\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Module 2 Link*** - **[http://a.impartus.com/ilc/#/video/id/4682002](http://a.impartus.com/ilc/#/video/id/4653903)**

***T2-Data-Mining-Concepts-&-Techniques / Chapter 3 - Data Preprocessing***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DATA PREPROCESSING**

* Preprocess data in order to help improve the quality of the data and, consequently, of the mining results.
  + **Data cleaning** : smoothing noise, fixing inconsistencies, filling in missing values, identifying or removing outliers
  + **Data integration** : merge data from multiple sources into a coherent data store such as a data warehouse. Naming inconsistencies
  + **Data reduction** : reduce data size by, for instance, aggregating, eliminating redundant features, or clustering.
    - **Dimensionality reduction**, data encoding schemes are applied so as to obtain a reduced or “compressed” representation of the original data. Examples include data compression techniques (e.g., *wavelet transforms* and *principal components analysis*), *attribute subset selection* (e.g., removing irrelevant attributes), and *attribute construction* (e.g., where a small set of more useful attributes is derived from the original set).
    - **Numerosity reduction**, the data are replaced by alternative, smaller representations using parametric models (e.g., *regression* or *log-linear models*) or nonparametric models (e.g., *histograms, clusters*, *sampling*, or *data aggregation*).
    - **Data Compression**
  + **Data transformations** (e.g., normalization) : improves the accuracy and efficiency of mining algorithms involving distance measurements. Eg, Discretization and concept hierarchy generation
* **Data Challenges** :
  + **Incomplete** (lacking attribute values or certain attributes of interest, or containing only aggregate data)
  + **Inaccurate** / noisy (containing errors, or values that deviate from the expected)
    - Data collection instruments may be faulty.
    - Human or computer errors occurring at data entry.
    - Users may purposely submit incorrect data values for mandatory fields.
    - Errors in data transmission can also occur.
    - Technology limitations such as limited buffer size for coordinating synchronised data transfer and consumption.
    - Inconsistencies in naming conventions or data codes, or inconsistent formats for input fields (e.g., *date*). Duplicate tuples also require data cleaning
  + **Inconsistent** (e.g., containing discrepancies in the department codes used to categorise items)
    - Attributes of interest may not always be available, such as customer information for sales transaction data.
    - not be included simply because they were not considered important at the time of entry.
    - not be recorded due to a misunderstanding or because of equipment malfunctions.
    - Data that were inconsistent with other recorded data may have been deleted.
    - recording of the data history or modifications may have been overlooked. Missing data, particularly for tuples with missing values for some attributes, may need to be inferred.
* **Timeliness , Believability, Interpretability**

1. **DATA CLEANING**

* **Missing Values** : It is important to note that, in some cases, a missing value may not imply an error in the data!
  + Ignore the records / tuple
  + Fill manually
  + Replace by the same constant such as a label like *“Unknown”* or −∞. The mining program may mistakenly think that the “Unknown” form an interesting concept.
  + Replace with mean for normal (symmetric) data distributions, while skewed data distribution should employ the median
  + Replace with mean or median for all samples belonging to the same target class
  + 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.
* Methods 3 through 6 bias the data—the filled-in value may not be correct. Method 6, however, in comparison to the other methods uses the most information from the present data to predict missing values.
* **Noisy Data** : random error or variance; smooth the data
  + **Binning** : local smoothing; sorted values are distributed into a number of “buckets,” or bins; concept hierarchy generation
  + **Regression**
  + **Clustering**, for example, where similar values are organised into groups. values that fall outside of the set of clusters may be considered outliers

**Smoothing techniques described before reduce the number of distinct values per attribute. This acts as a form of data reduction for logic-based data mining methods, such as decision tree induction, which repeatedly makes value comparisons on sorted data.**

* **Discrepancy detection** :
  + Causes
    - poorly designed data entry forms that have many optional fields,
    - human error in data entry,
    - deliberate errors (e.g., respondents not want- ing to divulge information about themselves), and
    - data decay (e.g., outdated addresses).
    - inconsistent data representations and inconsistent use of codes.
    - errors in instrumentation devices that record data and system errors.
    - data (inadequately) used for purposes other than originally intended.
    - data integration (e.g., where a given attribute can have different names in different databases).
  + Solutions
    - **metadata,** use any knowledge you may already have regarding properties of the data.
    - what are the data type and domain of each attribute?
    - What are the acceptable values for each attribute?
    - find the mean, median, and mode values.
    - Are the data symmetric or skewed?
    - What is the range of values?
    - Do all values fall within the expected range?
    - What is the standard deviation of each attribute? Values that are more than two standard deviations away from the mean for a given attribute may be flagged as potential outliers.
    - Are there any known dependencies between attributes?
    - **A unique rule** says that each value of the given attribute must be different from all other values for that attribute.
    - A **consecutive rule** says that there can be no miss- ing values between the lowest and highest values for the attribute, and that all values must also be unique (e.g., as in check numbers).
    - A **null rule** specifies the use of blanks, question marks, special characters, or other strings that may indicate the null condition (e.g., where a value for a given attribute is not available), and how such values should be handled.
  + Commercial Tools
    - **Data scrubbing tools** use simple domain knowledge (e.g., knowledge of postal addresses and spell-checking) to detect errors and make corrections in the data. These tools rely on parsing and fuzzy matching techniques when cleaning data from multiple sources.
    - **Data auditing tools** find discrepancies by analyzing the data to discover rules and relationships, and detecting data that violate such conditions. They are variants of data mining tools. For example, they may employ statistical analysis to find correlations, or clustering to identify outliers. They may also use the basic statistical data descriptions presented in Section 2.2.
    - **Data migration tools** allow simple transformations to be specified such as to replace the string *“gender”* by *“sex.”*
    - **ETL (extraction/transformation/loading) tools** allow users to specify transforms through a graphical user interface (GUI). These tools typically support only a restricted set of transforms so that, often, we may also choose to write custom scripts for this step of the data cleaning process.

The two-step process of discrepancy detection and data transformation (to correct discrepancies) iterates.

**b) DATA INTEGRATION**

* 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.
* **Entity identification problem** : equivalent real-world entities from multiple data sources be matched up. Schema integration and object matching.
  + customer id in one database and cust number in another refer to the same attribute
  + **Metadata** can be used to help avoid errors in schema integration. Examples of metadata for each attribute include the name, meaning, data type, and range of values permitted for the attribute, and null rules for handling blank, zero, or null values. Helps transform the data (e.g., where data codes for pay type in one database may be “H” and “S” but 1 and 2 in another).
  + Attribute functional dependencies and referential constraints in the source system match those in the target system. For example, in one system, a discount may be applied to the order, whereas in another system it is applied to each individual line item within the order. If this is not caught before integration, items in the target system may be improperly discounted.
* **Redundancy and Correlation Analysis** 
  + An attribute may be redundant if it can be “derived” from another attribute or set of attributes.
  + Inconsistencies in attribute or dimension naming
  + Can be detected by **correlation analysis**. Given two attributes, such analysis can measure how strongly one attribute implies the other, based on the available data.
  + Correlation does not imply causality
  + For **nominal / categorical data**, we use the **χ2 (*chi-square*) test**.
    - Null Hypothesis : No relation / independent
    - Alternate Hypothesis : Correlated
    - Expected value = ( row total \* column total ) / total
    - Degree of freedom = (number of row – 1) \* (number of column – 1)
    - Find theoretical χ2 based on significance level



* + The larger the Χ2 value, the more likely the variables are related. The groups that contribute the most to the Χ2 value are those whose actual count is very different from the expected count
  + For **numeric attributes**, we can use the correlation coefficientand covariance.
  + **Correlation coefficient ( Pearson’s product moment coefficient )**



* + - −1 ≤ *rA*,*B* ≤ +1.
    - If *rA*,*B* is greater than 0, then *A* and *B* are *positively correlated*, meaning that the values of *A* increase as the values of *B* increase.
    - The higher the value, the stronger the correlation (i.e., the more each attribute implies the other). Hence, a higher value may indicate that *A* (or *B*) may be removed as a redundancy.
    - If the resulting value is equal to 0, then *A* and *B* are *independent* and there is no correlation between them.
    - If the resulting value is less than 0, then *A* and *B* are *negatively correlated*, where the values of one attribute increase as the values of the other attribute decrease.
  + **Covariance**





* + - If *A* is larger than *A* ̄ (the expected value of *A*), then *B* is likely to be larger than *B* ̄ (the expected value of *B*). Therefore, the covariance between *A* and *B* is *positive*.
    - If one of the attributes tends to be above its expected value when the other attribute is below its expected value, then the covariance of *A* and *B* is *negative*.
    - If *A* and *B* are *independent* (i.e., they do not have correlation), then *E*(*A* · *B*) = *E*(*A*) · *E*(*B*). Therefore, the covariance is *Cov*(*A*, *B*) = *E*(*A* · *B*) − *A* ̄ *B* ̄ = *E*(*A*) · *E*(*B*) − *A* ̄ *B* ̄ = 0.
    - However, the converse is not true. Some pairs of random variables (attributes) may have a covariance of 0 but are not independent.
* **Duplicate Tuple**
  + two or more identical tuples for a given unique data entry case
  + use of denormalized tables; inaccurate data entry or updating some but not all data occurrences
* **Data Value Conflict**
  + attribute values from different sources may differ. This may be due to differences in representation, scaling, or encoding.
  + For example, price may involve different currencies, different grade system
  + transformation rules for conversion might not be clear

**c) DATA REDUCTION**

* **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 and produces the same (or almost the same) analytical results.
  + Complex data analysis and mining on huge amounts of data can take a long time, making such analysis impractical or infeasible.
  + reduce storage requirement
  + Better speed
  + remove redundant features
* **Dimensionality reduction**, reduce the number of random variables or attributes under consideration
  + **Wavelet transforms**
  + **Principal Components Analysis (PCA)**
    - The input data are normalised.
    - computes *k* orthonormal vectors that provide a basis for the normalised input data. These are unit vectors that each point in a direction perpendicular to the others. These vectors are referred to as the *principal components*. The input data are a linear combination of the principal components.
    - The principal components are sorted in order of decreasing “significance” or strength, serve as a new set of axes for the data, providing important information about variance.
    - Because the components are sorted in decreasing order of “significance,” the data size can be reduced by eliminating the weaker components, that is, those with low variance.
    - Using the strongest principal components, it should be possible to reconstruct a good approximation of the original data.
    - **PCA can be applied to ordered and unordered attributes, and can handle sparse data and skewed data.**
    - In comparison with wavelet transforms, PCA tends to be better at handling sparse data, whereas wavelet transforms are more suitable for data of high dimensionality.
  + **Feature or attribute subset selection**
    - 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
    - For *n* attributes, there are 2*n* possible subsets, prohibitively expensive, especially as *n* and the number of data classes increase.
    - Heuristic methods, typically greedy. The “best” (and “worst”) attributes are typically determined using tests of statistical significance, which assume that the attributes are independent of one another.
    - Many other attribute evaluation measures can be used such as the *information gain* measure used in building decision trees for classification
    - **Stepwise forward selection**: starts with an empty set of attributes as the reduced set. The best are added to the reduced set.
    - **Stepwise backward elimination**: starts with the full set of attributes. At each step, it removes the worst attribute remaining in the set.
    - **Combination of forward selection and backward elimination**: at each step, the procedure selects the best attribute and removes the worst from among the remaining attributes.
    - **Decision tree induction**: At each node, the algorithm chooses the “best” attribute to partition the data into individual classes. All attributes that do not appear in the tree are assumed to be irrelevant.
    - The stopping criteria for the methods may vary.



* **Numerosity reduction**
  + **Parametric**
    - **Regression and Log-Linear Models**
    - Both used on sparse data, although their application may be limited. While both methods can handle skewed data, regression does exceptionally well.
    - Regression can be computationally intensive when applied to high-dimensional data, whereas log-linear models show good scalability for up to 10 or so dimensions.
  + **Non-parametric**
    - **Histograms**
      * highly effective at approximating both sparse and dense data, as well as highly skewed and uniform data.
      * Multidimensional histograms can capture dependencies between attributes, found effective with up to five attributes.
      * Singleton buckets are useful for storing high-frequency outliers.
    - **Clustering**
      * partition the objects into groups, or *clusters*, so that objects within a cluster are “similar” to one another and “dis- similar” to objects in other clusters.
      * Similarity is how “close” the objects are in space, based on a distance function. The “quality” of a cluster may be represented by its diameter.
      * Centroid distanceis average distance of each cluster object from the cluster centroid
    - **Sampling**
      * **Simple random sample without replacement (SRSWOR) of size** *s*: This is created by drawing *s* of the *N* tuples from *D* (*s* < *N* ), where the probability of drawing any tuple in *D* is 1/*N.*
      * **Simple random sample with replacement (SRSWR) of size** *s*: This is similar to SRSWOR, except that each time a tuple is drawn from *D*, it is recorded and then *replaced*.
      * **Cluster sample**: If the tuples in *D* are grouped into *M* mutually disjoint “clusters,” then an SRS of *s* clusters can be obtained, where *s* < *M*. For example, retrieved a page at a time, a spatial database based on how closely different areas are located.
      * **Stratified sample**: If *D* is divided into mutually disjoint parts called *strata,* a stratified sample of *D* is generated by obtaining an SRS at each stratum. This helps ensure a representative sample, especially when the data are skewed. For example, a stratified sample may be obtained from customer data, where a stratum is created for each cus- tomer age group. In this way, the age group having the smallest number of customers will be sure to be represented.

An advantage of sampling for data reduction is that the cost of obtaining a sample *is proportional to the size of the sample*, *s*, as opposed to *N*, the data set size. Hence, sampling complexity is potentially *sublinear* to the size of the data. Other data reduction techniques can require at least one complete pass through *D*. For a fixed sample size, sampling complexity increases only linearly as the number of data dimensions, *n*, increases, whereas techniques using histograms, for example, increase exponentially in *n*.

When applied to data reduction, sampling is most commonly used to estimate the answer to an aggregate query. It is possible (using the central limit theorem) to deter- mine a sufficient sample size for estimating a given function within a specified degree of error. This sample size, *s*, may be extremely small in comparison to *N*. Sampling is a natural choice for the progressive refinement of a reduced data set. Such a set can be further refined by simply increasing the sample size.



* + - **Data cube aggregation**
* **Data compression**
  + If the original data can be *reconstructed* from the compressed data without any information loss, the data reduction is called **lossless**.
  + If, instead, we can reconstruct only an approximation of the original data, then the data reduction is called **lossy**.

**d) DATA TRANSFORMATION**

* **Smoothing**, remove noise, binning, regression, and clustering.
* **Attribute construction**
* **Aggregation**
* **Normalization**, where the attribute data are scaled so as to fall within a smaller range, such as −1.0 to 1.0, or 0.0 to 1.0.

In general, expressing an attribute in smaller units will lead to a larger range for that attribute, and thus tend to give such an attribute greater effect or “weight.”

* **Discretization**
  + If the discretization process uses class information, then we say it is **supervised** **discretization**. Otherwise, it is unsupervised.
  + **Top-down discretization** or splitting, starts by first finding one or a few points (called *split points* or *cut points*) to split the entire attribute range, and then repeats this recursively on the resulting intervals.
  + **Bottom-up discretization** or merging, starts by considering all of the continuous values as potential split-points, removes some by merging neighbourhood values to form intervals, and recursively applies this process to the resulting intervals.
  + **Data discretization** or concept hierarchy generation. **Concept hierarchy generation** for nominal data, where attributes such as *street* can be generalized to higher-level concepts, like *city* or *country*.