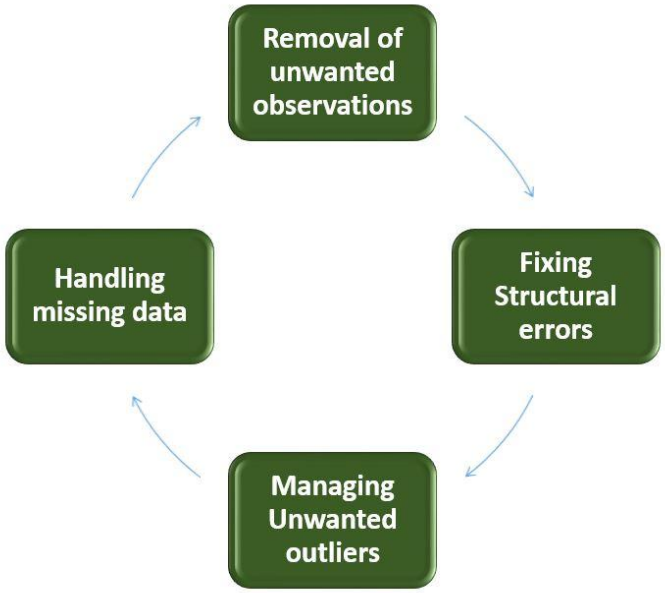
What is data cleaning?

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.

For Example:

Data cleaning is correcting errors or inconsistencies, or restructuring data to make it easier to use. This includes things like standardizing dates and addresses, making sure field values (e.g., “Closed won” and “Closed Won”) match, parsing area codes out of phone numbers, and flattening nested data structures.

Data cleaning is correcting errors or inconsistencies, or restructuring data to make it easier to use. This includes things like standardizing dates and addresses, making sure field values (e.g., “Closed won” and “Closed Won”) match, parsing area codes out of phone numbers, and flattening nested data structures.



- **Removal of Unwanted Observations**: Identify and eliminate irrelevant or redundant observations from the dataset. The step involves scrutinizing data entries for duplicate records, irrelevant information, or data points that do not contribute meaningfully to the analysis. Removing unwanted observations streamlines the dataset, reducing noise and improving the overall quality.
- **Fixing Structure errors:** Address structural issues in the dataset, such as inconsistencies in data formats, naming conventions, or variable types. Standardize formats, correct naming discrepancies, and ensure uniformity in data representation.

Fixing structure errors enhances data consistency and facilitates accurate analysis and interpretation.

- **Managing Unwanted outliers:** Identify and manage outliers, which are data points significantly deviating from the norm. Depending on the context, decide whether to remove outliers or transform them to minimize their impact on analysis. Managing outliers is crucial for obtaining more accurate and reliable insights from the data.
- **Handling Missing Data:** Devise strategies to handle missing data effectively. This may involve imputing missing values based on statistical methods, removing records with missing values, or employing advanced imputation techniques. Handling missing data ensures a more complete dataset, preventing biases and maintaining the integrity of analyses.

Data Integration in Data Mining

INTRODUCTION :

- Data integration in data mining refers to the process of combining data from multiple sources into a single, unified view. This can involve cleaning and transforming the data, as well as resolving any inconsistencies or conflicts that may exist between the different sources. The goal of data integration is to make the data more useful and meaningful for the purposes of analysis and decision making. Techniques used in data integration include data warehousing, ETL (extract, transform, load) processes, and data federation.

Data Integration is a data preprocessing technique that combines data from multiple heterogeneous data sources into a coherent data store and provides a unified view of the data. These sources may include multiple data cubes, databases, or flat files.

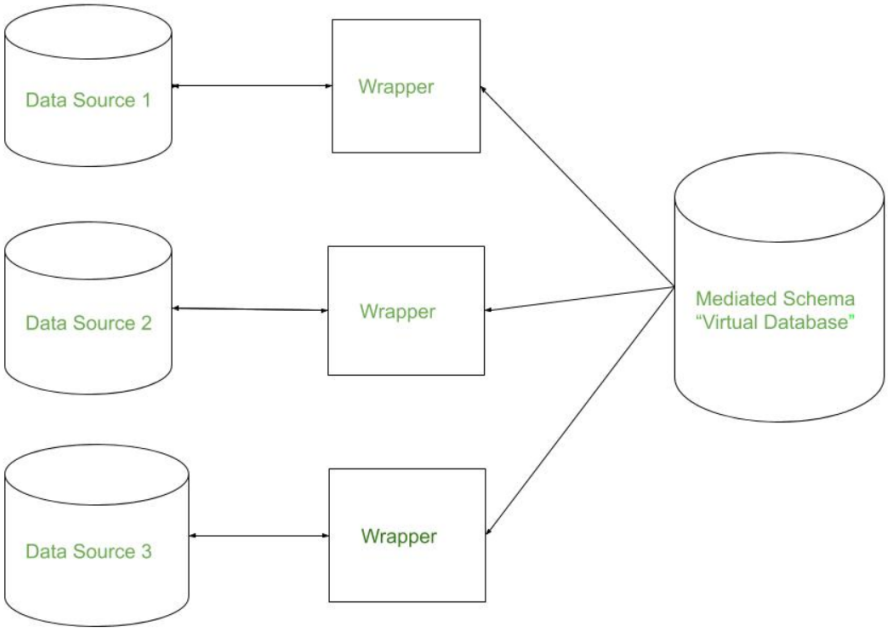
The data integration approaches are formally defined as triple <G, S, M> where,
G stand for the global schema,
S stands for the heterogeneous source of schema,
M stands for mapping between the queries of source and global schema.

What is data integration :

Data integration is the process of combining data from multiple sources into a cohesive and consistent view. This process involves identifying and accessing the different data sources, mapping the data to a common format, and reconciling any inconsistencies or discrepancies between the sources. The goal of data integration is to make it easier to access and analyze data that is spread across multiple systems or platforms, in order to gain a more complete and accurate understanding of the data.

Data integration can be challenging due to the variety of data formats, structures, and semantics used by different data sources. Different data sources may use different data types, naming conventions, and schemas, making it difficult to combine the data into a single view. Data integration typically involves a combination of manual and automated processes, including data profiling, data mapping, data transformation, and data reconciliation.

Data integration is used in a wide range of applications, such as business intelligence, data warehousing, master data management, and analytics. Data integration can be critical to the success of these applications, as it enables organizations to access and analyze data that is spread across different systems, departments, and lines of business, in order to make better decisions, improve operational efficiency, and gain a competitive advantage.



There are mainly 2 major approaches for data integration – one is the "tight coupling approach" and another is the "loose coupling approach".

Tight Coupling:
This approach involves creating a centralized repository or data warehouse to store the integrated data. The data is extracted from various sources, transformed and loaded into a data warehouse. Data is integrated in a tightly coupled manner, meaning that the data is integrated at a high level, such as at the level of the entire dataset or schema. This approach is also known as data warehousing, and it enables data consistency and integrity, but it can be inflexible and difficult to change or update.

- Here, a data warehouse is treated as an information retrieval component.
- In this coupling, data is combined from different sources into a single physical location through the process of ETL – Extraction, Transformation, and Loading.

Loose Coupling:
This approach involves integrating data at the lowest level, such as at the level of individual data elements or records. Data is integrated in a loosely coupled manner, meaning that the data is integrated at a low level, and it allows data to be integrated without having to create a central repository or data warehouse. This approach is also known as data federation, and it enables data flexibility and easy updates, but it can be difficult to maintain consistency and integrity across multiple data sources.

- Here, an interface is provided that takes the query from the user, transforms it in a way the source database can understand, and then sends the query directly to the source databases to obtain the result.
- And the data only remains in the actual source databases.

Issues in Data Integration:
There are several issues that can arise when integrating data from multiple sources, including:

1. **Data Quality:** Inconsistencies and errors in the data can make it difficult to combine and analyze.
2. **Data Semantics:** Different sources may use different terms or definitions for the same data, making it difficult to combine and understand the data.

3. **Data Heterogeneity:** Different sources may use different data formats, structures, or schemas, making it difficult to combine and analyze the data.
4. **Data Privacy and Security:** Protecting sensitive information and maintaining security can be difficult when integrating data from multiple sources.
5. **Scalability:** Integrating large amounts of data from multiple sources can be computationally expensive and time-consuming.
6. **Data Governance:** Managing and maintaining the integration of data from multiple sources can be difficult, especially when it comes to ensuring data accuracy, consistency, and timeliness.
7. **Performance:** Integrating data from multiple sources can also affect the performance of the system.
8. **Integration with existing systems:** Integrating new data sources with existing systems can be a complex task, requiring significant effort and resources.
9. **Complexity:** The complexity of integrating data from multiple sources can be high, requiring specialized skills and knowledge.

There are three issues to consider during data integration: Schema Integration, Redundancy Detection, and resolution of data value conflicts. These are explained in brief below.

1. Schema Integration:

- Integrate metadata from different sources.
- The real-world entities from multiple sources are referred to as the entity identification problem.ER

2. Redundancy Detection:

- An attribute may be redundant if it can be derived or obtained from another attribute or set of attributes.
- Inconsistencies in attributes can also cause redundancies in the resulting data set.
- Some redundancies can be detected by correlation analysis.

3. Resolution of data value **conflicts:**

- This is the third critical issue in data integration.
- Attribute values from different sources may differ for the same real-world entity.
- An attribute in one system may be recorded at a lower level of abstraction than the "same" attribute in another.

Data Reduction in Data Mining

Prerequisite – Data Mining

The method of data reduction may achieve a condensed description of the original data which is much smaller in quantity but keeps the quality of the original data.

INTRODUCTION:

Data reduction is a technique used in data mining to reduce the size of a dataset while still preserving the most important information. This can be beneficial in situations where the dataset is too large to be processed efficiently, or where the dataset contains a large amount of irrelevant or redundant information.

There are several different data reduction techniques that can be used in data mining, including:

1. **Data Sampling:** This technique involves selecting a subset of the data to work with, rather than using the entire dataset. This can be useful for reducing the size of a dataset while still preserving the overall trends and patterns in the data.
2. **Dimensionality Reduction:** This technique involves reducing the number of features in the dataset, either by removing features that are not relevant or by combining multiple features into a single feature.
3. **Data Compression:** This technique involves using techniques such as lossy or lossless compression to reduce the size of a dataset.
4. **Data Discretization:** This technique involves converting continuous data into discrete data by partitioning the range of possible values into intervals or bins.
5. **Feature Selection:** This technique involves selecting a subset of features from the dataset that are most relevant to the task at hand.
6. It's important to note that data reduction can have a trade-off between the accuracy and the size of the data. The more data is reduced, the less accurate the model will be and the less generalizable it will be.

In conclusion, data reduction is an important step in data mining, as it can help to improve the efficiency and performance of machine learning algorithms by reducing the size of the dataset. However, it is important to be aware of the trade-off between the size and accuracy of the data, and carefully assess the risks and benefits before implementing it.

Methods of data reduction:

These are explained as following below.

1. Data Cube Aggregation:

This technique is used to aggregate data in a simpler form. For example, imagine the information you gathered for your analysis for the years 2012 to 2014, that data includes the revenue of your company every three months. They involve you in the annual sales, rather than the quarterly average, So we can summarize the data in such a way that the resulting data summarizes the total sales per year instead of per quarter. It summarizes the data.

2. Dimension reduction:

Whenever we come across any data which is weakly important, then we use the attribute required for our analysis. It reduces data size as it eliminates outdated or redundant features.

- **Step-wise Forward Selection –**
  The selection begins with an empty set of attributes later on we decide the best of the original attributes on the set based on their relevance to other attributes. We know it as a p-value in statistics.
  Suppose there are the following attributes in the data set in which few attributes are redundant.

Initial attribute Set: {X1, X2, X3, X4, X5, X6}

Initial reduced attribute set: { }

Step-1: {X1}

Step-2: {X1, X2}

Step-3: {X1, X2, X5}

Final reduced attribute set: {X1, X2, X5}

- **Combination of forwarding and Backward Selection –**
  It allows us to remove the worst and select the best attributes, saving time and making the process faster.

3. Data Compression:

The data compression technique reduces the size of the files using different encoding mechanisms (Huffman Encoding & run-length Encoding). We can divide it into two types based on their compression techniques.

- **Lossless Compression –**
  Encoding techniques (Run Length Encoding) allow a simple and minimal data size reduction. Lossless data compression uses algorithms to restore the precise original data from the compressed data.
- **Lossy Compression –**
  Methods such as the Discrete Wavelet transform technique, PCA (principal component analysis) are examples of this compression. For e.g., the JPEG image format is a lossy compression, but we can find the meaning equivalent to the original image. In lossy-data compression, the decompressed data may differ from the original data but are useful enough to retrieve information from them.

4. Numerosity Reduction:

In this reduction technique, the actual data is replaced with mathematical models or smaller representations of the data instead of actual data, it is important to only store the model parameter. Or non-parametric methods such as clustering, histogram, and sampling.

5. Discretization & Concept Hierarchy Operation:

Techniques of data discretization are used to divide the attributes of the continuous nature into data with intervals. We replace many constant values of the attributes by labels of small intervals. This means that mining results are shown in a concise, and easily understandable way.

- **Top-down discretization –**
  If you first consider one or a couple of points (so-called breakpoints or split points) to divide the whole set of attributes and repeat this method up to the end, then the process is known as top-down discretization also known as splitting.
- **Bottom-up discretization –**
  If you first consider all the constant values as split points, some are discarded through a combination of the neighborhood values in the interval, that process is called bottom-up discretization.

Concept Hierarchies:

It reduces the data size by collecting and then replacing the low-level concepts (such as 43 for age) with high-level concepts (categorical variables such as middle age or Senior).
For numeric data following techniques can be followed:

- **Binning –**
  Binning is the process of changing numerical variables into categorical counterparts. The number of categorical counterparts depends on the number of bins specified by the user.
- **Histogram analysis –**
  Like the process of binning, the histogram is used to partition the value for the attribute X, into disjoint ranges called brackets. There are several partitioning rules:

1. **Equal Frequency partitioning:** Partitioning the values based on their number of occurrences in the data set.
2. **Equal Width Partitioning:** Partitioning the values in a fixed gap based on the number of bins i.e. a set of values ranging from 0-20.
3. **Clustering:** Grouping similar data together.

ADVANTAGED OR DISADVANTAGES OF Data Reduction in Data Mining :

Data reduction in data mining can have a number of advantages and disadvantages.

Advantages:

1. Improved efficiency: Data reduction can help to improve the efficiency of machine learning algorithms by reducing the size of the dataset. This can make it faster and more practical to work with large datasets.
2. Improved performance: Data reduction can help to improve the performance of machine learning algorithms by removing irrelevant or redundant information from the dataset. This can help to make the model more accurate and robust.
3. Reduced storage costs: Data reduction can help to reduce the storage costs associated with large datasets by reducing the size of the data.
4. Improved interpretability: Data reduction can help to improve the interpretability of the results by removing irrelevant or redundant information from the dataset.

Disadvantages:

1. Loss of information: Data reduction can result in a loss of information, if important data is removed during the reduction process.
2. Impact on accuracy: Data reduction can impact the accuracy of a model, as reducing the size of the dataset can also remove important information that is needed for accurate predictions.
3. Impact on interpretability: Data reduction can make it harder to interpret the results, as removing irrelevant or redundant information can also remove context that is needed to understand the results.
4. Additional computational costs: Data reduction can add additional computational costs to the data mining process, as it requires additional processing time to reduce the data.
5. In conclusion, data reduction can have both advantages and disadvantages. It can improve the efficiency and performance of machine learning algorithms by reducing the size of the dataset. However, it can also result in a loss of information, and make it harder to interpret the results. It’s important to weigh the pros and cons of data reduction and carefully assess the risks and benefits before implementing it.

Discretization in data mining

Data discretization refers to a method of converting a huge number of data values into smaller ones so that the evaluation and management of data become easy. In other words, data discretization is a method of converting attributes values of continuous data into a finite set of intervals with minimum data loss. There are two forms of data discretization first is

supervised discretization, and the second is unsupervised discretization. Supervised discretization refers to a method in which the class data is used. Unsupervised discretization refers to a method depending upon the way which operation proceeds. It means it works on the top-down splitting strategy and bottom-up merging strategy.

Now, we can understand this concept with the help of an example

Suppose we have an attribute of Age with the given values

Age	1,5,9,4,7,11,14,17,13,18, 19,31,33,36,42,44,46,70,74,78,77

Table before Discretization

Attribute	Age	Age	Age	Age
	1,5,4,9,7	11,14,17,13,18,19	31,33,36,42,44,46	70,74,77,78
After Discretization	Child	Young	Mature	Old

Another example is analytics, where we gather the static data of website visitors. For example, all visitors who visit the site with the IP address of India are shown under country level.

Some Famous techniques of data discretization

Histogram analysis

Histogram refers to a plot used to represent the underlying frequency distribution of a continuous data set. Histogram assists the data inspection for data distribution. For example, Outliers, skewness representation, normal distribution representation, etc.

Binning

Binning refers to a data smoothing technique that helps to group a huge number of continuous values into smaller values. For data discretization and the development of idea hierarchy, this technique can also be used.

Cluster Analysis

Cluster analysis is a form of data discretization. A clustering algorithm is executed by dividing the values of x numbers into clusters to isolate a computational feature of x.

Data discretization using decision tree analysis

Data discretization refers to a decision tree analysis in which a top-down slicing technique is used. It is done through a supervised procedure. In a numeric attribute discretization, first, you need to select the attribute that has the least entropy, and then you need to run it with the

help of a recursive process. The recursive process divides it into various discretized disjoint intervals, from top to bottom, using the same splitting criterion.

Data discretization using correlation analysis

Discretizing data by linear regression technique, you can get the best neighboring interval, and then the large intervals are combined to develop a larger overlap to form the final 20 overlapping intervals. It is a supervised procedure.

Data discretization and concept hierarchy generation

The term hierarchy represents an organizational structure or mapping in which items are ranked according to their levels of importance. In other words, we can say that a hierarchy concept refers to a sequence of mappings with a set of more general concepts to complex concepts. It means mapping is done from low-level concepts to high-level concepts. For example, in computer science, there are different types of hierarchical systems. A document is placed in a folder in windows at a specific place in the tree structure is the best example of a computer hierarchical tree model. There are two types of hierarchy: top-down mapping and the second one is bottom-up mapping.

Let's understand this concept hierarchy for the dimension location with the help of an example.

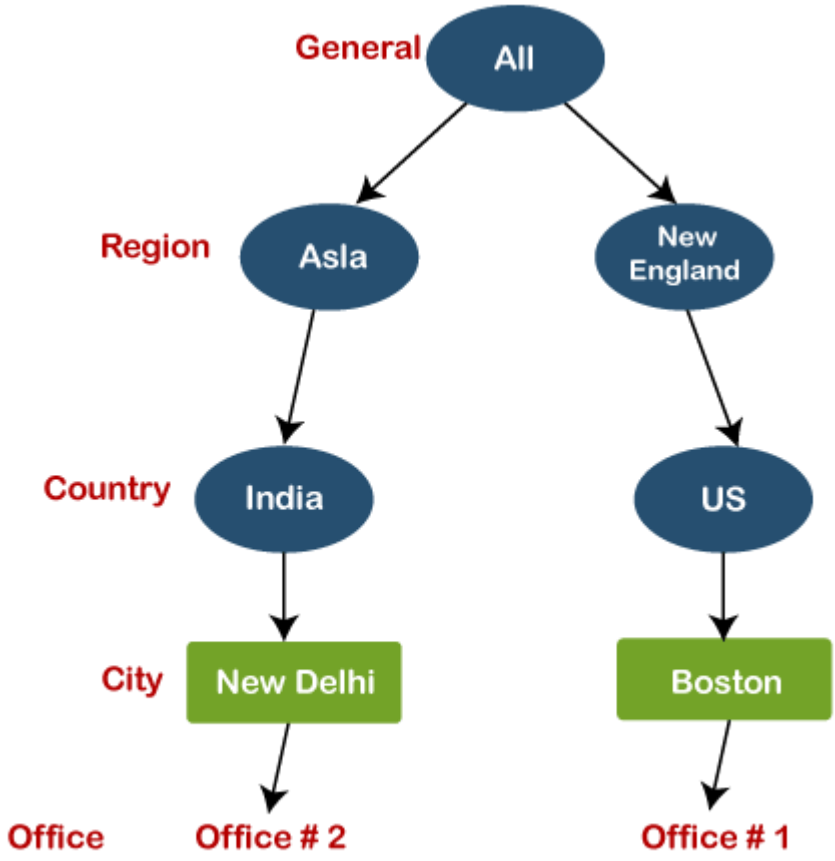
A particular city can map with the belonging country. For example, New Delhi can be mapped to India, and India can be mapped to Asia.

Top-down mapping

Top-down mapping generally starts with the top with some general information and ends with the bottom to the specialized information.

Bottom-up mapping

Bottom-up mapping generally starts with the bottom with some specialized information and ends with the top to the generalized information.



Concept Hierarchy Generation

Data discretization and binarization in data mining

Data discretization is a method of converting attributes values of continuous data into a finite set of intervals with minimum data loss. In contrast, data binarization is used to transform the continuous and discrete attributes into binary attributes.

Why is Discretization important?

As we know, an infinite of degrees of freedom mathematical problem poses with the continuous data. For many purposes, data scientists need the implementation of discretization. It is also used to improve signal noise ratio.

Attribute-Oriented Induction

The Attribute-Oriented Induction (AOI) approach to data generalization and summarization – based characterization was first proposed in 1989 (KDD ‘89 workshop) a few years before the introduction of the data cube approach.

The data cube approach can be considered as a data warehouse – based, pre computational – oriented, materialized approach.

It performs off-line aggregation before an OLAP or data mining query is submitted for processing.

On the other hand, the attribute oriented induction approach, at least in its initial proposal, a relational database query – oriented, generalized – based, on-line data analysis technique.

However, there is no inherent barrier distinguishing the two approaches based on online aggregation versus offline precomputation.

Some aggregations in the data cube can be computed on-line, while off-line precomputation of multidimensional space can speed up attribute-oriented induction as well.

It was proposed in 1989 (KDD '89 workshop).

It is not confined to categorical data nor particular measures.

How it is done?

- Collect the task-relevant data(initial relation) using a relational database query
- Perform generalization by attribute removal or attribute generalization.
- Apply aggregation by merging identical, generalized tuples and accumulating their respective counts.
- Reduces the size of the generalized data set.
- Interactive presentation with users.

Basic Principles Of Attribute Oriented Induction

Data focusing:

- Analyzing task-relevant data, including dimensions, and the result is the initial relation.

Attribute-removal:

- To remove attribute A if there is a large set of distinct values for A but (1) there is no generalization operator on A, or (2) A's higher-level concepts are expressed in terms of other attributes.

Attribute-generalization:

- If there is a large set of distinct values for A, and there exists a set of generalization operators on A, then select an operator and generalize A.

Attribute-threshold control:

- Typical 2-8, specified/default.

Generalized relation threshold control (10-30):

- To control the final relation/rule size.

Algorithm for Attribute Oriented Induction

InitialRel:

- It is nothing but query processing of task-relevant data and deriving the initial relation.

PreGen:

- It is based on the analysis of the number of distinct values in each attribute and to determine the generalization plan for each attribute: removal? or how high to generalize?

PrimeGen:

- It is based on the PreGen plan and performing the generalization to the right level to derive a “prime generalized relation” and also accumulating the counts.

Presentation:

- User interaction: (1) adjust levels by drilling, (2) pivoting, (3) mapping into rules, cross tabs, visualization presentations.

Example

Let's say there is a University database that is to be characterized, for that its corresponding DMQL will be

use University_DB
mine characteristics as “Science_Students”
in relevance to name, gender, major, birth_place, birth_date, residence, phone_no, GPA
from student

Its corresponding SQL statement can be:

Select name, gender, major, birth_place, birth_date, residence, phone_no, GPA
from student
where status in {“Msc”, “MBA”, “Ph.D.” }

Now for this database let's create a characterized view:

InitialRel:

- From this table, we are querying task-relevant data.
- From this table, we also removed a few attributes like name and phoneno, because they make no sense in concluding insights.

PreGen

- Now, we have generalized these results by removing a few attributes and retaining important attributes.
- And also we have generalized a few attributes by naming them "Country" rather than "Birth_Place", "Age Range" rather than "Birth_data", "City" rather than "Residence" and so on as per the table given below.

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
…	…	…	…	…	…	…	…
Removed	Retained	Sci,Eng, Bus	Country	Age range	City	Removed	Excl, VG,..

PrimeGen

- Based on the PreGen plan we've performed generalization to the right level to derive a “prime generalized relation” and also we've accumulated the counts.

Gender	Major	Birth_region	Age_range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very-good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
…	…	…	…	…	…	…

Final Results

- Now we've and analyzed and concluded our final generalized results as shown below.

Gender \ Birth_Region	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

Presentation Of Results

Generalized relation:

- Relations where some or all attributes are generalized, with counts or other aggregation values accumulated.

Cross-tabulation:

- Mapping results into cross-tabulation form (similar to contingency tables).

Visualization techniques:

- Pie charts, bar charts, curves, cubes, and other visual forms.

Quantitative characteristic rules:

- Mapping generalized results in characteristic rules with quantitative information associated with it.

Mining Class Comparisons In Data Mining

Class Comparison Methods & Implementations

Data Collection:

- The set of associated data from the databases and data warehouses is collected by query processing and is partitioned into the target class and contrasting class.

Dimension Relevance Analysis:

- When many dimensions are to be processed and is required that analytical comparison should be performed, then dimension relevance analysis should be performed on these classes, and only the highly relevant dimensions are included in the further analysis.

Synchronous Generalization:

- The process of generalization is performed upon the target class to the level controlled by the user or expert specified dimension threshold, which results in a prime target class relation/cuboid.

The concepts in the contrasting class or classes are generalized to the same level as those in the prime target class relation/cuboid, forming the prime contrasting class relation/cuboid.

Presentation of the derived comparison:

- The resulting class comparison description can be visualized in the form of tables, charts, and rules.

This presentation usually includes a " contrasting" measure (such as count%) that reflects the comparison between the target and contrasting classes.

Example

Task - Compare graduate and undergraduate students using the discriminant rule.

for this, the DMQL query would be.

use University_Database
mine comparison as "graduate_students vs_undergraduate_students"
in relevance to name, gender, program, birth_place, birth_date, residence, phone_no, GPA
for "graduate_students"
where status in "graduate"
versus "undergraduate_students"
where status in "undergraduate"
analyze count%
from student

Now from this, we can formulate that

- **attributes** = name, gender, program, birth_place, birth_date, residence, phone_no, and GPA.
- **Gen(ai)** = concept hierarchies on attributes ai.
- **Ui** = attribute analytical thresholds for attributes ai.
- **Ti** = attribute generalization thresholds for attributes ai.
- **R** = attribute relevance threshold.

1. Data collection -Understanding **Target** and **Contrasting** classes.

2. Attribute relevance analysis - It is used to **remove attributes** name, gender, program, phone_no.

3. Synchronous generalization - It is controlled by user-specified dimension thresholds, a prime target, and contrasting class(es) relations/cuboids.

Initial target class working relation (graduate student)

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Jim Woodman	M	CS	Vancouver,BC, Canada	8-12-76	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-75	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-70	125 Austin Ave., Burnaby	420-5232	3.83
...	...	...	...	...	...	...	...

Initial contrasting class working relation (graduate student)

Name	Gender	Major	Birth-Place	Birth_date	Residence	Phone #	GPA
Bob Schumann	M	Chem	Calagary, Alt, Canada	10-1-78	2642 Halifax St, Burnaby	294-4291	2.96
Ammy. Eau	F	Bio	Golden, BC, Canada	30-3-76	463 Sunset Cres, Vancouer	681-5417	3.52
...	...	...	...	...	...	...	...

4. Drill down, roll up and other OLAP operations on target and contrasting classes to adjust levels of abstractions of resulting description.

Prime generalized relation for the target class: Graduate students

Major	Age_range	Gpa	Count%
Science	20-25	Good	5.53%
Science	26-30	Good	2.32%
Science	Over_30	Very_good	5.86%
…	…	…	…
Business	Over_30	Excellent	4.68%

Prime generalized relation for the contrasting class: Undergraduate students

Major	Age_range	Gpa	Count%
Science	15-20	Fair	5.53%
Science	15-20	Good	4.53%
…	…	…	…
Science	26-30	Good	5.02%
…	…	…	…
Business	Over_30	Excellent	0.68%

6. The presentation- Data is presented as generalized relations, crosstabs, bar charts, pie charts, or rules, contrasting measures to reflect a comparison between target and contrasting classes. e.g. count%

GENERALIZED REALTION

location	item	sales (in million dollars)	count (in thousands)
Asia	TV	15	300
Europe	TV	12	250
North_America	TV	28	450
Asia	computer	120	1000
Europe	computer	150	1200
North_America	computer	200	1800

Table 5.3: A generalized relation for the sales in 1997.

CROSS TAB

location \ item	TV		computer		*both_items*	
	sales	count	sales	count	sales	count
Asia	15	300	120	1000	135	1300
Europe	12	250	150	1200	162	1450
North_America	28	450	200	1800	228	2250
all_regions	45	1000	470	4000	525	5000

Table 5.4: A crosstab for the sales in 1997.

Quantitative Discriminant Rules

To find out the discriminative features of target and contrasting classes can be described as a discriminative rule.

It associates an interestingness measure **d-weight** with each tuple.

- Cj - target class
- Qa - a generalized tuple covers some tuples of class, but can also cover some tuples of contrasting class
- d-weight - range: [0, 1]

d-weight = count(Qa)/summation(count(Qa))

Example

Status	Birth_country	Age_range	Gpa	Count
Graduate	Canada	25-30	Good	90
Undergraduate	Canada	25-30	Good	210

In the above example, suppose that the count distribution for major =‘science’ and age_range = ’20..25” and GPA =‘good’ is shown in the tables.

The d_weight would be 90/(90+210) = 30% w.r.t to target class and the d_weight would be 210/(90+210) = 70% w.r.t to contrasting class. i.e.

The student majoring in science is 21 to 25 years old and has a good GPA then based on the data, there is a probability that she is a graduate student versus a 70% probability that she is an undergraduate student. Similarly, the d-weights for other tuples also can be derived.

How is class comparison performed?

Class discrimination or comparison mines characterization that categorize a target class from its contrasting classes. The target and contrasting classes should be comparable providing they share same dimensions and attributes. For instance, the three classes, person, address, and elements, are not comparable. But the sales in the last three years are comparable classes, and so are computer science candidates versus physics candidates.

The techniques developed can be continued to manage class comparison across multiple comparable classes. For instance, the attribute generalization process defined for class characterization can be changed so that the generalization is implemented synchronously between all the classes compared. This enables the attributes in some classes to be generalized to the similar levels of abstraction.

Suppose, for example, that it is given the AllElectronics data for sales in 2003 and sales in 2004 and can compare these two classes. Consider the dimension areas with abstractions at the city, province or state, and country levels. Every class of data must be generalized to the similar location level.

That is, they are synchronously all generalized to the city level, or the responsibility or state level, or the country level. This is more helpful than comparing, say, the sales in Vancouver in 2003 with the sales in the United States in 2004 (i.e., where every set of sales data is generalized to a multiple level).

The users must have the option to overwrite including automated, synchronous comparison with their choices, when chosen. There are several procedures which is as follows −

- **Data collection** − The set of relevant records in the database is collected by query processing and is separate accordingly into a target class and one or a set of contrasting classes.
- **Dimension relevance analysis** − If there are several dimensions, then dimension relevance analysis must be implemented on these classes to choose only the highly relevant dimensions for more analysis.

- **Synchronous generalization** − Generalization is implemented on the target class to the level managed by a user-or professional-specified dimension threshold, which outcomes in a prime target class relation.
- **Presentation of the derived comparison** − The resulting class comparison description can be anticipated in the form of tables, graphs, and rules. This presentation generally involves a "contrasting" measure including count% (percentage count) that reflects the comparison among the target and contrasting classes.

The user can regulate the comparison description by using drill-down, roll-up, and different OLAP operations to the target and contrasting classes, as acquired.