CH:7

Flexible min-support thresholds: Some items are more valuable but less frequent
Redundancy Filtering: Some rules may be redundant due to "ancestor" relationships between items

- Categorical Attributes: finite number of possible values, no ordering among values—data cube approach
- Quantitative Attributes: Numeric, implicit ordering among values—discretization, clustering, and gradient approaches

Mining Quantitative Associations: Techniques can be categorized by how numerical attributes, such as age or salary are treated:

1. Static discretization based on predefined concept hierarchies (data cube methods)
2. Dynamic discretization based on data distribution
3. Clustering: Distance-based association
4. Deviation

Rare patterns: Very low support but interesting

- E.g., buying Rolex watches
- Mining: Setting individual-based or special group-based support threshold for valuable items

Negative patterns Since it is unlikely that one buys Ford Expedition (an SUV car) and Toyota Prius (a hybrid car) together, Ford Expedition and Toyota Prius are likely negatively correlated patterns

Constraint-based mining:

- User flexibility: provides constraints on what to be mined
- Optimization: explores such constraints for efficient mining

Constraints in Data Mining:

- Knowledge type constraint: classification, association
- Data constraint — using SQL-like queries
- Dimension/level constraint
- Rule (or pattern) constraint
- Interestingness constraint

Constraint-Based Frequent Pattern Mining:

- Pattern space pruning constraints
  - Anti-monotonic: If constraint c is violated, its further mining can be terminated
  - Monotonic: If c is satisfied, no need to check c again
  - Succinct: c must be satisfied, so one can start with the data sets satisfying c
  - Convertible: c is not monotonic nor anti-monotonic, but it can be converted into it if items in the transaction can be properly ordered
- Data space pruning constraint
  - Data succinct: Data space can be pruned at the initial pattern mining process
  - Data anti-monotonic: If a transaction t does not satisfy c, t can be pruned from its further mining

Can Apriori Handle Convertible Constraints?

- A convertible, not monotone nor anti-monotone nor succinct constraint cannot be pushed deep into the an Apriori mining algorithm
- But it can be pushed into frequent-pattern growth framework

CH8:

Supervised vs. Unsupervised Learning:

- Supervised learning (classification)
  - Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
  - New data is classified based on the training set
- Unsupervised learning (clustering)
  - The class labels of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data

Prediction Problems: Classification vs. Numeric Prediction:

- Classification
  - predicts categorical class labels (discrete or nominal)
  - classifies data (constructs a model) based on the training set and the values (class labels) in a classifying attribute and uses it in classifying new data
- Numeric Prediction
  - models continuous-valued functions, i.e., predicts unknown or missing values

Classification—A Two-Step Process:
- Model construction: describing a set of predetermined classes
- Model usage: for classifying future or unknown objects

Comparing Attribute Selection Measures: The three measures, in general, return good results but

1. Information gain:
   - biased towards multivalued attributes
2. Gain ratio:
   - tends to prefer unbalanced splits in which one partition is much smaller than the others
3. Gini index:
   - biased to multivalued attributes
   - has difficulty when # of classes is large
   - tends to favor tests that result in equal-sized partitions and purity in both partitions

Overfitting: An induced tree may overfit the training data.

Two approaches to avoid overfitting
- Prepruning: Halt tree construction early-do not split a node if this would result in the goodness measure falling below a threshold
- Postpruning: Remove branches from a "fully grown" tree—get a sequence of progressively pruned trees

Why is decision tree induction popular?
- relatively faster learning speed (than other classification methods)
- convertible to simple and easy to understand classification rules
- can use SQL queries for accessing databases
- comparable classification accuracy with other methods

Methods for estimating a classifier's accuracy:
- Holdout method, random subsampling
- Cross-validation
- Bootstrap

Comparing classifiers:
- Confidence intervals
- Cost-benefit analysis and ROC Curves

Classifier Evaluation Metrics:
- Precision: exactness – what % of tuples that the classifier labeled as positive are actually positive
- Recall: completeness – what % of positive tuples did the classifier label as positive?
- F measure (F_1 or F-score): harmonic mean of precision and recall

Evaluating Classifier Accuracy:
- Holdout method: Given data is randomly partitioned into two independent sets(Training and Test set)
- Cross-validation: Randomly partition the data into k mutually exclusive subsets, each approximately equal size

Model Selection: ROC Curves: ROC (Receiver Operating Characteristics) curves: for visual comparison of classification models

Issues Affecting Model Selection:
- Accuracy: classifier accuracy: predicting class label
- Speed: time to construct the model, time to use the model.
- Robustness: handling noise and missing values
- Scalability: efficiency in disk-resident databases
- Interpretability: understanding and insight provided by the model

Ensemble methods: Use a combination of models to increase accuracy

Popular ensemble methods:
- Bagging: averaging the prediction over a collection of classifiers
- Boosting: weighted vote with a collection of classifiers
- Ensemble: combining a set of heterogeneous classifiers

Class-imbalance problem: Rare positive example but numerous negative ones, e.g., medical diagnosis, fraud, oil-spill, fault, etc.

Typical methods for imbalance data in 2-class classification:
- Oversampling: re-sampling of data from positive class
- Under-sampling: randomly eliminate tuples from negative class
- Threshold-moving: moves the decision threshold, t, so that the rare class tuples are easier to classify, and hence, less chance of costly false negative errors
- Ensemble techniques: Ensemble multiple classifiers introduced above

CH9:

Bayesian belief network (also known as Bayesian network, probabilistic network): allows class conditional independencies between subsets of variables

- Two components: (1) A directed acyclic graph (called a structure) and (2) a set of conditional probability tables (CPTs)

How Are Bayesian Networks Constructed?

- Subjective construction: Identification of (direct) causal structure
- Synthesis from other specifications
- Learning from data

Backpropagation: A neural network learning algorithm

A neural network: A set of connected input/output units where each connection has a weight associated with it

How A Multi-Layer Neural Network Works:

- The inputs to the network correspond to the attributes measured for each training tuple
- Inputs are fed simultaneously into the units making up the input layer
- They are then weighted and fed simultaneously to a hidden layer
- The number of hidden layers is arbitrary, although usually only one
- The weighted outputs of the last hidden layer are input to units making up the output layer, which emits the network's prediction
- The network is feed-forward: None of the weights cycles back to an input unit or to an output unit of a previous layer
- From a statistical point of view, networks perform nonlinear regression: Given enough hidden units and enough training samples, they can closely approximate any function

Backpropagation Steps

- Initialize weights to small random numbers, associated with biases
- Propagate the inputs forward (by applying activation function)
- Backpropagate the error (by updating weights and biases)
- Terminating condition (when error is very small, etc.)

Neural Network as a Classifier:

- Weakness
  - Long training time
  - Require a few parameters typically best determined empirically, e.g., the network topology or "structure."
  - Poor interpretability: Difficult to interpret the symbolic meaning behind the learned weights and of "hidden units" in the network
- Strength
  - High tolerance to noisy data
  - Ability to classify untrained patterns
  - Well-suited for continuous-valued inputs and outputs
  - Successful on an array of real-world data, e.g., hand-written letters
  - Algorithms are inherently parallel
  - Techniques have recently been developed for the extraction of rules from trained neural networks

Classification: predicts categorical class labels

Linear Classification

- Binary Classification problem
- Data above the red line belongs to class 'x'
- Data below red line belongs to class 'o'
- Examples: SVM, Perceptron, Probabilistic Classifiers

Discriminative Classifiers

- Advantages
  - Prediction accuracy is generally high
  - Robust, works when training examples contain errors
  - Fast evaluation of the learned target function
- Criticism
  - Long training time
  - Difficult to understand the learned function (weights)
  - Not easy to incorporate domain knowledge

Lazy vs. eager learning

- Lazy learning (e.g., instance-based learning): Simply stores training data (or only minor processing) and waits until it is given a test tuple
- Eager learning (the above discussed methods): Given a set of training tuples, constructs a classification model before receiving new (e.g., test) data to classify

Lazy Learner: Instance-Based Methods:
- k-nearest neighbor approach: Instances represented as points in a Euclidean space.
- Locally weighted regression: Constructs local approximation
- Case-based reasoning: Uses symbolic representations and knowledge-based inference

Case-Based Reasoning (CBR): Uses a database of problem solutions to solve new problems

CH10

Cluster: A collection of data objects (Unsupervised learning).
Cluster analysis: Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
Applications of Cluster Analysis:
- Data reduction
  - Summarization: Preprocessing for regression, PCA, classification, and association analysis
  - Compression: Image processing: vector quantization
- Hypothesis generation and testing
- Prediction based on groups
- Finding K-nearest Neighbors
- Outlier detection

Clustering: Application Examples: Biology, Information retrieval, Marketing, City-planning.
Basic Steps to Develop a Clustering Task:
- Feature selection
- Proximity measure
- Clustering criterion
- Clustering algorithms
- Validation of the results
- Interpretation of the results

Quality: What Is Good Clustering?
- A good clustering method will produce high quality clusters
  - high intra-class similarity: cohesive within clusters
  - low inter-class similarity: distinctive between clusters

The quality of a clustering method depends on
  - the similarity measure used by the method
  - its implementation, and
  - Its ability to discover some or all of the hidden patterns

Considerations for Cluster Analysis:
- Partitioning criteria
- Separation of clusters
- Similarity measure
- Clustering space

Requirements and Challenges for Cluster:
- Scalability
- Ability to deal with different types of attributes
- Constraint-based clustering
- Interpretability and usability

Major Clustering Approaches:
- Partitioning approach: Typical methods: k-means, k-medoids, CLARANS
- Hierarchical approach: Typical methods: Diana, Agnes, BIRCH, CAMELEON
- Density-based approach: Typical methods: DBSACN, OPTICS, DenClue
- Grid-based approach: Typical methods: STING, WaveCluster, CLIQUE
- Model-based: Typical methods: EM, SOM, COBWEB
- Frequent pattern-based: Typical methods: p-Cluster
- User-guided or constraint-based: Typical methods: COD (obstacles), constrained clustering
- Link-based clustering

Partitioning method: Partitioning a database D of n objects into a set of k clusters, such that the sum of squared distances is minimized (where c_i is the centroid or medoid of cluster C_i)
k-means: Each cluster is represented by the center of the cluster
k-medoids or PAM (Partition around medoids): Each cluster is represented by one of the objects in the cluster

K-Means Method:
Strength: Efficient
Weakness:
- Applicable only to objects in a continuous n-dimensional space
- Need to specify k, the number of clusters, in advance
- Sensitive to noisy data and outliers
- Not suitable to discover clusters with non-convex shapes

What Is the Problem of the K-Means Method?
- The k-means algorithm is sensitive to outliers !

AGNES (Agglomerative Nesting)
DIANA (Divisive Analysis)
Distance between Clusters:
- Single link: smallest distance between an element in one cluster and an element in the other
- Complete link: largest distance between an element in one cluster and an element in the other
- Average: avg distance between an element in one cluster and an element in the other
- Centroid: distance between the centroids of two clusters
- Medoid: distance between the medoids of two clusters

Major features OF Density-Based Clustering Methods:
- Discover clusters of arbitrary shape
- Handle noise
- One scan
- Need density parameters as termination condition

Two parameters of Density-Based Clustering:
- Eps: Maximum radius of the neighbourhood
- MinPts: Minimum number of points in an Eps-neighbourhood of that point

Density-Reachable and Density-Connected:
- Density-reachable: A point p is density-reachable from a point q w.r.t. Eps, MinPts if there is a chain of points is directly density-reachable from
- Density-connected: A point p is density-connected to a point q w.r.t. Eps, MinPts if there is a point o such that both, p and q are density-reachable from o w.r.t. Eps and MinPts

DBSCAN: Density-Based Spatial Clustering of Applications with Noise
Grid-Based Clustering Method: Using multi-resolution grid data structure
- STING (a STatistical INformation Grid approach)
- CLIQUE:Both grid-based and subspace clustering
- WaveCluster: A multi-resolution clustering approach using wavelet method

STING Algorithm and Its Analysis:
- Advantages:
  - Query-independent, easy to parallelize, incremental update
  - O(K), where K is the number of grid cells at the lowest level
- Disadvantages:
  - All the cluster boundaries are either horizontal or vertical, and no diagonal boundary is detected

Determine the Number of Clusters:
- Empirical method
- Elbow method
- Cross validation method:
  - Divide a given data set into m parts
  - Use m – 1 parts to obtain a clustering model
  - Use the remaining part to test the quality of the clustering
  - For any k > 0, repeat it m times, compare the overall quality measure w.r.t. different k's, and find # of clusters that fits the data the best

Measuring Clustering Quality:
- External: supervised, employ criteria not inherent to the dataset
- Internal: unsupervised, criteria derived from data itself
- Relative: directly compare different clusterings, usually those obtained via different parameter settings for the same algorithm

Some Commonly Used External Measures:

- Matching-based measures
- Entropy-Based Measures
- Pair-wise measures
- Correlation measures

The k-means algorithm has two steps at each iteration:

- Expectation Step (E-step): Given the current cluster centers, each object is assigned to the cluster whose center is closest to the object: An object is expected to belong to the closest cluster
- Maximization Step (M-step): Given the cluster assignment, for each cluster, the algorithm adjusts the center so that the sum of distance from the objects assigned to this cluster and the new center is minimized

The EM (Expectation Maximization) Algorithm: A framework to approach maximum likelihood or maximum a posteriori estimates of parameters in statistical models.

- E-step assigns objects to clusters according to the current fuzzy clustering or parameters of probabilistic clusters
- M-step finds the new clustering or parameters that maximize the sum of squared error (SSE) or the expected likelihood

Advantages and Disadvantages of Mixture Models:

- Strength
  - Mixture models are more general than partitioning and fuzzy clustering
  - Clusters can be characterized by a small number of parameters
  - The results may satisfy the statistical assumptions of the generative models
- Weakness
  - Converge to local optimal (overcome: run multi-times w. random initialization)
  - Computationally expensive if the number of distributions is large
  - Need large data sets
  - Hard to estimate the number of clusters

Clustering High-Dimensional Data: Clustering high-dimensional data

- Major challenges:
  - Many irrelevant dimensions may mask clusters
  - Distance measure becomes meaningless—due to equi-distance
  - Clusters may exist only in some subspaces
- Methods
  - Subspace-clustering: Search for clusters existing in subspaces of the given high dimensional data space
  - Dimensionality reduction approaches: Construct a much lower dimensional space and search for clusters

Why Traditional Distance Measures May Not Be Effective on High-D Data?

- Traditional distance measure could be dominated by noises in many dimensions
- Clustering should not only consider dimensions but also attributes (features)

The Curse of Dimensionality:

- Data in only one dimension is relatively packed
- Adding a dimension "stretch" the points across that dimension, making them further apart
- Adding more dimensions will make the points further apart—high dimensional data is extremely sparse
- Distance measure becomes meaningless—due to equi-distance

Subspace Clustering Methods:

- Subspace search methods: Search various subspaces to find clusters
  - Bottom-up approaches
  - Top-down approaches
- Correlation-based clustering methods
  - E.g., PCA based approaches
- Bi-clustering methods
  - Optimization-based methods
  - Enumeration methods

Graph clustering methods

  - Minimum cuts: FastModularity
  - Density-based clustering: SCAN

Graph Clustering: Challenges of Finding Good Cuts:

- High computational cost
- Sophisticated graphs
- High dimensionality
- Sparsity

Two approaches for clustering graph data
- Use generic clustering methods for high-dimensional data
- Designed specifically for clustering graphs

A Social Network Model:
- Individuals in a tight social group, or clique, know many of the same people, regardless of the size of group
- Individuals who are hubs know many people in different groups but belong to no single group. Politicians, for example bridge multiple groups
- Individuals who are outliers reside at the margins of society. Hermits, for example, know few people and belong to no group

Why Constraint-Based Cluster Analysis?
- Need user feedback: Users know their applications the best
- Less parameters but more user-desired constraints, e.g., an ATM allocation problem: obstacle & desired clusters

Categorization of Constraints:
- Constraints on instances: specifies how a pair or a set of instances should be grouped in the cluster analysis
- Constraints on clusters: specifies a requirement on the clusters
- Constraints on similarity measurements: specifies a requirement that the similarity calculation must respect
- Issues: Hard vs. soft constraints; conflicting or redundant constraints
- Handling hard constraints: Strictly respect the constraints in cluster assignments

Constraint-Based Clustering Methods:
- Handling hard constraints: Strictly respect the constraints in cluster assignments
  - Example: The COP-k-means algorithm
  - Generate super-instances for must-link constraints
  - Conduct modified k-means clustering to respect cannot-link constraints
- Handling Soft Constraints:
  - Treated as an optimization problem: When a clustering violates a soft constraint, a penalty is imposed on the clustering
  - Overall objective: Optimizing the clustering quality, and minimizing the constraint violation penalty

Objective function: Sum of distance used in k-means, adjusted by the constraint violation penalties
- Penalty of a must-link violation
  - If objects x and y must-be-linked but they are assigned to two different centers, c_1 and c_2, dist(c_1, c_2) is added to the objective function as the penalty
- Penalty of a cannot-link violation
  - If objects x and y cannot-be-linked but they are assigned to a common center c, dist(c, c'), between c and c' is added to the objective function as the penalty, where c' is the closest cluster to c that can accommodate x or y

User-specified feature (in the form of attribute) is used as a hint, not class labels
Semi-supervised clustering: User provides a training set consisting of "similar" ("must-link) and "dissimilar" pairs of objects
User-guided clustering: User specifies an attribute as a hint, and more relevant features are found for clustering

Why Not Semi-Supervised Clustering?
- Much information (in multiple relations) is needed to judge whether two tuples are similar
- A user may not be able to provide a good training set
- It is much easier for a user to specify an attribute as a hint, such as a student's research area

CH12

Outlier: A data object that deviates significantly from the normal objects as if it were generated by a different mechanism
- Ex.: Unusual credit card purchase, sports: Michael Jordon, Wayne Gretzky, ...

Outliers are different from the noise data
- Noise is random error or variance in a measured variable
- Noise should be removed before outlier detection

Outliers Applications:
- Credit card fraud detection
- Telecom fraud detection
- Customer segmentation
- Medical analysis

Types of Outliers:
- Global outlier: Object is O_g if it significantly deviates from the rest of the data set
- Contextual outlier: Object is O_c if it deviates significantly based on a selected context
- Collective Outliers: A subset of data objects *collectively* deviate significantly from the whole data set, even if the individual data objects may not be outliers

Challenges of Outlier Detection:

- Modeling normal objects and outliers properly
- Application-specific outlier detection
- Handling noise in outlier detection
- Understandability

Outlier Detection:

- Supervised Methods
- Unsupervised Methods
- Semi-Supervised Methods
- Statistical methods (also known as model-based methods) assume that the normal data follow some statistical model (a stochastic model)
  - The data not following the model are outliers.
- Proximity-Based Methods: An object is an outlier if the nearest neighbors of the object are far away, i.e., the proximity of the object is significantly deviates from the proximity of most of the other objects in the same data set
- Clustering-Based Methods: Normal data belong to large and dense clusters, whereas outliers belong to small or sparse clusters, or do not belong to any clusters

Statistical Approaches:

- Parametric method
  - Assumes that the normal data is generated by a parametric distribution with parameter
  - The probability density function of the parametric distribution f(x, θ) gives the probability that object x is generated by the distribution
  - The smaller this value, the more likely x is an outlier
- Non-parametric method
  - Not assume an a-priori statistical model and determine the model from the input data
  - Not completely parameter free but consider the number and nature of the parameters are flexible and not fixed in advance
  - Examples: histogram and kernel density estimation

Intuition: Objects that are far away from the others are outliers

Assumption of proximity-based approach: The proximity of an outlier deviates significantly from that of most of the others in the data set

Two types of proximity-based outlier detection methods

- Distance-based outlier detection: An object o is an outlier if its neighborhood does not have enough other points
- Density-based outlier detection: An object o is an outlier if its density is relatively much lower than that of its neighbors

Clustering-Based Method: Strength and Weakness:

- Strength
  - Detect outliers without requiring any labeled data
  - Work for many types of data
  - Clusters can be regarded as summaries of the data
  - Once the cluster are obtained, need only compare any object against the clusters to determine whether it is an outlier (fast)
- Weakness
  - Effectiveness depends highly on the clustering method used—they may not be optimized for outlier detection
  - High computational cost: Need to first find clusters
  - A method to reduce the cost: Fixed-width clustering

CH13:

Major Statistical Data Mining Methods:

- Regression: predict the value of a response (dependent) variable from one or more predictor (independent) variables where the variables are numeric
- Generalized Linear Model: allow a categorical response variable (or some transformation of it) to be related to a set of predictor variables
- Analysis of Variance: Analyze experimental data for two or more populations described by a numeric response variable and one or more categorical variables (factors)
- Mixed-Effect Models: For analyzing grouped data, i.e. data that can be classified according to one or more grouping variables
- Factor Analysis: determine which variables are combined to generate a given factor
- Discriminant Analysis: predict a categorical response variable, commonly used in social science
- Survival Analysis: Predicts the probability that a patient undergoing a medical treatment would survive at least to time t (life span prediction)

Views on Data Mining Foundations:

- Data reduction
- Data compression
- Probability and statistical theory
- Microeconomic view
- Pattern Discovery and Inductive databases

Visual Data Mining:

- Visualization: Use of computer graphics to create visual images which aid in the understanding of complex, often massive representations of data
- Visual Data Mining: discovering implicit but useful knowledge from large data sets using visualization techniques

Purpose of Visualization:

- Gain insight
- Provide qualitative overview
- Search
- Help find interesting regions and suitable parameters
- Provide a visual proof

Examples of Data Mining Result Visualization:

- Scatter plots and boxplots (obtained from descriptive data mining)
- Decision trees
- Association rules
- Clusters
- Outliers
- Generalized rules

Interactive Visual Data Mining: Using visualization tools in the data mining process to help users make smart data mining decisions

- Example: Display the data distribution in a set of attributes using colored sectors or columns

Data Mining Applications:

- Data Mining for Financial data analysis
  - Ex: View the debt and revenue changes by month, by region, by sector, and by other factors
- Data Mining for Retail and Telecommunication Industries
  - Identify customer buying behaviors
  - Discover customer shopping patterns and trends
- Data Mining in Science and Engineering
- Data Mining for Intrusion Detection and Prevention
- Data Mining and Recommender Systems
  - Content-based: Recommends items that are similar to items the user preferred or queried in the past
  - Collaborative filtering: Consider a user's social environment, opinions of other customers who have similar tastes or preferences

Majority of intrusion detection and prevention systems use

- Signature-based detection: use signatures, attack patterns that are preconfigured and predetermined by domain experts
- Anomaly-based detection: build profiles (models of normal behavior) and detect those that are substantially deviate from the profiles

Ubiquitous Data Mining: Data mining is used everywhere, e.g., online shopping
Invisible Data Mining: Data mining functions are built in daily life operations
Privacy, Security and Social Impacts of Data Mining:

- The real privacy concern: unconstrained access of individual records, especially privacy-sensitive information
  - Method 1: Removing sensitive IDs associated with the data
  - Method 2: Data security-enhancing methods
    - Multi-level security model: permit to access to only authorized level
    - Encryption
  - Method 3: Privacy-preserving data mining methods

Trends of Data Mining:

- Application exploration: Dealing with application-specific problems
- Scalable and interactive data mining methods
- Integration of data mining with Web search engines, database systems, data warehouse systems and cloud computing systems
- Mining social and information networks
- Mining spatiotemporal, moving objects and cyber-physical systems
- Mining multimedia, text and web data
- Mining biological and biomedical data
- Data mining with software engineering and system engineering
- Visual and audio data mining
- Distributed data mining and real-time data stream mining
- Privacy protection and information security in data mining