A Comprehensive Approach Towards Data Preprocessing Techniques & Association Rules

Jasdeep Singh Malik, Prachi Goyal, [3]Mr.Akhilesh K Sharma
[3]Assistant Professor, IES-IPS Academy, Rajendra Nagar Indore – 452012 , India
jasdeepsinghmalik@gmail.com , engineer.prachi@gmail.com
[3]akhileshshm@yahoo.com

ABSTRACT

Data pre-processing is an important and critical step in the data mining process and it has a huge impact on the success of a data mining project.[1](3) Data pre-processing is a step of the Knowledge discovery in databases (KDD) process that reduces the complexity of the data and offers better conditions to subsequent analysis. Through this the nature of the data is better understood and the data analysis is performed more accurately and efficiently. Data pre-processing is challenging as it involves extensive manual effort and time in developing the data operation scripts. There are a number of different tools and methods used for pre-processing, including: sampling, which selects a representative subset from a large population of data; transformation, which manipulates raw data to produce a single input; denoising, which removes noise from data; normalization, which organizes data for more efficient access; and feature extraction, which pulls out specified data that is significant in some particular context. Pre-processing technique is also useful for association rules algo. Like- Aprior, Partitioned, Princer-search algo. and many more algos.

KEYWORDS: KDD, Data mining, association rules, Preprocessing algos.,Data warehouse,Two sin-wave.

INTRODUCTION

Data analysis is now integral to our working lives. It is the basis for investigations in many fields of knowledge, from science to engineering and from management to process control. Data on a particular topic are acquired in the form of symbolic and numeric attributes. Analysis of these data gives a better understanding of the phenomenon of interest. When development of a knowledge-based system is planned, the data analysis involves discovery and generation of new knowledge for building a reliable and comprehensive knowledge base. Data preprocessing is an important issue for both data warehousing and data mining, as real-world data tend to be incomplete, noise, and inconsistent. Data preprocessing include data cleaning, data integration, data transformation, and data reduction. Data cleaning can be applied to remove noise and correct inconsistencies in the data. Data integration merge data from multiple source into a coherent data store, such as a data warehouse. Data transformation, such as normalization, may be applied. [2]Data reduction can reduce the data size by aggregation, elimination redundant feature, or clustering, for instance. By the help of this all data preprocessed techniques we can improve the quality of data and consequently of the mining results. Also we can improve the efficiency of mining process.

Data preprocessing techniques helpful in OLTP (online transaction Processing) and OLAP (online analytical processing). Preprocessing technique is also use full for association rules algo.like- aprior, partitional, princer search algo and many more algos. Data preprocessing is important stage for Data warehousing and Data mining.

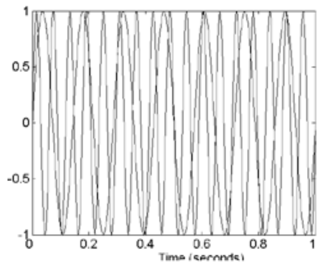
[2]Many efforts are being made to analyze data using a commercially available tool or to develop an analysis tool that meets the requirements of a particular application. Almost all these efforts have ignored the fact that some form of data pre-processing is usually required to intelligently analyze the data. This means that through data pre-processing one can learn more about the nature of the data, solve problems that may exist in the raw data (e.g. irrelevant or missing attributes in the data sets), change the structure of data (e.g. create levels of granularity) to prepare the data for a more efficient and intelligent data analysis, and solve problems such as the problem of very large data sets. There are several different types of problems, related to data collected from the real world, that may have to be solved through data pre-processing. Examples are: (i) data with missing, out of range or corrupt elements, (ii) noisy data, (iii) data from several levels of granularity, (iv) large data sets, data dependency, and irrelevant data, and (v) multiple sources of data.

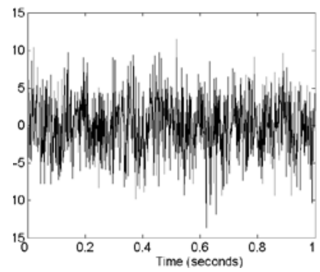
NEEDS

Problem with huge real-world database

- Incomplete Data :- Missing value.
- Noisy.
- Inconsistent.

[2](1)Noise refers to modification of original values
Examples:- distortion of a person’s voice when talking on a poor phone and “snow” on television screen





Two Sine Waves **Two Sine Waves + Noise**

WHY DATA PREPROCESSING?

Data in the real world is dirty

incomplete: missing attribute values, lack of certain attributes of interest, or containing only aggregate data

e.g., occupation=""

noisy: containing errors or outliers

e.g., Salary="-10"

inconsistent: containing discrepancies in codes or names

e.g., Age="42" Birthday="03/07/1997"

e.g., Was rating "1,2,3", now rating "A, B, C"

e.g., discrepancy between duplicate records

A well-accepted multi-dimensional view:

Accuracy,	Completeness
Consistency,	Timeline
Believability,	Value added
Interpretability,	Accessibility.

Major Tasks in Data Pre-processing

1) Data cleaning

2) Data integration

3) Data transformation

4) Data reduction

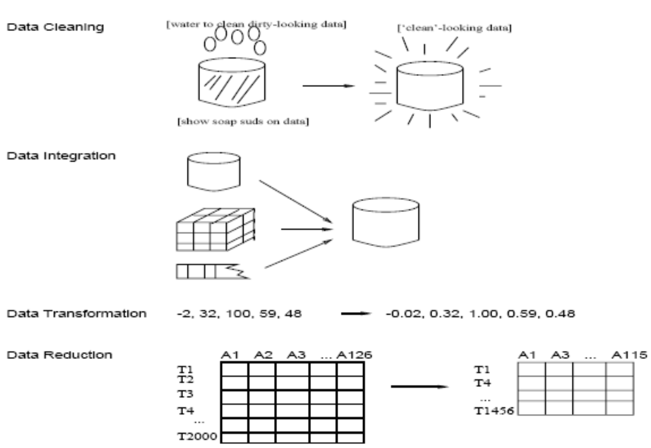


Figure 3.1: Forms of data preprocessing.

***1*) Data cleaning**

a) Missing values:

i. Ignore the tuple

ii. Fill in the missing value manually

iii. Use a global constant to fill in the missing value

iv. Use the attribute mean to fill in the missing value

v. Use the attribute mean for all samples belonging to the same class.

vi.Use the most probable value to fill in the missing value

b) Noisy data:

i. Binning

ii. Clustering

iii. Regression

c) Inconsistent data

2) Data Integration and Data Transformation

a) Data Integration

b) Data Transformation

i.Smoothing

ii.Aggregation

iii.Generalization

iv.Normalization

v. Attribute construction

3) Data reduction

a) Data cube aggregation

b) Attribute subset selection

c) Dimensional reduction .

d) Data Sampling.

e) Numerosity reduction

f) Discretization and concept hierarchy generation

1) DATA CLEANING

Real world data tend to be in complete, noisy and inconsistent .data cleaning routines attempts to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.[2]

a)Ways for handling missing values:

a. Ignore the tuple: this is usually done when class label is missing. This method is not very effective, unless tuple contains several attributes with missing values. It is especially poor when the percentage of missing values per attributes varies considerably.

b. Fill in the missing value manually: this approach is time consuming and may not be feasible given a large data set with missing values.

c. Use a global constant to fill in the missing value: replace all missing attribute values by the same constant, such as label like "unknown". if missing value are replaced by ,say unknown then the mining program may mistakenly think that they form an interesting concept ,since they all have a value common – that of "unknown". Hence, although this method is simple, it is not foolproof.

- Sorted data for price (in dollars): 4,8,15,21,21,24,25,28,34
- Partition into (equi-depth) bins:

-- Bin 1: 4, 8, 15

-- Bin 2: 21, 21, 24

-- Bin 3: 25, 28, 34

■ Smoothing by bin means:
--Bin 1: 9, 9, 9
--Bin 2: 22, 22, 22
--Bin 3: 29,29,29

■ Smoothing by bin boundaries:
--Bin 1: 4, 4, 15
--Bin 2: 21, 21, 24
--Bin 3: 25, 25, 34

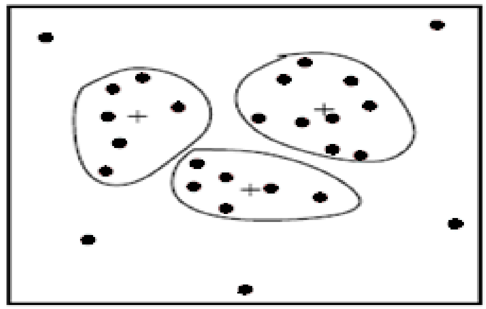


Fig:- Clustering

d. Use the attribute mean to fill in the missing value:
For example, suppose that the average income of AllElectronics customers is $28,000.Use this value to replace the missing value for income.

e. Use the attribute mean for all samples belonging to the same class as the given tuple: For example, if classifying customers according to credit risk, replace the missing value with the average income value for customers in the same credit risk category as that of the given tuple.

f. Use the most probable value to fill in the missing value: This may be determined with regression, inference-based tools using a Bayesian formalism, or decision tree induction. For example, using the other customer attributes in your data set, you may construct a decision tree to predict the missing values for income.

b) Noisy data

“What is noise?" Noise is a random error or variance in a measured variable. Given a numeric attribute such as, say, price, how can we “smooth" out the data to remove the noise? Let's look at the following data smoothing techniques.

a. Binning methods:[1] Binning methods smooth a sorted data value by consulting the “neighborhood", or values around it. The sorted values are distributed into a number of “buckets", or bins. Because binning methods consult the neighborhood of values, they perform local smoothing. Figure illustrates some binning techniques. In this example, the data for price are first sorted and then partitioned into equal-frequency bins of size 3 (i.e., each bin contains 3 values). In smoothing by bin means, each value in a bin is replaced by the mean value of the bin. For example, the mean of the values 4, 8, and 15 in Bin 1 is 9. Therefore, each original value in this bin is replaced by the value 9. Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median. In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries.
Each bin value is then replaced by the closest boundary value. In general, the larger the width, the greater the effect of the smoothing. Alternatively, bins may be equal-width, where the interval range of values in each bin is constant.

Sorted data for price (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

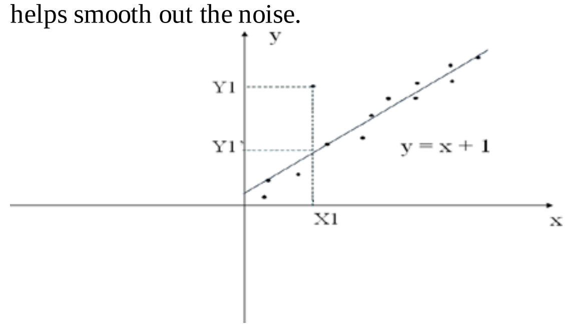
Partition into (equal-depth) bins:
Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:
Bin 1: 9, 9, 9,
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:
Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

b. Clustering: Outliers may be detected by clustering, where similar values are organized into groups or “clusters". Intuitively, values which fall outside of the set of clusters may be considered outliers in Figure.

c. **Regression**: Data can be smoothed by fitting the data to a function, such as with regression. Linear regression involves finding the “best" line to fit two variables, so that one variable can be used to predict the other. Multiple linear regression is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface. Using regression to find a mathematical equation to fit the data helps smooth out the noise.



c) Inconsistent data

There may be inconsistencies in the data recorded for some transactions. Some data inconsistencies may be corrected manually using external references. For example, errors made at data entry may be corrected by performing a paper trace. This may be coupled with routines designed to help correct the inconsistent use of codes. Knowledge engineering tools may also be used to detect the violation of known data constraints. For example, known functional dependencies between attributes can be used to find values contradicting the functional constraints.

There may also be inconsistencies due to data integration, where a given attribute can have different names in different databases. Redundancies may also result.

2) DATA INTEGRATION & TRANSFORMATION.

a) Data integration

It is likely that your data analysis task will involve data integration, which combines data from multiple sources into a coherent data store, as in data warehousing. These sources may include multiple databases, data cubes, or flat files.

There are a number of issues to consider during data integration. Schema integration can be tricky. How can like real-world entities from multiple data sources be "matched up"? This is referred to as the entity identification problem. For example, how can the data analyst or the computer is sure that customer _id in one database, and cust_number in another refer to the same entity? Databases and data warehouses typically have metadata - that is, data about the data. Such metadata can be used to help avoid errors in schema integration.

Redundancy is another important issue. An attribute may be redundant if it can be "derived" from another attributes. Inconsistencies in attribute or dimension naming can also cause redundancies in the resulting data set.

Some redundancies can be detected by **correlation analysis**.

DataIntegration : Correlation Analysis (Numerical Data)

Correlation coefficient (also called Pearson's product moment coefficient)

where n is the number of tuples, and are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(AB)$ is the sum of the AB cross-product.

- If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.
- $r_{A,B} = 0$: independent; $r_{A,B} < 0$: negatively correlated

DataIntegration : Correlation Analysis (Categorical Data)

- X^2 (chi-square) test

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- The larger the X^2 value, the more likely the variables are related

(3)The cells that contribute the most to the X^2 value are those whose actual count is very different from the expected count

Correlation does not imply causality

-- # of hospitals and # of car-theft in a city are correlated

-- Both are causally linked to the third variable: population

Chi-Square Calculation: An Example

	Play chess	Not play chess	Sum (row)
Like science fiction	250(90)	200(360)	450
Not like science fiction	50(210)	1000(840)	1050
Sum(col.)	300	1200	1500

X^2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840}$$

It shows that like_science_fiction and play_chess are correlated in the group

b) Data transformation

In data transformation, the data are transformed or consolidated into forms appropriate or mining. Data transformation can involve the following

a. **Smoothing**: which works to remove the noise from data. Such techniques include binning, clustering, and regression

b. **Aggregation:** where summary or aggregation operations are applied to the data. For example, the daily sales data may be aggregated so as to compute monthly and annual total amounts. This step is typically used in

constructing a data cube for analysis of the data at multiple granularities

c. **Generalization of the data**: where low level or "primitive" (raw) data are replaced by higher level concepts through the use of concept hierarchies. For example, categorical attributes, like street, can be generalized to higher level concepts, like city or county. Similarly, values for numeric attributes, like age, may be mapped to higher level concepts, like young, middle-aged, and senior

d. **Normalization**: where the attribute data are scaled so as to fall within a small specified range, such as -1.0 to 1.0, or 0 to 1.0.

e. **Attribute construction**: where new attributes are constructed and added from the given set of attributes to help the mining process.

(3)Smoothing is a form of data cleaning, Aggregation and generalization also serve as forms of data reduction. We therefore discuss normalization and attribute construction. An attribute is normalized by scaling its values so that they fall within a small specified range, such as 0 to 1.0.

Normalization is particularly useful for classification algorithms involving neural networks, or distance measurements such as nearest-neighbor classification and clustering. If using the neural network back propagation algorithm for classification mining normalizing the input values for each attribute measured in the training samples will help speed up the learning phase. For distance-based methods, normalization helps prevent attributes with initially large ranges (e.g., income) from outweighing attributes with initially smaller ranges (e.g.binary attributes).

There are many methods for data normalization. We study three: min-max normalization, z-score normalization, and normalization by decimal scaling.

Min-max normalization performs a linear transformation on the original data. Suppose that minA and maxA are the minimum and maximum values of an attribute A. Min-max normalization maps a value ,v of A to v' in the range [new _minA; new _maxA] by computing

$$v' = \frac{v - minA}{maxA - minA} * (new\_maxA - new\_minA) + new\_minA$$

Min-max normalization preserves the relationships among the original data values. It will encounter an "out of bounds" error if a future input case for normalization falls outside of the original data range for A.

3) DATA REDUCTION

Why data reduction?

A database/data warehouse may store terabytes of data

Complex data analysis/mining may take a very long time to run on the complete data set

Data reduction

Obtain a reduced representation of the data set that is much smaller in volume but yet produce the same (or almost the same) analytical results

Data reduction strategies

Aggregation

Sampling

Dimensionality Reduction

Feature subset selection

Feature creation

Discretization and Binarization

Attribute Transformation

Imagine that you have selected data from the AllElectronics data warehouse for analysis. The data set will likely be huge! Complex data analysis and mining on huge amounts of data may take a very long time, making such analysis impractical or infeasible.

Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data. That is, mining on the reduced data set should be more efficient yet produce the same analytical results.

Strategies for data reduction include the following.

a**. Data cube aggregation**: where aggregation operations are applied to the data in the construction of a data cube.

b. **Dimension reduction**: where irrelevant, weakly relevant, or redundant attributes or dimensions may be detected and removed.

c. **Data compression:** where encoding mechanisms are used to reduce the data set size.

d. **Numerosity reduction**: where the data are replaced or estimated by alternative, smaller data representations such as parametric models, or nonparametric methods such as clustering, sampling, and the use of histogram.

e. **Discretization and concept hierarchy generation**: where raw data values for attributes are replaced by ranges or higher conceptual levels.

Discretization and Concept hierarchies are powerful tools of data mining, in that they allow the mining of data at multiple levels of abstraction.

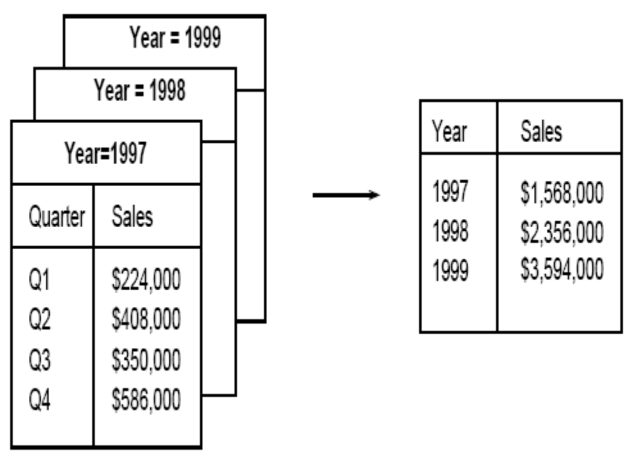


Figure : Sales data for a given branch of All Electronics for the years 1997 to 1999. In the data on the left, the
sales are shown per quarter. In the data on the right, the data are aggregated to provide the annual sales.

a) Data cube aggregation

Imagine that you have collected the data for your analysis. These data consist of the AllElectronics sales per quarter,for the years 1997 to 1999. You are, however, interested in the annual sales (total per year), rather than the total per quarter. Thus the data can be aggregated so that the resulting data summarize the total sales per year instead of per quarter. This aggregation is illustrated in Figure. The resulting data set is smaller in volume, without loss of information necessary for the analysis task.

(3)Data cubes store multidimensional aggregated information. For example, Figure : shows a data cube for multidimensional analysis of sales data with respect to annual sales per item type for each AllElectronics branch. Each cell holds an aggregate data value, corresponding to the data point in multidimensional space. Concept hierarchies may exist for each attribute, allowing the analysis of data at multiple levels of abstraction. For example, a hierarchy for branch could allow branches to be grouped into regions, based on their address. Data cubes provide fast access to
precomputed, summarized data, thereby benefiting on-line analytical processing as well as data mining.

The cube created at the lowest level of abstraction is referred to as the base *cuboid*. A cube for the highest level of abstraction is the *apex cuboid*. For the sales data of Figure, the apex cuboid would give one total- the total sales for all three years, for all item types, and for all branches. Data cubes created for varying levels of abstraction are sometimes referred to as *cuboids*, so that a "data cube" may instead refer to a *lattice of cuboids*. Each higher level of abstraction further reduces the resulting data size. When replying to data mining requests, the smallest available cuboid relevant to the given task should be used.

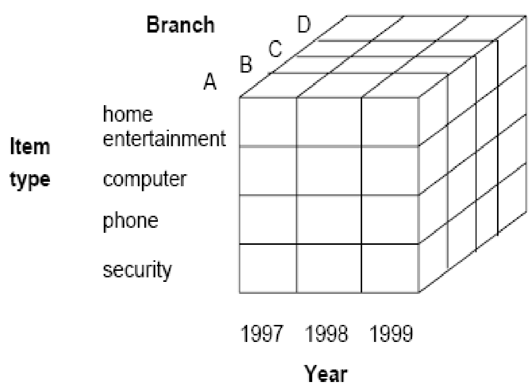
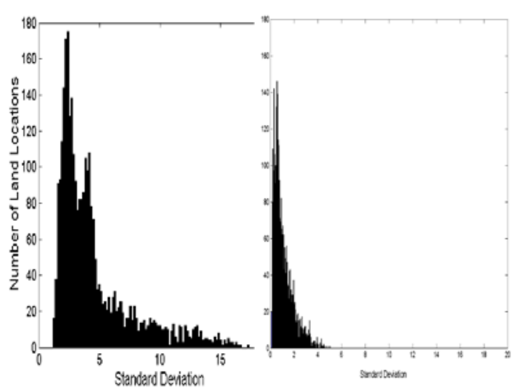


Fig:-- A data cube for Sale at Allelectronics

Variation of Precipitation in Australia



Standard Deviation of Average Monthly Precipitation

Standard Deviation of Avg. Yearly Precipitation

b) Attribute Subset Selection

Attribute subset selection reduces the data set size by removing such attributes (or dimensions) from it. Typically, methods of attribute subset selection are applied. The goal of attribute subset selection is to find a minimum set of attributes such that the resulting probability distribution of the data classes is as close as possible to the original distribution obtained using all attributes. Mining on a reduced set of attributes has an additional benefit. It reduces the number of attributes appearing in the discovered patterns, helping to make the patterns easier to understand.

[2]Basic heuristic methods of attribute subset selection include the following techniques, some of which are illustrated in Figure:

a. Step-wise forward selection: The procedure starts with an empty set of attributes. The best of the original attributes is determined and added to the set. At each subsequent iteration or step, the best of the remainingoriginal attributes is added to t

b. Step-wise backward elimination: The procedure starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.

c. Combination forward selection and backward elimination: The step-wise forward selection and backward elimination methods can be combined so that, at each step, the procedure selects the best attribute and
removes the worst from among the remaining attributes.

d. Decision tree induction: Decision tree algorithms, were originally intended for classification. Decision tree induction constructs a flow-chart-like structure where each internal (non-leaf) node denotes a test on an attribute, each branch corresponds to an outcome of the test, and each external (leaf) node denotes a class prediction. At each node, the algorithm chooses the "best" attribute to partition the data into individual classes.

When decision tree induction is used for attribute subset selection, a tree is constructed from the given data. All attributes that do not appear in the tree are assumed to be irrelevant. The set of attributes appearing in the tree form the reduced subset of attributes.

Sampling

Choose a representative subset of the data

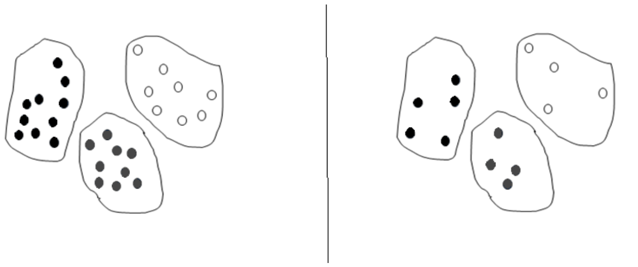
---Simple random sampling may have poor performance in the presence of skew.

Develop adaptive sampling methods

---Stratified sampling:

Approximate the percentage of each class (or subpopulation of interest) in the overall database

--- Used in conjunction with skewed data



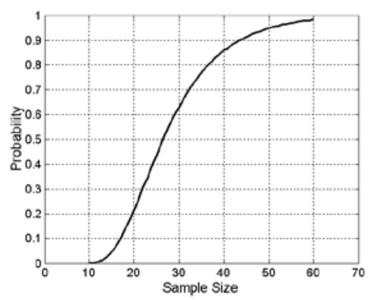
Raw Data Cluster/Stratified Sample

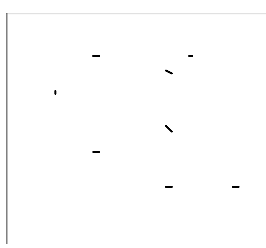
Types of sampling

Simple Random SamplinSimple Random Sampling There is an equal probability of selecting any particular item Sampling without replacement As each item is selected, it is removed from thepopulation Sampling with replacement Objects are not removed from the population as they are selected for the sample. In sampling with replacement, the same object can be picked up more than once Stratified sampling Split the data into several partitions; then draw random samples from each partition

Sample size:-

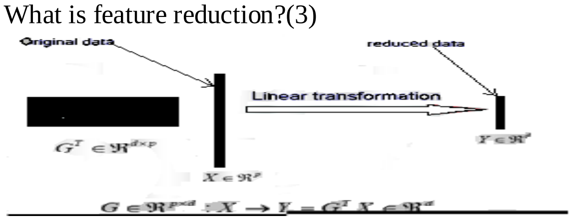
What sample size is necessary to get at least one object from each of 10 groups.





Probability a sample contains pts from each group

What is feature reduction?(3)



c. Dimensionality reduction

In Dimensionality reduction, data encoding or transformation are applied so as to obtain a reduced or "compressed" representation of the original data .if the original data can be reconstructed from the compressed data without any loss of information, the data reduction is called lossless .if ,instead ,we can reconstruct only an approximation of the original data, then the reduction is called lossy. There are special well-tuned algorithms for string compression .although they are typically lossless, they allow only limited manipulation of the data. There are two popular and effective methods of lossy dimensionality reduction:

1) Wavelets transform
2) Principal components analysis

DataReduction : Feature Creation

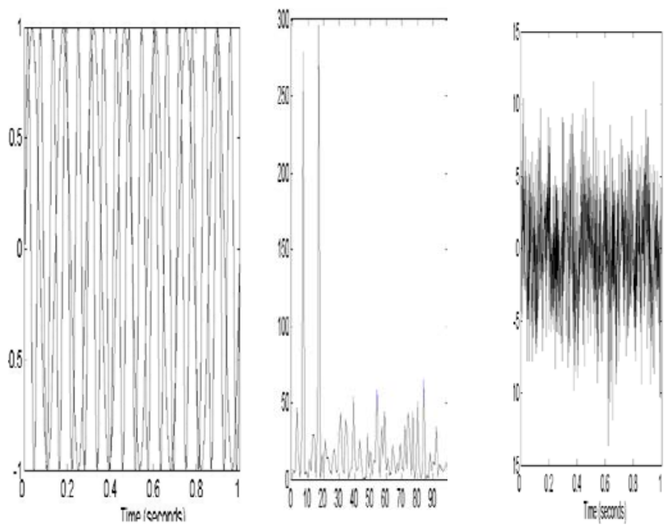
Create new attributes that can capture the important information in a data set much more efficiently than the original attributes

Three general methodologies:

1.Feature Extraction 2. domain-specific 3. Mapping Data to New Space 4.Feature Construction 5.combining features

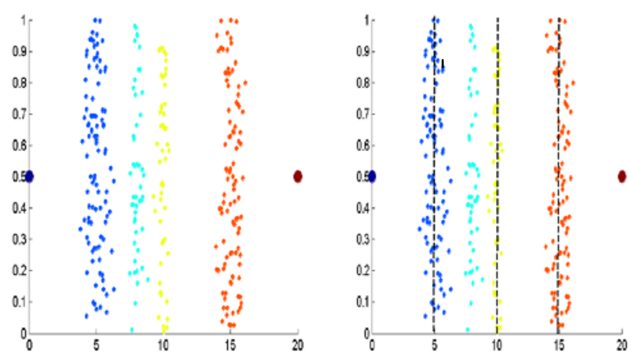
Data Reduction : Mapping Data to a New Space

- Fourier transform
- Wavelet transform

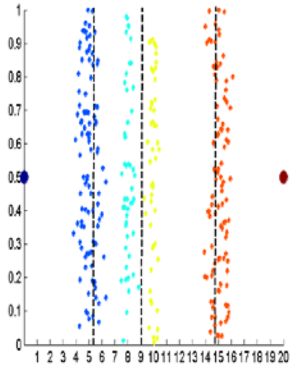


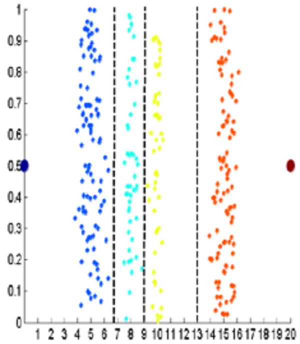
Two Sine Waves Two Sine Waves + Noise Frequency

DataReduction : Discretization Without Using Class Labels



Data Equal interval width





Equal frequency K-means

DataReduction : Attribute Transformation
A function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
Simple functions: x^k, log(x), e^x, |x|
Standardization and Normalization

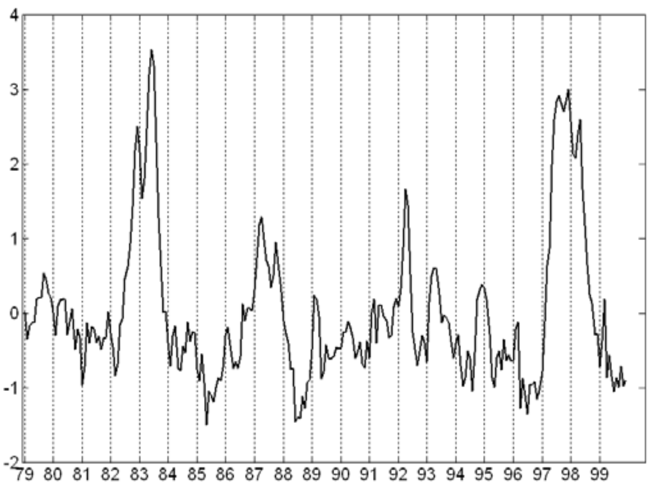


Diagram :-- KDD Process

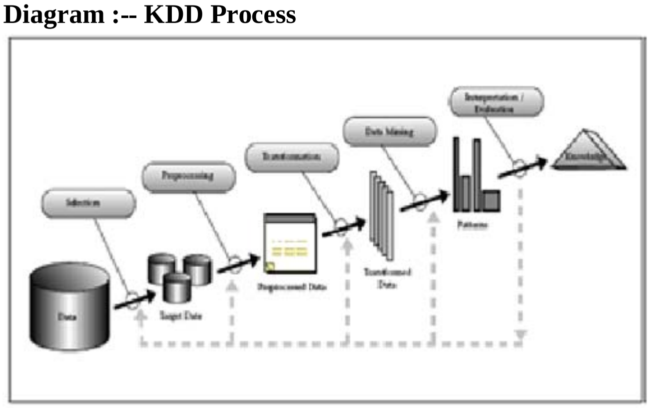
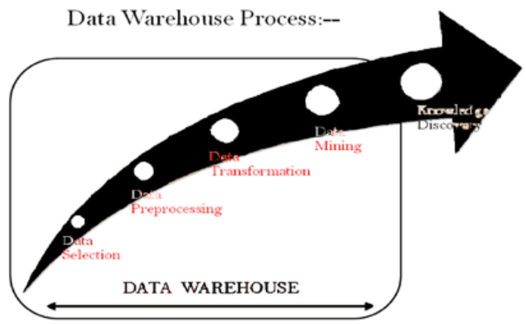


Figure 1. An Overview of the Steps That Compose the KDD Process.

Diagram :-- Data Warehouse



OVERVIEW:
Data preparation is an important issue for both data warehousing and data mining, as real-world data tends to be incomplete, noisy, and inconsistent. Data preparation includes data cleaning, data integration, data transformation, and data reduction.

Data cleaning routines can be used to fill in missing values, smooth noisy data, identify outliers, and correct data inconsistencies.

Data integration combines data from multiples sources to form a coherent data store. Metadata, correlation analysis, data conflict detection, and the resolution of semantic heterogeneity contribute towards smooth data integration.

Data transformation routines convert the data into appropriate forms for mining. For example, attribute data may be normalized so as to fall between small ranges, such as 0 to 1.0

Data reduction techniques such as data cube aggregation, dimension reduction, data compression, Numerosity reduction, and Discretization can be used to obtain a reduced representation of the data, while minimizing the loss of information content.

Data Discretization and Automatic generation of concept hierarchies For numeric data, techniques such as binning, histogram analysis, and clustering analysis can be used.

Although several methods of data preparation have been developed, data preparation remains an active area of research.

4.ANALYSIS

After data preprocessing, we will have data preprocessing result in the tabular form that contains spatial data transformation result. Mining process of spatial association rule used Apriori based [1] and FP-Growth algorithms [6]. The reason of using of both algorithms are widely used of those algorithms and using of two different approaches. Some interesting patterns got from mining process are:

IF Population_Density_Low
THEN DBD_Low (0.61) (0.75)
IF Health_Facility_No
AND Population_Density_ Low
THEN DBD_Low (0.47) (0.8)
IF Health_Facility_No
AND Population_Density_ Low
AND Close-to_Bog
THEN DBD_Low (0.86) (0.86) Etc

From software testing, it is indicated that data preprocessing is time consuming. This caused by spatial joint process execution that require much time.

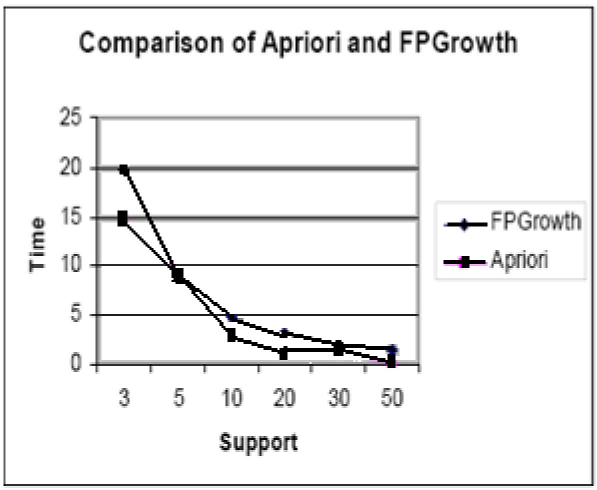


Figure . Comparison of Apriori and FP-Growth

Executing both association rule algorithms resulted or indicates that both algorithms generate the same patterns. Another interesting result is that Apriori algorithm is faster than FP-Growth. This result may be caused by relatively few number of data (there are 163 records of subdistrictin Surabaya). Another reason is the patterns that used in the case study is not a long pattern, whereas, one of several FP-Growth advantages is better to work with long pattern [6].

5. CONCLUSION & FURTHER WORK

In this paper, we have proposed methodology and implementation of data preprocessing and then performed mining l association rule with conventional association algorithms. (5)Main steps in this data preprocessing is spatial and non-spatial feature selection based on parameter determined, reduction of dimension, selection and categorization of non-spatial attributes, join operation for the spatial objects based on spatial parameter spatial and transforms into form of output wanted. While finding spatial association rule used Apriori and FP-Growth algorithm. For the near future, we plan to continue this research to accommodate temporal constraint to spatial association rule mining.We are doing further enhancement in data preprocessing techniques.

6. REFERENCES

[1] Data Pre-processing & Mining Algorithm, Knowledge & Data Mining & Preprocessing, 3rd edition, *Han & Kamber*.

[2]Mohd Helmy Abd Wahab, *Mohd Norzali Haji Mohd, Hafizul Fahri Hanafi*, *Mohamad Farhan(1998) Mohamad Mohsin*

[3]. Agrawal, Rakesh and Ramakrishnan Srikant, “Fast Algorithms for Mining & Preprocessing Assosiation Rules”, *Proceedings of the 20th VLDB Conference*, Santiago, Chile (1994).

[4] Salleb, Ansaf and Christel Vrain, “An Application of Assosiation *Knowledge Discovery and Data Mining (PKDD) 2000*, LNAI 1910, pp. 613-618, Springer Verlag (2000).

[5] Agrawal, R., and Psaila, G. 1995. Active Data Mining. In Proceedings on Knowledge Discovery and Data Mining (KDD-95), 3–8. Menlo Park.