance policy holders would be potential customers for my House Consent Insurance policy.

1. Scalable methods for mining sequential patterns

Module 2: Classification and Prediction

1. Frequent item sets

Frequent item-sets are collections of items that appear together in a dataset with a frequency above a specific threshold. They are a fundamental concept in association rule mining, a technique used to uncover interesting relationships, patterns or associations among a set of large databases.

1. Apriori Algorithm for frequent itemset generation.

Apriori algorithm is an influential algorithm for mining frequent item sets for Boolean association rules.

Key Concepts ->

* Frequent Item Sets – Collection of items that appear together in a dataset with a frequency above a specific threshold.
* Apriori property – Any subset of a frequent item-set must be frequent.
* Join Operation – To find Lk, a set of candidate k-item-sets is generated by join L(k-1) by itself.

Algorithm ->

* 1. Let k=1
  2. Generate frequent item-sets of length 1
  3. Repeat until no new frequent item-sets are identified:
     1. Generate length (k+1) candidate item-sets from length k frequency item-sets.
     2. Prune candidate item-sets containing subsets of length k that are infrequent.
     3. Count the support of each candidate by scanning the database
     4. Eliminate candidates that are infrequent leaving only those that are frequent.
  4. Generate strong rules (based on confidence values)

1. Dynamic Itemset counting algorithm.

Dynamic Itemset Counting algorithm is an alternative for Apriori itemset generation. Here, itemsets are dynamically added and deleted as transactions are read and it relies on the fact that for an Itemset to be frequent, all of its subsets must also be frequent, so we will only examine those itemsets whose subsets are frequent.

Algorithm stops after M transactions to add more itemsets.

Itemsets are marked in four different ways as they are counted ->

* Solid Box: Confirmed frequent itemset
* Solid Circle: Confirmed frequent itemset
* Dashed box: Suspected frequent itemset
* Dashed circle: Suspected frequent itemset

DIC Algorithm ->

1. Mark the empty itemset with a solid square. Mark all the 1-itemsets with dashed circle. Leave all other itemsets unmarked.
2. While any dashed itemsets remain:
   * 1. Read M transactions. For each transaction, increment the counters for the itemsets that appear in the transaction and are marked with dashes.
     2. If the dashed circle’s count exceeds the minsupp (minimum support), turn it into a dashed square.

If any immediate superset of it has all of its subsets as solid or dashed squares, add a new counter for it and make it a dashed circle.

* + 1. Once a dashed itemset has been counted through all the transactions, make it solid and stop counting it.

1. Clustering. Why is it required.

Cluster analysis, also known as clustering, is a data mining method which groups similar data points together.

The goal of cluster analysis is to divide a dataset into groups (clusters) such that the data points within each group are more similar to each other than to data points in other groups.

This process is often used for exploratory data analysis and can help identify patterns or relationships within the data which may not be obvious.

Different clustering algorithms include: k-means, hierarchical clustering, density-based clustering.

Why do we need clustering / Advantages ->

1. It can help identify patterns and relationships within a dataset which may not be immediately obvious.
2. It can used in exploratory analysis and help with feature selection.
3. It can be used to reduce dimensionality of the data.
4. It can be used for anomaly detection and outlier identification.

Disadvantages ->

1. It can be sensitive to the choice of initial conditions and the number of clusters.
2. It can be sensitive to the presence of noise or outlier in the data.
3. It can be computationally expensive for large dataset.
4. The result of the analysis can be affected ny the choice of clustering approach used.

Applications ->

1. Widely used in image processing, data analysis, pattern recognition
2. Helps marketers to find distinct groups in their customer base.
3. Helps in information discovery by classifying documents on the web.

Types of clustering ->

1. **Partition clustering:**

It is used to make partitions on the data in order to form clusters. If ‘n’ partitions are done on ‘p’ objects of the database then each partition is represented by a cluster and n<p. Only two conditions that need to be satisfied are:

* One objective should belong to only one group.
* There should be no group without even a single purpose.

In the partitioning method, there is a technique called iterative relocation, which means an object will be moved from one group to another to improve the partitioning.

1. **Hierarchical Clustering:**

In this method, a hierarchical decomposition of the given set of data objects is created. It can be further classified into two approaches:

* **Agglomerative approach [bottom-up]:** The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects known as **dendrogram**.
* **Divisive approach [top-down]:** The divisive clustering works just the opposite of agglomerative clustering. It starts by considering all data points into a big single cluster and later on splitting them into smaller heterogeneous clusters continuously until all data points are in their own cluster. Thus, they are good at identifying large clusters. It’s more efficient than agglomerative clustering.

1. Metadata and metadata catalog. Categories of metadata
2. Difference between Partition clustering and hierarchical clustering

Check Q4

1. Dimensional modelling

Data in a warehouse are usually multidimensional having **measure** and **dimension** attributes.

‘Measure attributes’ measure some values and can be aggregated upon those values [sum, avg ...].

‘Dimension attributes’ define the dimensions on which the measure attributes and their summary functions work upon.

The main goal of dimensional modelling is to improve the data retrieval so it is optimized for SELECT operation. The advantage of using this model is that we can store data in such a way that it is easier to store and retrieve the data once stored in a warehouse.

Elements of dimensional data modelling ->

* Facts – Facts are the measurable data elements that represent the business metrics of interest. For example, in a sales data warehouse, the facts might include sales revenue, units sold, and profit margins.
* Dimension – Dimensions are the descriptive data elements that are used to categorize or classify the data. E.g. in a sales data warehouse, the dimensions might include product, customer, time and location.
* Attributes – Characteristics of dimensions in data modelling are known as attributes. Used to filter, search facts. E.g. for a dimension of location, attributes can be state, country, zip code.
* Fact Table – The fact table is the central table that contains the measures or metrics of interest, surrounded by dimension tables that describe the attributes of the measures.
* Dimension Table – Dimensions of a fact are mentioned by the dimension table and they are basically joined by a foreign key.

1. K-means [A PARTITIOINING ALGORITHM]

K-means clustering is an unsupervised learning algorithm. There is no labeled data for this clustering, unlike in supervised learning. K-means performs the division of objects into clusters that share similarities and are dissimilar to the objects belonging to another cluster [Intra-cluster similarity is high and inter-cluster similarity is low].

K-means takes the i/p parameter k and partitions a set of n-objects into k-clusters so that the resulting intra-cluster similarity is high and inter-cluster similarity is low.

Algorithm ->

Input – k: no of clusters, D: dataset containing n objects [x1, x2, x3 … xn].

Output – A set of k clusters.

1. Step 1: Choose k random points as initial cluster centers called centroids.
2. Step 2: Assign each x(i) to the closest cluster by implementing its distance to each centroid.
3. Step 3: For each cluster, new centroids are computed by taking the average of the assigned points (updating the cluster means).
4. Step 4: Keep repeating steps 2 and 3 until convergence is achieved.
5. Linkage methods used in Hierarchical clustering

Hierarchical clustering is a powerful unsupervised learning technique used to group similar observations together based on their distance or similarity measures. The linkage method used in HC determines how the distance between clusters is calculated.

Different types of linkages are ->

* **Single Linkage** – Also known as nearest neighbor linkage, determines the distance between two clusters as the shortest distance between any two points in the two clusters.

In other words, the distance between two clusters is denoted by the distance between the closest points in the two clusters.

* **Complete Linkage** – Also known as farthest neighbor linkage, determines the distance between two clusters as the longest distance between any two points in two clusters.

In other words, the distance between two clusters is denoted by the distance between the closest points in the two clusters.

* **Average Linkage** – Determines the distance between two clusters as the average distance between all pairs of points in the two clusters.
* **Ward Linkage** – Also known as minimum variance linkage, determines the distance between two clusters by minimizing the increase in variance when the two clusters are merged.

1. K-medoids clustering algorithm [A PARTITIONING ALGORITHM]

K-Medoids clustering is a partition-based unsupervised clustering algorithm (with unlabeled data to be clustered) like K-means. It is an improvised version of K-means algorithm mainly designed to deal with outlier data sensitivity. Compared to other partitioning algorithms, k-medoids is simple, fast and easy to implement.

A medoid is a point in the cluster from which the sum of distances to other data points in that cluster is minimal.

K-medoids use medoids as reference points instead of centroids in case of k-means.

There are 3 types of k-medoids clustering algorithms ->

1. PAM [Partition Around Medoids]

This algorithm is intended to find a sequence of points called medoids which are centrally located in clusters. Objects that are tentatively defined as medoids are placed into a set S of selected objects.

If O is the set of total objects, then U=O-S is the set of unselected objects.

The goal of this algorithm is to minimize the average dissimilarity of objects to their closest selected objects.

The algorithm has two phases:

* BUILD – A collection of k objects is selected for an initial set S.
* SWAP – We try to improve the quality of the clustering by exchanging selected objects with unselected objects.

1. CLARA [Clustering Large Applications]

CLARA clustering algorithm is a type of k-medoids clustering method based on sampling. Only a small area of the real data is chosen as a representative of the data and medoids are chosen from this sample utilizing PAM.

The idea is that if the sample is selected in a fairly random manner, then it correctly represents the whole dataset and therefore, the representative objects(medoids) chosen will be similar as if chosen from the whole dataset.

CLARA can deal with a larger dataset than PAM.

1. CLARANS [Randomized Clustering Large Applications]

CLARANS algorithm combines both PAM and CLARA by searching only the subset of the dataset and it does not constraint itself to some sample at any given time.

While CLARA has a constant sample at each phase of the search, CLARANS draws a sample with some randomness at every phase of the search.

Module 3: Mining Time Series Data

* + - 1. Decision Tree
      2. Advantages and Disadvantages of decision tree method
      3. Principle of tree construction
      4. Classification
      5. Difference between supervised and unsupervised classification
      6. Regression
      7. Disadvantages of linear regression
      8. Define time series data mining
      9. How is time series data used for pattern analysis. Pearson’s formula
      10. Bayesian classification
      11. Uses of train data set and test data set in decision tree method
      12. Periodicity analysis for time related sequence data