DATA MINING

Data Pre-processing

Measure of Data Quality

- Accuracy
- Completeness
- Consistency
- Timeliness
- Believability
- Interpretability
- Accessibility
- Value added

Why Pre-Processing ?

- Real world data are generally
  - Incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
  - Noisy: containing errors or outliers
  - Inconsistent: containing discrepancies in codes or names

Why Pre-Processing ?

- Tasks in data preprocessing
  - Data cleaning: fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies.
  - Data integration: using multiple databases, data cubes, or files.
  - Data transformation: normalization and aggregation.
  - Data reduction: reducing the volume but producing the same or similar analytical results.
  - Data discretization: part of data reduction, replacing numerical attributes with nominal ones.

Data Cleaning

- Fill in missing values (attribute or class value):
  - Ignore the tuple: usually done when class label is missing.
  - Use the attribute mean (or majority nominal value) to fill in the missing value.
  - Use the attribute mean (or majority nominal value) for all samples belonging to the same class.
  - Predict the missing value by using a learning algorithm: consider the attribute with the missing value as a dependent (class) variable and run a learning algorithm (usually Bayes or decision tree) to predict the missing value.

Data Cleaning

- Identify outliers and smooth out noisy data:
  - Binning
    - Sort the attribute values and partition them into bins (see "Unsupervised discretization" below);
    - Then smooth by bin means, bin median, or bin boundaries.
  - Clustering: group values in clusters and then detect and remove outliers (automatic or manual)
  - Regression: smooth by fitting the data into regression functions.
  - Correct inconsistent data: use domain knowledge or expert decision.

Data Transformation

- Normalization:
  - Scaling attribute values to fall within a specified range.
    - Example: to transform V in [min, max] to V' in [0,1], apply V'=(V-Min)/(Max-Min)
  - Scaling by using mean and standard deviation (useful when min and max are unknown or when there are outliers): V'=(V-Mean)/StDev
- Aggregation: moving up in the concept hierarchy on numeric attributes.
- Generalization: moving up in the concept hierarchy on nominal attributes.
- Attribute construction: replacing or adding new attributes inferred by existing attributes.

Data Reduction

- Reducing the number of attributes
  - Data cube aggregation: applying roll-up, slice or dice operations.
  - Removing irrelevant attributes: attribute selection (filtering and wrapper methods), searching the attribute space (see Lecture 5: Attribute-oriented analysis).
  - Principle component analysis (numeric attributes only): searching for a lower dimensional space that can best represent the data.
- Reducing the number of attribute values
  - Binning (histograms): reducing the number of attributes by grouping them into intervals (bins).
  - Clustering: grouping values in clusters.
  - Aggregation or generalization
- Reducing the number of tuples
  - Sampling

Discretization

- Unsupervised discretization - class variable is not used.
  - Equal-interval (equal width) binning: split the whole range of numbers in intervals with equal size.
  - Equal-frequency (equal depth) binning: use intervals containing equal number of values.
- Supervised discretization - uses the values of the class variable.
  - Using class boundaries. Three steps:
    - Sort values.
    - Place breakpoints between values belonging to different classes.
    - If too many intervals, merge intervals with equal or similar class distributions.
  - Entropy (information)-based discretization.