What is dimensional modelling?

Dimensional Modeling (DM) is a data structure technique optimized for data storage in a Data warehouse. The purpose of dimensional modeling is to optimize the database for faster retrieval of data. The concept of Dimensional Modelling was developed by Ralph Kimball and consists of “fact” and “dimension” tables.

A dimensional model in data warehouse is designed to read, summarize, analyze numeric information like values, balances, counts, weights, etc. in a data warehouse. In contrast, relation models are optimized for addition, updating and deletion of data in a real-time Online Transaction System.

OLAP vs OLTP

Parameters	**OLTP**	**OLAP**
Process	It is an online transactional system. It manages database modification.	OLAP is an online analysis and data retrieving process.
Characteristic	It is characterized by large numbers of short online transactions.	It is characterized by a large volume of data.
Functionality	OLTP is an online database modifying system.	OLAP is an online database query management system.
Method	OLTP uses traditional DBMS.	OLAP uses the data warehouse.
Query	Insert, Update, and Delete information from the database.	Mostly select operations
Table	Tables in OLTP database are normalized.	Tables in OLAP database are **not** normalized.
Source	OLTP and its transactions are the sources of data.	Different OLTP databases become the source of data for OLAP.
Data Integrity	OLTP database must maintain data integrity constraint.	OLAP database does not get frequently modified. Hence, data integrity is not an issue.
Response time	It's response time is in millisecond.	Response time in seconds to minutes.
Data quality	The data in the OLTP database is always detailed and organized.	The data in OLAP process might not be organized.
Usefulness	It helps to control and run fundamental business tasks.	It helps with planning, problem-solving, and decision support.

Parameters	OLTP	OLAP
Operation	Allow read/write operations.	Only read and rarely write.
Audience	It is a market orientated process.	It is a customer orientated process.
Query Type	Queries in this process are standardized and simple.	Complex queries involving aggregations.
Back-up	Complete backup of the data combined with incremental backups.	OLAP only need a backup from time to time. Backup is not important compared to OLTP
Design	DB design is application oriented. Example: Database design changes with industry like Retail, Airline, Banking, etc.	DB design is subject oriented. Example: Database design changes with subjects like sales, marketing, purchasing, etc.
User type	It is used by Data critical users like clerk, DBA & Data Base professionals.	Used by Data knowledge users like workers, managers, and CEO.
Purpose	Designed for real time business operations.	Designed for analysis of business measures by category and attributes.
Performance metric	Transaction throughput is the performance metric	Query throughput is the performance metric.
Number of users	This kind of Database users allows thousands of users.	This kind of Database allows only hundreds of users.
Productivity	It helps to Increase user's self-service and productivity	Help to Increase productivity of the business analysts.
Challenge	Data Warehouses historically have been a development project which may prove costly to build.	An OLAP cube is not an open SQL server data warehouse. Therefore, technical knowledge and experience is essential to manage the OLAP server.
Process	It provides fast result for daily used data.	It ensures that response to the query is quicker consistently.
Characteristic	It is easy to create and maintain.	It lets the user create a view with the help of a spreadsheet.
Style	OLTP is designed to have fast response time, low data redundancy and is normalized.	A data warehouse is created uniquely so that it can integrate different data sources for building a consolidated database

What is apriori? What is it used for? What is support formula?

Apriori algorithm is a classical **algorithm** in data mining. It is used for mining frequent itemsets and relevant association rules. It is devised to **operate** on a database containing a lot of transactions, for instance, items brought by customers in a store.

$$\text{Support } (A) = \frac{\text{Number of transaction in which A appears}}{\text{Total number of transactions}}$$

$$\text{Confidence } (A \longrightarrow B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A)}$$

Apriori says:

The probability that item I is not frequent is if:

- P(I) < minimum support threshold, then I is not frequent.
- P (I+A) < minimum support threshold, then I+A is not frequent, where A also belongs to itemset.
- If an itemset set has value less than minimum support then all of its supersets will also fall below min support, and thus can be ignored. This property is called the Antimonotone property.

The steps followed in the Apriori Algorithm of data mining are:

1. **Join Step**: This step generates (K+1) itemset from K-itemsets by joining each item with itself.
2. **Prune Step**: This step scans the count of each item in the database. If the candidate item does not meet minimum support, then it is regarded as infrequent and thus it is removed. This step is performed to reduce the size of the candidate itemsets.

Steps In Apriori

Apriori algorithm is a sequence of steps to be followed to find the most frequent itemset in the given database. This data mining technique follows the join and the prune steps iteratively until the most frequent itemset is achieved. A minimum support threshold is given in the problem or it is assumed by the user.

1) In the first iteration of the algorithm, each item is taken as a 1-itemsets candidate. The algorithm will count the occurrences of each item.

2) Let there be some minimum support, min_sup (eg 2). The set of 1 – itemsets whose occurrence is satisfying the min sup are determined. Only those candidates which count more than or equal to min_sup, are taken ahead for the next iteration and the others are pruned.

3) Next, 2-itemset frequent items with min_sup are discovered. For this in the join step, the 2-itemset is generated by forming a group of 2 by combining items with itself.

4) The 2-itemset candidates are pruned using min-sup threshold value. Now the table will have 2 –itemsets with min-sup only.

5) The next iteration will form 3 –itemsets using join and prune step. This iteration will follow antimonotone property where the subsets of 3-itemsets, that is the 2 –itemset subsets of each group fall in min_sup. If all 2-itemset subsets are frequent then the superset will be frequent otherwise it is pruned.

6) Next step will follow making 4-itemset by joining 3-itemset with itself and pruning if its subset does not meet the min_sup criteria. The algorithm is stopped when the most frequent itemset is achieved.

What is data cleaning? What are the steps of data cleaning?

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset.

When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct. There is no one absolute way to prescribe the exact steps in the data cleaning process because the processes will vary from dataset to dataset. But it is crucial to establish a template for your data cleaning process so you know you are doing it the right way every time.

How do you clean data?

1. **Step** 1: Remove duplicate or irrelevant observations. Remove unwanted observations from your dataset, including duplicate observations or irrelevant observations. ...
2. **Step** 2: Fix structural errors. ...
3. **Step** 3: Filter unwanted outliers. ...
4. **Step** 4: Handle missing **data**. ...
5. **Step** 4: Validate and QA.

Step 1: Remove duplicate or irrelevant observations

Remove unwanted observations from your dataset, including duplicate observations or irrelevant observations. Duplicate observations will happen most often during data collection. When you combine data sets from multiple places, scrape data, or receive data from clients or multiple departments, there are opportunities to create duplicate data. De-duplication is one of the largest areas to be considered in this process.

Irrelevant observations are when you notice observations that do not fit into the specific problem you are trying to analyze. For example, if you want to analyze data regarding millennial customers, but your dataset includes older generations, you might remove those irrelevant observations. This can make analysis more efficient and minimize distraction from your primary target—as well as creating a more manageable and more performant dataset.

Step 2: Fix structural errors

Structural errors are when you measure or transfer data and notice strange naming conventions, typos, or incorrect capitalization. These inconsistencies can cause mislabeled categories or classes. For example, you may find “N/A” and “Not Applicable” both appear, but they should be analyzed as the same category.

Step 3: Filter unwanted outliers

Often, there will be one-off observations where, at a glance, they do not appear to fit within the data you are analyzing. If you have a legitimate reason to remove an outlier, like improper data-entry, doing so will help the performance of the data you are working with. However, sometimes it is the appearance of an outlier that will prove a theory you are working on.

Remember: just because an outlier exists, doesn’t mean it is incorrect. This step is needed to determine the validity of that number. If an outlier proves to be irrelevant for analysis or is a mistake, consider removing it.

Step 4: Handle missing data

You can’t ignore missing data because many algorithms will not accept missing values. There are a couple of ways to deal with missing data. Neither is optimal, but both can be considered.

1. As a first option, you can drop observations that have missing values, but doing this will drop or lose information, so be mindful of this before you remove it.
2. As a second option, you can input missing values based on other observations; again, there is an opportunity to lose integrity of the data because you may be operating from assumptions and not actual observations.
3. As a third option, you might alter the way the data is used to effectively navigate null values.

Step 4: Validate and QA

At the end of the data cleaning process, you should be able to answer these questions as a part of basic validation:

- Does the data make sense?
- Does the data follow the appropriate rules for its field?
- Does it prove or disprove your working theory, or bring any insight to light?
- Can you find trends in the data to help you form your next theory?
- If not, is that because of a data quality issue?

False conclusions because of incorrect or “dirty” data can inform poor business strategy and decision-making. False conclusions can lead to an embarrassing moment in a reporting meeting when you realize your data doesn’t stand up to scrutiny.

Before you get there, it is important to create a culture of quality data in your organization. To do this, you should document the tools you might use to create this culture and what data quality means to you.

Benefits of data cleaning

Having clean data will ultimately increase overall productivity and allow for the highest quality information in your decision-making. Benefits include:

- Removal of errors when multiple sources of data are at play.
- Fewer errors make for happier clients and less-frustrated employees.
- Ability to map the different functions and what your data is intended to do.
- Monitoring errors and better reporting to see where errors are coming from, making it easier to fix incorrect or corrupt data for future applications.
- Using tools for data cleaning will make for more efficient business practices and quicker decision-MAKING

What can be used to increase efficiency of apriori?

Optimal pruning:

When production of the subsequent K dimension subsets on the basis of (K - 1) dimensions frequent itemsets, we can use a temporary table to count the iteration of each item of frequent patterns. In case the iteration number of an item is less than K -1 it cannot be used to produce the next K items. Therefore, all repeated patterns containing that item can be deleted. This will result in a reduction in the sub-sets and the production of repeated patterns.

Methods To Improve Apriori Efficiency

Many methods are available for improving the efficiency of the algorithm.

1. **Hash-Based Technique:** This method uses a hash-based structure called a hash table for generating the k-itemsets and its corresponding count. It uses a hash function for generating the table.
2. **Transaction Reduction:** This method reduces the number of transactions scanning in iterations. The transactions which do not contain frequent items are marked or removed.
3. **Partitioning:** This method requires only two database scans to mine the frequent itemsets. It says that for any itemset to be potentially frequent in the database, it should be frequent in at least one of the partitions of the database.
4. **Sampling:** This method picks a random sample S from Database D and then searches for frequent itemset in S. It may be possible to lose a global frequent itemset. This can be reduced by lowering the min_sup.

5. **Dynamic Itemset Counting:** This technique can add new candidate itemsets at any marked start point of the database during the scanning of the database.
6.

Applications Of Apriori Algorithm

Some fields where Apriori is used:

1. **In Education Field:** Extracting association rules in data mining of admitted students through characteristics and specialties.
2. **In the Medical field:** For example Analysis of the patient's database.
3. **In Forestry:** Analysis of probability and intensity of forest fire with the forest fire data.
4. Apriori is used by many companies like Amazon in the **Recommender System** and by Google for the auto-complete feature.

Reducing the length of transactions:

When the database is scanned for generating K +1-dimension frequent itemset, calculated the length of the transaction, and all transactions are removed directly from the subsequent K + 1-dimension and are removed for the remaining transactions of the sk - 1 - sk items and are again calculated. The sk contains the set of the next K items of frequent patterns when the database is scanned for the production of K + 1 sets. The items containing sk - 1 - sk do not require scanning them Therefore, first with the above technique to produce sub - sets of K-dimension, the transactions are pressed and the volume of the information bank is reduced.

Improved algorithm:

Using the above two strategies and applying it to the A - Priori algorithm, efficiency can be increased by the algorithm..

The improved version follows:

```
1)  L1={1-dimensional frequent itemsets}
2)  For(k=2;Lk-1 !=Null;k++){
3)       If (k!=2){
4)          Count the frequency named |LK-1(a)| of every item named a exists in Lk-1.
5)       If (|Lk-1(a)| < k-1) delete all the itemsets contained a in |Lk-1| and update LK-1.
6)                  }
7)      For  each  l1 € Lk-1,for each l2 € LK-1
8)      {
9)           If ((l1[1]=l2[1] ^ … ^ (l1[k-2]=l2[k-2] ^ (l1[k-1]<l2[k-1]))
10)             { c=l1 ∞ l2;
11)              For each  (k-1)  subsets of c{
```

```
1)  L1={1-dimensional frequent itemsets}
2)  For(k=2;Lk-1 !=Null;k++){
3)       If (k!=2){
4)          Count the frequency named |LK-1(a)| of every item named a exists in Lk-1.
5)       If (|Lk-1(a)| < k-1) delete all the itemsets contained a in |Lk-1| and update LK-1.
6)                  }
7)      For  each  l1 € Lk-1,for each l2 € LK-1
8)      {
9)           If ((l1[1]=l2[1] ^ … ^ (l1[k-2]=l2[k-2] ^ (l1[k-1]<l2[k-1]))
10)             { c=l1 ∞ l2;
11)              For each  (k-1)  subsets of c{
12)                   If  (s Ɇ Lk-1)
13)                     {delete c;}       Break;
14)                  }
15)                 Ck=Ck ∪{C};
16)                 For each t € D {
17)                  If  (Count[t]<k)      delete t;
18)                  Else {delete item of t including in (Sk-1 – Sk) and recalculate the length of t;
19)                 If   (Count[t]<k) delete     t;
20)                 Ct=subset(Ct,t);
21)                For each c € Ct    c.count ++;}
22)                }
23)              Lk={c € Ck |c.count>min_sup};
24)          Return L=L1 ∪L2 ∪ … ∪ Lk;}
```

Explain hash based technique in detail for apriori

Hash-Based Technique: This **method** uses a **hash-based** structure called a **hash** table for generating the k-itemsets and its corresponding count. It uses a **hash function** for generating the table. Transaction Reduction: This **method** reduces the number of transactions scanning in iterations.

What is a Star Schema?

Star Schema in data warehouse, in which the center of the star can have one fact table and a number of associated dimension tables. It is known as star schema as its structure resembles a star. The Star Schema data model is the simplest type of Data Warehouse schema. It is also known as Star Join Schema and is optimized for querying large data sets.

What is a factless fact table

Factless facts are those fact tables that have no measures associated with the transaction. Factless facts are a simple collection of dimensional keys which define the transactions or describing condition for the time period of the fact.

A factless fact table is a fact table that does not have any measures. It is essentially an intersection of dimensions. On the surface, a factless fact table does not make sense, since a fact table is, after all, about facts. However, there are situations where having this kind of relationship makes sense in data warehousing.

Drawbacks of apriori algorithm

The major **drawback** with **Apriori algorithm** is of time and space. It generates numerous uninteresting itemsets which lead to generate various rules which are of completely of no use. The two factors considered for association rules generation are Minimum Support Threshold and Minimum Confidence Threshold.

Apriori

• Apriori Algorithm is an influential algorithm for mining frequent itemsets for boolean association rules.

• It uses **prior knowledge** of frequent itemset properties (Aprior).

• It uses K frequent itemsets to find K+1 itemsets.

• It is based on three concept: **Frequent itemset**, **Apriori property and join operations**

Advantages :

• Easy to understand and implement.

• Can be easily parallelized.

• Uses large itemset property.

Disadvantages:

• Requires many database scans.

• Assumes transaction database is memory resident.

Explain Nbayes

Naïve Bayes Classifier Algorithm

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.
- It is mainly used in *text classification* that includes a high-dimensional training dataset.
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.
- **It is a probabilistic classifier, which means it predicts on the basis of the probability of an object**.

- Some popular examples of Naïve Bayes Algorithm are **spam filtration, Sentimental analysis, and classifying articles**.

Why is it called Naïve Bayes?

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

- **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem

Bayes' Theorem:

- Bayes' theorem is also known as **Bayes' Rule** or **Bayes' law**, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.
- The formula for Bayes' theorem is given as:

$$P(A|B)= \frac{P(B|A)P(A)}{P(B)}$$

Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

P(A) is Prior Probability: Probability of hypothesis before observing the evidence.

P(B) is Marginal Probability: Probability of Evidence.

What are aggregate functions? (Sum, Average, Min, Max, Count) Which aggregate function works for characters? (I answered COUNT)

Difference between K means and K medoids?

Parameters	k-means	k-medoids
Complexity	O (i k n)	O (i k (n-k)2)
Efficiency	Comparatively more	Comparatively less
Implementation	Easy	Complicated
Sensitive to Outliers?	Yes	No
Advance specification of No. of clusters 'k'	Required	Required
Does initial partition affects result and Runtime?	yes	yes
Optimized for	Separated clusters	Separated clusters, small dataset

What is concept hierarchy?

A **concept hierarchy** defines a sequence of mappings from a set of low-level **concepts** to higher-level, more general **concepts**. ... These mappings form a **concept hierarchy** for the dimension location, mapping a set of low-level **concepts** (i.e., cities) to higher-level, more general **concepts** (i.e., countries).

Difference between K mean and K medoid?

Where are Apriori algorithms used?

The **Apriori algorithm** is **used** for mining frequent itemsets and devising association rules from a transactional database. The parameters "support" and "confidence" are **used**. Support refers to items' frequency of occurrence; confidence is a conditional probability. Items in a transaction form an item set.

1. **In Education Field:** Extracting association rules in data mining of admitted students through characteristics and specialties.
2. **In the Medical field:** For example Analysis of the patient's database.
3. **In Forestry:** Analysis of probability and intensity of forest fire with the forest fire data.
4. Apriori is used by many companies like Amazon in the **Recommender System** and by Google for the auto-complete feature.