

DATA MINING/IT0467

UNIT-I

An Introduction on Data Mining and Preprocessing

Chapter 1. Introduction

- Motivation: Why data mining?
- What is data mining?
- Data Mining: On what kind of data?
- Data mining functionality
- Classification of data mining systems
- Top-10 most popular data mining algorithms
- Major issues in data mining
- Overview of the course

Why Data Mining?

- The Explosive Growth of Data: from terabytes to petabytes
 - Data collection and data availability
 - Automated data collection tools, database systems, Web, computerized society
 - Major sources of abundant data
 - Business: Web, e-commerce, transactions, stocks, ...
 - Science: Remote sensing, bioinformatics, scientific simulation, ...
 - Society and everyone: news, digital cameras, YouTube
- We are drowning in data, but starving for knowledge!
- “Necessity is the mother of invention”—Data mining—Automated analysis of massive data sets

What Is Data Mining?

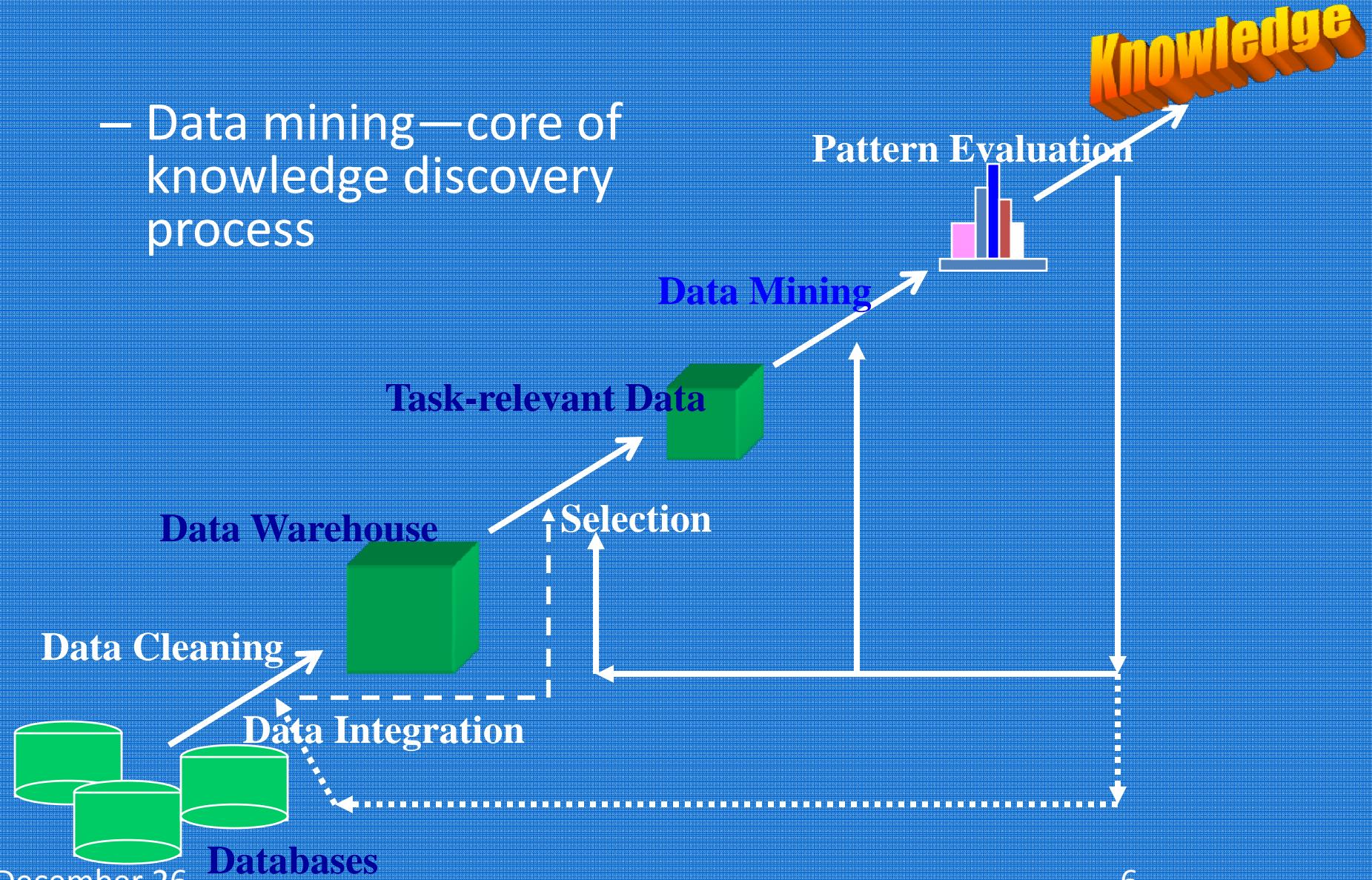


- Data mining (knowledge discovery from data)
 - Extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data
 - Data mining: a misnomer?
- Alternative names
 - Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archeology, data dredging, information harvesting, business intelligence, etc.
- Watch out: Is everything “data mining”?
 - Simple search and query processing
 - (Deductive) expert systems

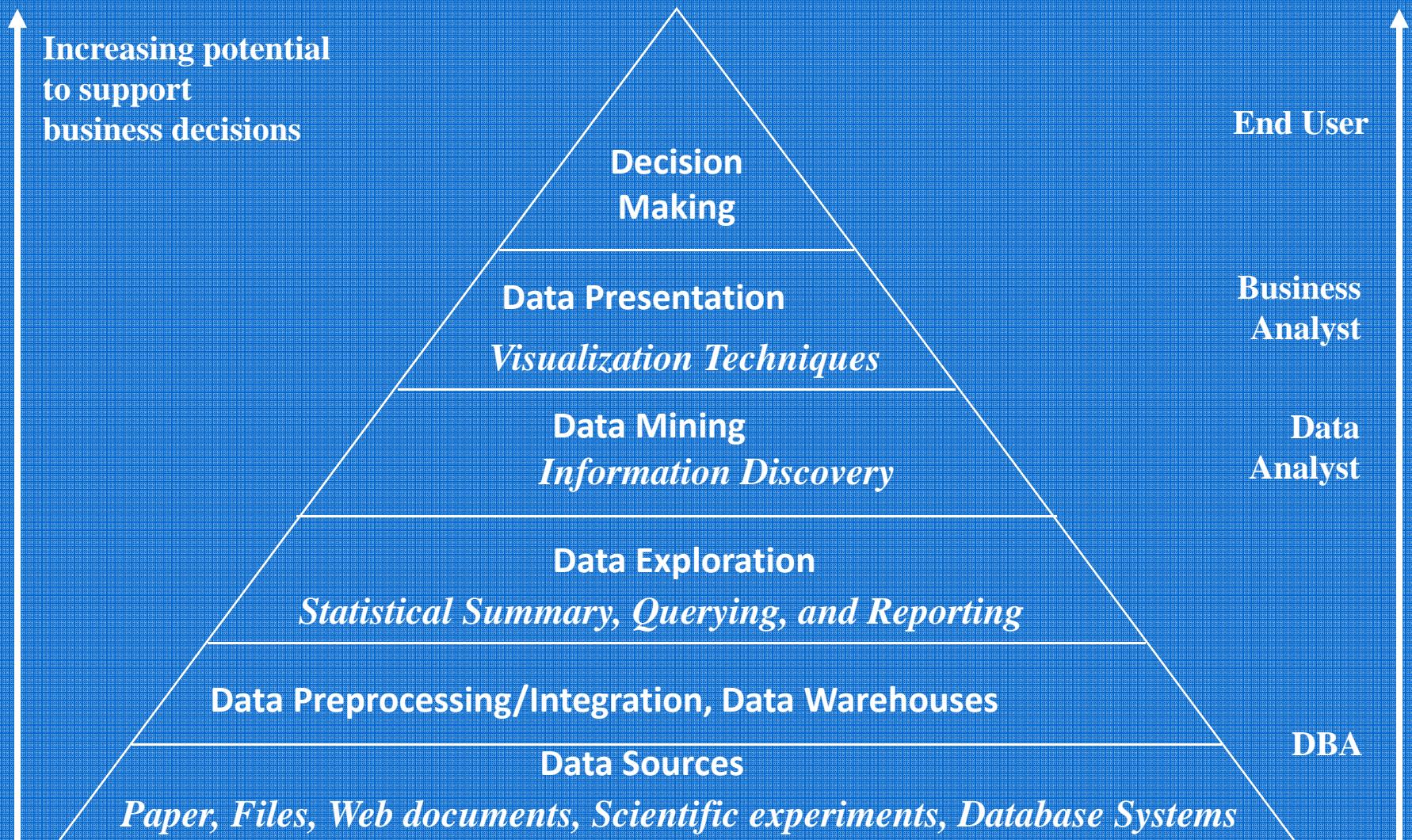


Knowledge Discovery (KDD) Process

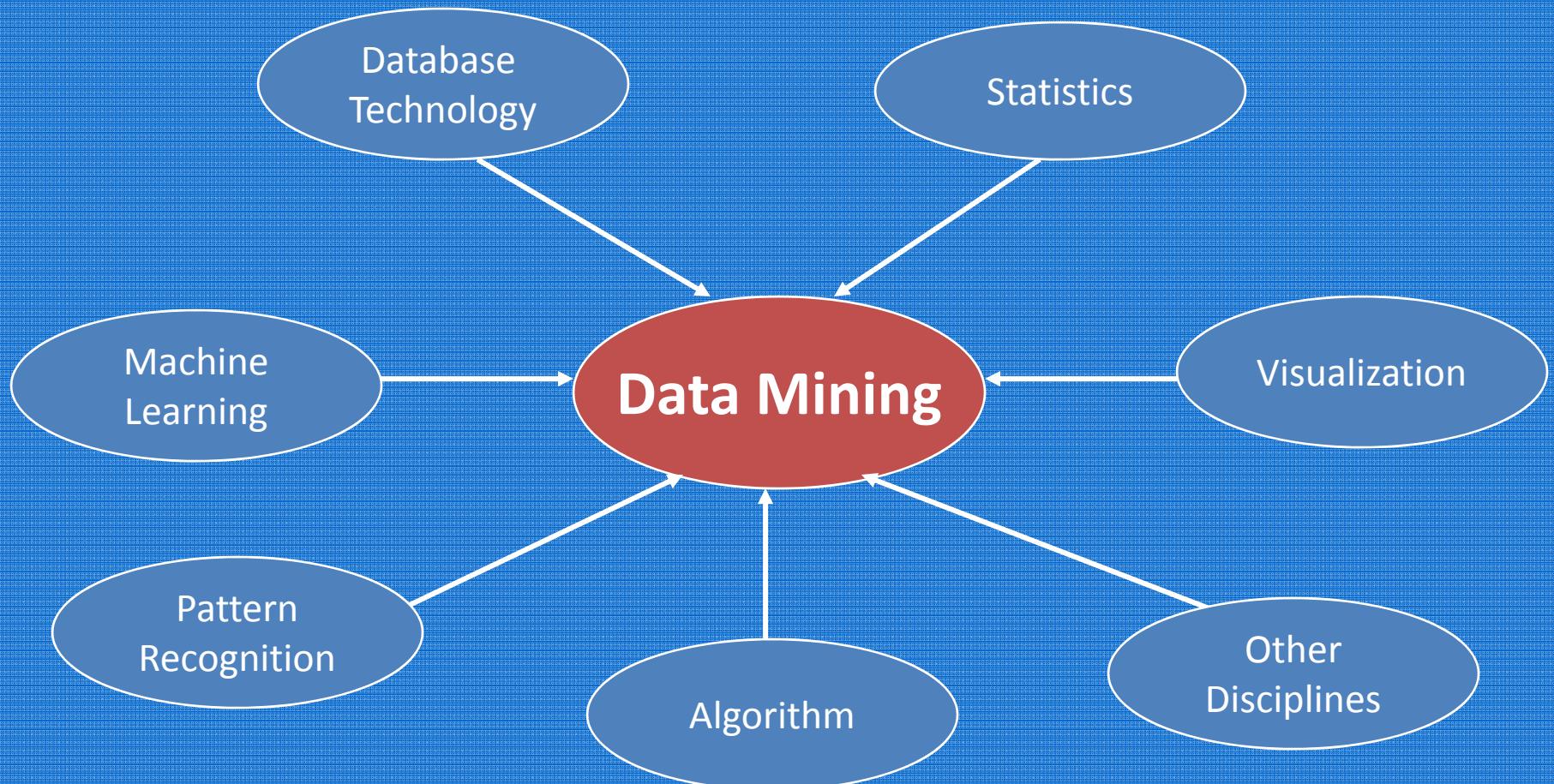
- Data mining—core of knowledge discovery process



Data Mining and Business Intelligence



Data Mining: Confluence of Multiple Disciplines



Why Not Traditional Data Analysis?

- Tremendous amount of data
 - Algorithms must be highly scalable to handle such as tera-bytes of data
- High-dimensionality of data
 - Micro-array may have tens of thousands of dimensions
- High complexity of data
 - Data streams and sensor data
 - Time-series data, temporal data, sequence data
 - Structure data, graphs, social networks and multi-linked data
 - Heterogeneous databases and legacy databases
 - Spatial, spatiotemporal, multimedia, text and Web data
 - Software programs, scientific simulations
- New and sophisticated applications

Multi-Dimensional View of Data Mining

- **Data to be mined**
 - Relational, data warehouse, transactional, stream, object-oriented/relational, active, spatial, time-series, text, multi-media, heterogeneous, legacy, WWW
- **Knowledge to be mined**
 - Characterization, discrimination, association, classification, clustering, trend/deviation, outlier analysis, etc.
 - Multiple/integrated functions and mining at multiple levels
- **Techniques utilized**
 - Database-oriented, data warehouse (OLAP), machine learning, statistics, visualization, etc.
- **Applications adapted**
 - Retail, telecommunication, banking, fraud analysis, bio-data mining, stock market analysis, text mining, Web mining, etc.

Data Mining: Classification Schemes

- General functionality
 - Descriptive data mining
 - Predictive data mining
- Different views lead to different classifications
 - Data view: Kinds of data to be mined
 - Knowledge view: Kinds of knowledge to be discovered
 - Method view: Kinds of techniques utilized
 - Application view: Kinds of applications adapted

Data Mining: On What Kinds of Data?

- Database-oriented data sets and applications
 - Relational database, data warehouse, transactional database
- Advanced data sets and advanced applications
 - Data streams and sensor data
 - Time-series data, temporal data, sequence data (incl. bio-sequences)
 - Structure data, graphs, social networks and multi-linked data
 - Object-relational databases
 - Heterogeneous databases and legacy databases
 - Spatial data and spatiotemporal data
 - Multimedia database
 - Text databases
 - The World-Wide Web

Data Mining Functionalities

- Multidimensional concept description: Characterization and discrimination
 - Generalize, summarize, and contrast data characteristics, e.g., dry vs. wet regions
- Frequent patterns, association, correlation vs. causality
 - Diaper → Beer [0.5%, 75%] (Correlation or causality?)
- Classification and prediction
 - Construct models (functions) that describe and distinguish classes or concepts for future prediction
 - E.g., classify countries based on (climate), or classify cars based on (gas mileage)
 - Predict some unknown or missing numerical values

Data Mining Functionalities (2)

- Cluster analysis
 - Class label is unknown: Group data to form new classes, e.g., cluster houses to find distribution patterns
 - Maximizing intra-class similarity & minimizing interclass similarity
- Outlier analysis
 - Outlier: Data object that does not comply with the general behavior of the data
 - Noise or exception? Useful in fraud detection, rare events analysis
- Trend and evolution analysis
 - Trend and deviation: e.g., regression analysis
 - Sequential pattern mining: e.g., digital camera → large SD memory
 - Periodicity analysis
 - Similarity-based analysis
- Other pattern-directed or statistical analyses

Major Issues in Data Mining

- Mining methodology
 - Mining different kinds of knowledge from diverse data types, e.g., bio, stream, Web
 - Performance: efficiency, effectiveness, and scalability
 - Pattern evaluation: the interestingness problem
 - Incorporation of background knowledge
 - Handling noise and incomplete data
 - Parallel, distributed and incremental mining methods
 - Integration of the discovered knowledge with existing one: knowledge fusion
- User interaction
 - Data mining query languages and ad-hoc mining
 - Expression and visualization of data mining results
 - Interactive mining of knowledge at multiple levels of abstraction
- Applications and social impacts
 - Domain-specific data mining & invisible data mining
 - Protection of data security, integrity, and privacy

Why Data Mining Query Language?

- Automated vs. query-driven?
 - Finding all the patterns autonomously in a database?—unrealistic because the patterns could be too many but uninteresting
- Data mining should be an interactive process
 - User directs what to be mined
- Users must be provided with a set of **primitives** to be used to communicate with the data mining system
- Incorporating these primitives in a **data mining query language**
 - More flexible user interaction
 - Foundation for design of graphical user interface
 - Standardization of data mining industry and practice

Primitives that Define a Data Mining Task

- Task-relevant data
 - Database or data warehouse name
 - Database tables or data warehouse cubes
 - Condition for data selection
 - Relevant attributes or dimensions
 - Data grouping criteria
- Type of knowledge to be mined
 - Characterization, discrimination, association, classification, prediction, clustering, outlier analysis, other data mining tasks
- Background knowledge
- Pattern interestingness measurements
- Visualization/presentation of discovered patterns

DMQL—A Data Mining Query Language

- Motivation
 - A DMQL can provide the ability to support ad-hoc and interactive data mining
 - By providing a standardized language like SQL
 - Hope to achieve a similar effect like that SQL has on relational database
 - Foundation for system development and evolution
 - Facilitate information exchange, technology transfer, commercialization and wide acceptance
- Design
 - DMQL is designed with the primitives described earlier

An Example Query in DMQL

Example 1.11 Mining classification rules. Suppose, as a marketing manager of *AllElectronics*, you would like to classify customers based on their buying patterns. You are especially interested in those customers whose salary is no less than \$40,000, and who have bought more than \$1,000 worth of items, each of which is priced at no less than \$100. In particular, you are interested in the customer's age, income, the types of items purchased, the purchase location, and where the items were made. You would like to view the resulting classification in the form of rules. This data mining query is expressed in DMQL³ as follows, where each line of the query has been enumerated to aid in our discussion.

```
use database AllElectronics_db
use hierarchy location_hierarchy for T.branch, age_hierarchy for C.age
mine classification as promising_customers
in relevance to C.age, C.income, I.type, I.place_made, T.branch
from customer C, item I, transaction T
where I.item_ID = T.item_ID and C.cust_ID = T.cust_ID
      and C.income ≥ 40,000 and I.price ≥ 100
group by T.cust_ID
having sum(I.price) ≥ 1,000
display as rules
```

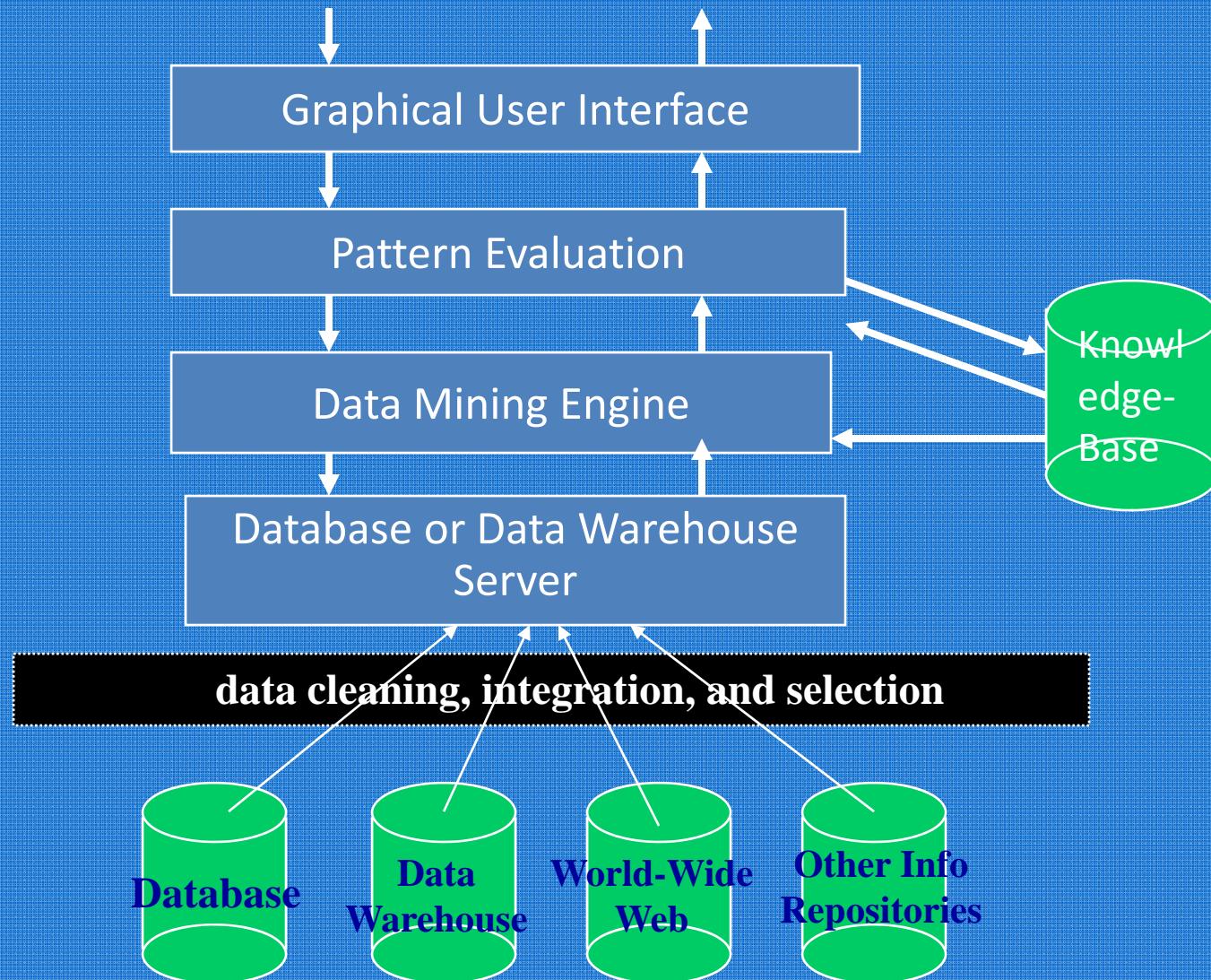
Integration of Data Mining and Data Warehousing

- **Data mining systems, DBMS, Data warehouse systems coupling**
 - No coupling, loose-coupling, semi-tight-coupling, tight-coupling
- **On-line analytical mining data**
 - integration of mining and OLAP technologies
- **Interactive mining multi-level knowledge**
 - Necessity of mining knowledge and patterns at different levels of abstraction by drilling/rolling, pivoting, slicing/dicing, etc.
- **Integration of multiple mining functions**
 - Characterized classification, first clustering and then association

Coupling Data Mining with DB/DW Systems

- No coupling—flat file processing, not recommended
- Loose coupling
 - Fetching data from DB/DW
- Semi-tight coupling—enhanced DM performance
 - Provide efficient implement a few data mining primitives in a DB/DW system, e.g., sorting, indexing, aggregation, histogram analysis, multiway join, precomputation of some stat functions
- Tight coupling—A uniform information processing environment
 - DM is smoothly integrated into a DB/DW system, mining query is optimized based on mining query, indexing, query processing methods, etc.

Architecture: Typical Data Mining System



Chapter-DATA Preprocessing

- Why preprocess the data?
- Descriptive data summarization
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Why Data Preprocessing?

- Data in the real world is dirty
 - incomplete: lacking attribute values, lacking 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

Why Is Data Dirty?

- Incomplete data may come from
 - “Not applicable” data value when collected
 - Different considerations between the time when the data was collected and when it is analyzed.
 - Human/hardware/software problems
- Noisy data (incorrect values) may come from
 - Faulty data collection instruments
 - Human or computer error at data entry
 - Errors in data transmission
- Inconsistent data may come from
 - Different data sources
 - Functional dependency violation (e.g., modify some linked data)
- Duplicate records also need data cleaning

Why Is Data Preprocessing Important?

- No quality data, no quality mining results!
 - Quality decisions must be based on quality data
 - e.g., duplicate or missing data may cause incorrect or even misleading statistics.
 - Data warehouse needs consistent integration of quality data
- Data extraction, cleaning, and transformation comprises the majority of the work of building a data warehouse

Multi-Dimensional Measure of Data Quality

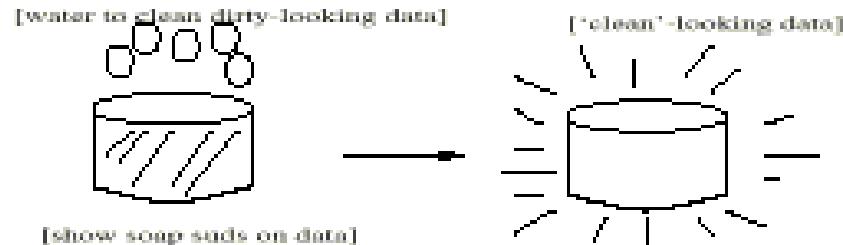
- A well-accepted multidimensional view:
 - Accuracy
 - Completeness
 - Consistency
 - Timeliness
 - Believability
 - Value added
 - Interpretability
 - Accessibility
- Broad categories:
 - Intrinsic, contextual, representational, and accessibility

Major Tasks in Data Preprocessing

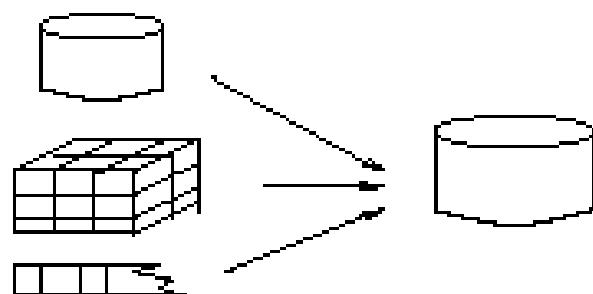
- Data cleaning
 - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- Data integration
 - Integration of multiple databases, data cubes, or files
- Data transformation
 - Normalization and aggregation
- Data reduction
 - Obtains reduced representation in volume but produces the same or similar analytical results
- Data discretization
 - Part of data reduction but with particular importance, especially for numerical data

Forms of Data Preprocessing

Data Cleaning



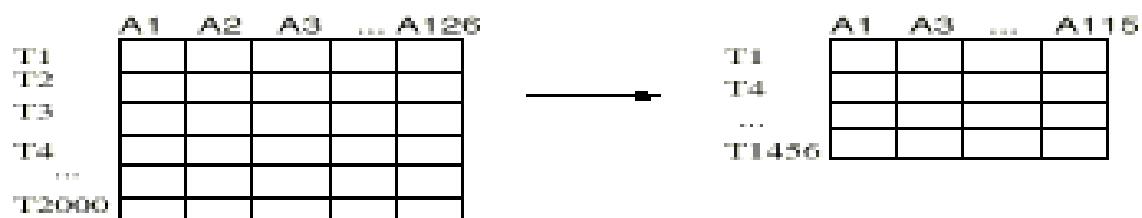
Data Integration



Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction



Data Preprocessing

- Why preprocess the data?
- Descriptive data summarization
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Mining Data Descriptive Characteristics

- Motivation
 - To better understand the data: central tendency, variation and spread
- Data dispersion characteristics
 - median, max, min, quantiles, outliers, variance, etc.
- Numerical dimensions correspond to sorted intervals
 - Data dispersion: analyzed with multiple granularities of precision
 - Boxplot or quantile analysis on sorted intervals
- Dispersion analysis on computed measures
 - Folding measures into numerical dimensions
 - Boxplot or quantile analysis on the transformed cube

Measuring the Central Tendency

- Mean (algebraic measure) (sample vs. population):

- Weighted arithmetic mean:
- Trimmed mean: chopping extreme values

- Median: A holistic measure

- Middle value if odd number of values, or average of the middle two values otherwise
- Estimated by interpolation (for *grouped data*):

- Mode

- Value that occurs most frequently in the data
- Unimodal, bimodal, trimodal
- Empirical formula:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad \mu = \frac{\sum x}{N}$$

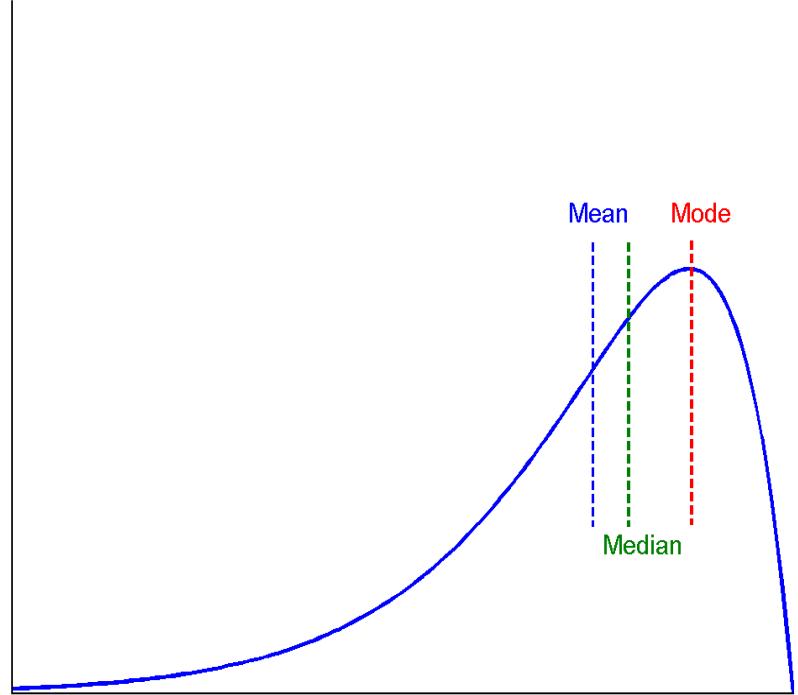
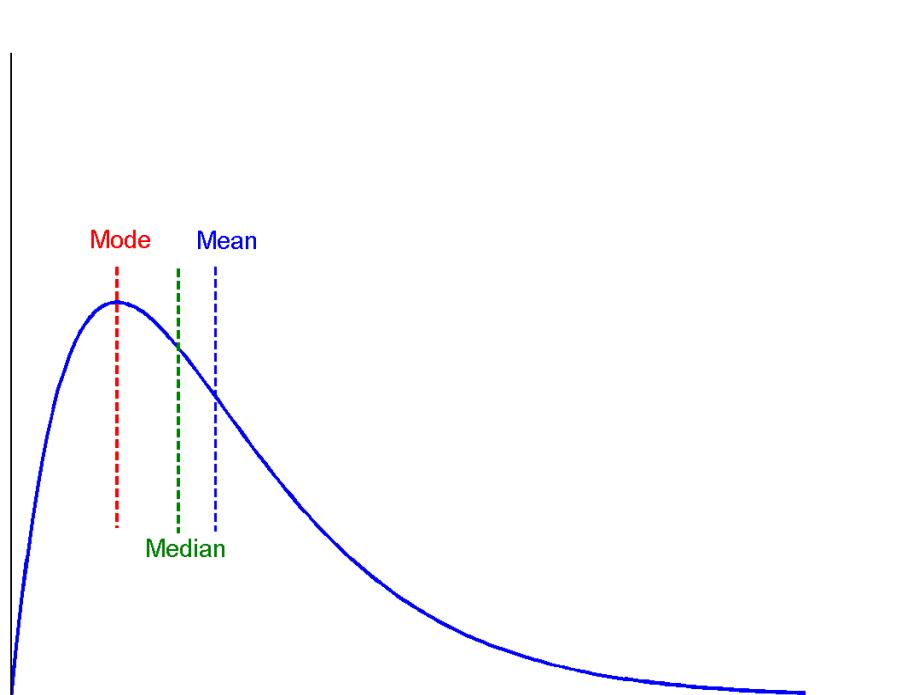
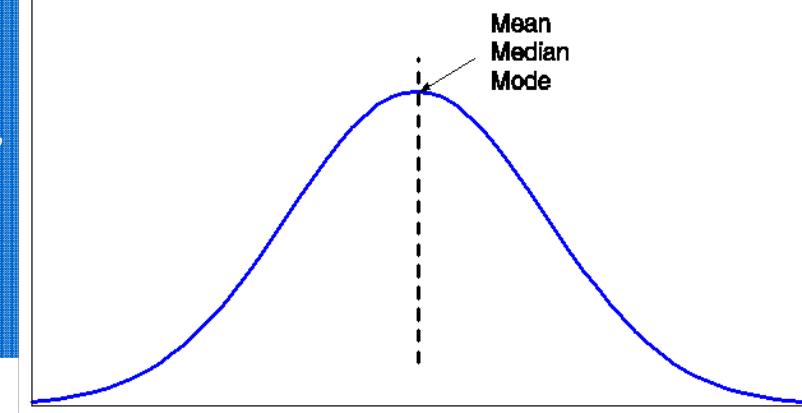
$$\bar{x} = \frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i}$$

$$median = L_1 + \left(\frac{n/2 - (\sum f)l}{f_{median}} \right) c$$

$$mean - mode = 3 \times (mean - median)$$

Symmetric vs. Skewed Data

- Median, mean and mode of symmetric, positively and negatively skewed data



Measuring the Dispersion of Data

- Quartiles, outliers and boxplots
 - Quartiles: Q_1 (25^{th} percentile), Q_3 (75^{th} percentile)
 - Inter-quartile range: $\text{IQR} = Q_3 - Q_1$
 - Five number summary: min, Q_1 , M, Q_3 , max
 - Boxplot: ends of the box are the quartiles, median is marked, whiskers, and plot outlier individually
 - Outlier: usually, a value higher/lower than $1.5 \times \text{IQR}$
- Variance and standard deviation (*sample: s, population: σ*)
 - Variance: (algebraic, scalable computation)

$$s^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 = \frac{1}{n-1} \left[\sum_{i=1}^n x_i^2 - \frac{1}{n} (\sum_{i=1}^n x_i)^2 \right]$$

Standard deviation s (or σ) is the square root of variance s^2 (or σ^2)

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^n (x_i - \mu)^2 = \frac{1}{N} \sum_{i=1}^n x_i^2 - \mu^2$$

Data Preprocessing

- Why preprocess the data?
- Descriptive data summarization
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Cleaning

- Importance
 - “Data cleaning is one of the three biggest problems in data warehousing”—Ralph Kimball
 - “Data cleaning is the number one problem in data warehousing”—DCI survey
- Data cleaning tasks
 - Fill in missing values
 - Identify outliers and smooth out noisy data
 - Correct inconsistent data
 - Resolve redundancy caused by data integration

Missing Data

- Data is not always available
 - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
 - equipment malfunction
 - inconsistent with other recorded data and thus deleted
 - data not entered due to misunderstanding
 - certain data may not be considered important at the time of entry
 - not register history or changes of the data
- Missing data may need to be inferred.

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably.)
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
 - a global constant : e.g., “unknown”, a new class?!
 - the attribute mean
 - the attribute mean for all samples belonging to the same class: smarter
 - the most probable value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may due to
 - faulty data collection instruments
 - data entry problems
 - data transmission problems
 - technology limitation
 - inconsistency in naming convention
- Other data problems which requires data cleaning
 - duplicate records
 - incomplete data
 - inconsistent data

How to Handle Noisy Data?

- **Binning**
 - first sort data and partition into (equal-frequency) bins
 - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- **Regression**
 - smooth by fitting the data into regression functions
- **Clustering**
 - detect and remove outliers
- **Combined computer and human inspection**
 - detect suspicious values and check by human (e.g., deal with possible outliers)

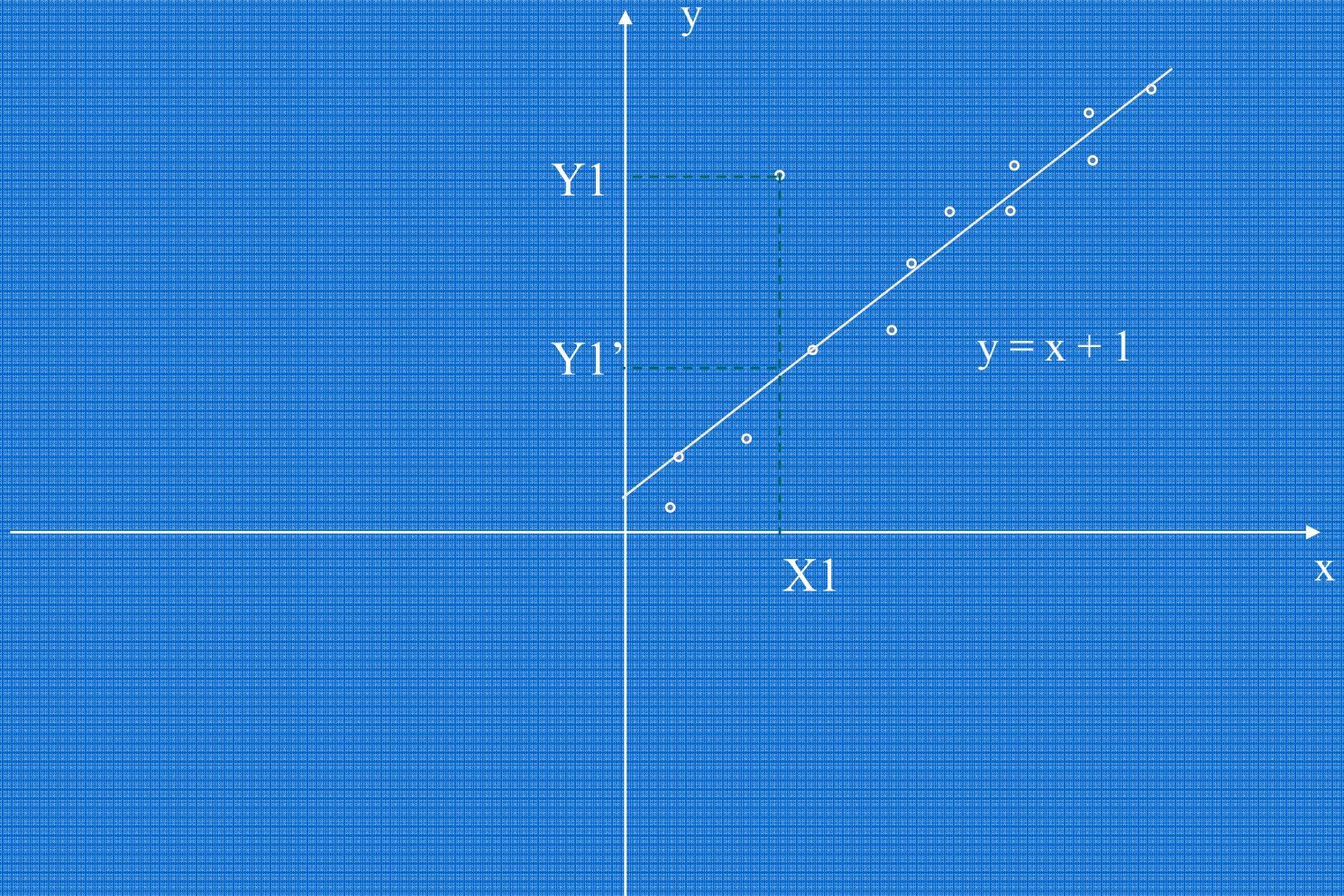
Simple Discretization Methods: Binning

- Equal-width (distance) partitioning
 - Divides the range into N intervals of equal size: uniform grid
 - if A and B are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$.
 - The most straightforward, but outliers may dominate presentation
 - Skewed data is not handled well
- Equal-depth (frequency) partitioning
 - Divides the range into N intervals, each containing approximately same number of samples
 - Good data scaling
 - Managing categorical attributes can be tricky

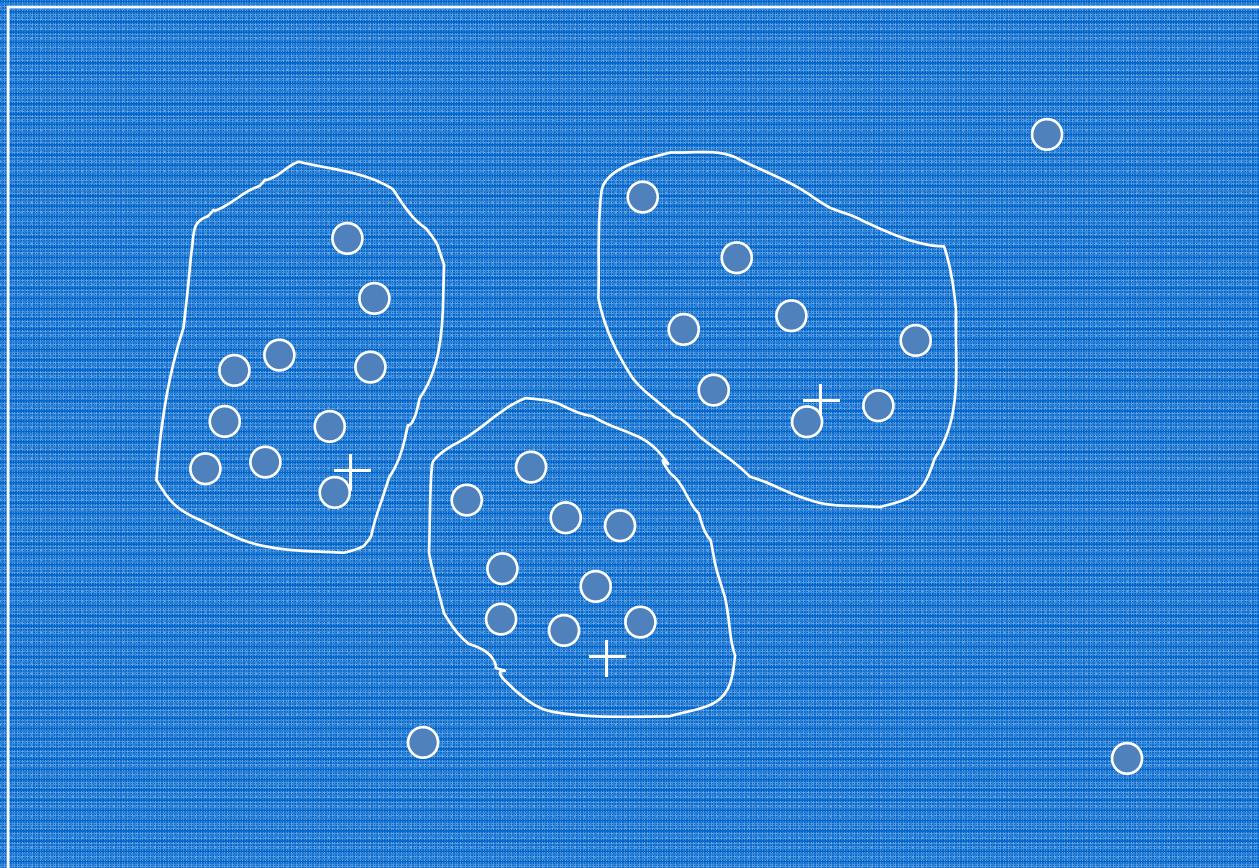
Binning Methods for Data Smoothing

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 21, 24, 25, 26, 28, 29, 34
- * Partition into equal-frequency (equi-depth) bins:
 - Bin 1: 4, 8, 9, 15
 - Bin 2: 21, 21, 24, 25
 - Bin 3: 26, 28, 29, 34
- * Smoothing by bin means:
 - Bin 1: 9, 9, 9, 9
 - Bin 2: 23, 23, 23, 23
 - Bin 3: 29, 29, 29, 29
- * Smoothing by bin boundaries:
 - Bin 1: 4, 4, 4, 15
 - Bin 2: 21, 21, 25, 25
 - Bin 3: 26, 26, 26, 34

Regression



Cluster Analysis



Data Cleaning as a Process

- Data discrepancy detection
 - Use metadata (e.g., domain, range, dependency, distribution)
 - Check field overloading
 - Check uniqueness rule, consecutive rule and null rule
 - Use commercial tools
 - Data scrubbing: use simple domain knowledge (e.g., postal code, spell-check) to detect errors and make corrections
 - Data auditing: by analyzing data to discover rules and relationship to detect violators (e.g., correlation and clustering to find outliers)
- Data migration and integration
 - Data migration tools: allow transformations to be specified
 - ETL (Extraction/Transformation>Loading) tools: allow users to specify transformations through a graphical user interface
- Integration of the two processes
 - Iterative and interactive (e.g., Potter's Wheels)

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Integration

- Data integration:
 - Combines data from multiple sources into a coherent store
- Schema integration: e.g., A.cust-id = B.cust-#
 - Integrate metadata from different sources
- Entity identification problem:
 - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
- Detecting and resolving data value conflicts
 - For the same real world entity, attribute values from different sources are different
 - Possible reasons: different representations, different scales, e.g., metric vs. British units

Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases
 - *Object identification:* The same attribute or object may have different names in different databases
 - *Derivable data:* One attribute may be a “derived” attribute in another table, e.g., annual revenue
- Redundant attributes may be able to be detected by *correlation analysis*
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Correlation Analysis (Numerical Data)

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

$$r_{A,B} = \frac{\sum (A - \bar{A})(B - \bar{B})}{(n - 1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\bar{A}\bar{B}}{(n - 1)\sigma_A \sigma_B}$$

where n is the number of tuples, \bar{A} and \bar{B} 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

Correlation Analysis (Categorical Data)

- χ^2 (chi-square) test

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

- The larger the χ^2 value, the more likely the variables are related
- The cells that contribute the most to the χ^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

Data Transformation

- Smoothing: remove noise from data
- Aggregation: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
 - min-max normalization
 - z-score normalization
 - normalization by decimal scaling
- Attribute/feature construction
 - New attributes constructed from the given ones

Data Transformation: Normalization

- Min-max normalization: to $[new_min_A, new_max_A]$

$$v' = \frac{v - min_A}{max_A - min_A} (new_max_A - new_min_A) + new_min_A$$

- Ex. Let income range \$12,000 to \$98,000 normalized to [0.0, 1.0]. Then \$73,000 is mapped to

$$\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0) + 0 = 0.716$$

- Z-score normalization (μ : mean, σ : standard deviation):

$$v' = \frac{v - \mu_A}{\sigma_A}$$

- Ex. Let $\mu = 54,000$, $\sigma = 16,000$. Then

- Normalization by decimal scaling

$$\frac{73,600 - 54,000}{16,000} = 1.225$$

$$v' = \frac{v}{10^j} \quad \text{Where } j \text{ is the smallest integer such that } \text{Max}(|v'|) < 1$$

Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Data Reduction Strategies

- 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
 - Data cube aggregation:
 - Dimensionality reduction — e.g., remove unimportant attributes
 - Data Compression
 - Numerosity reduction — e.g., fit data into models
 - Discretization and concept hierarchy generation

Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
 - The aggregated data for an individual entity of interest
 - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
 - Further reduce the size of data to deal with
- Reference appropriate levels
 - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

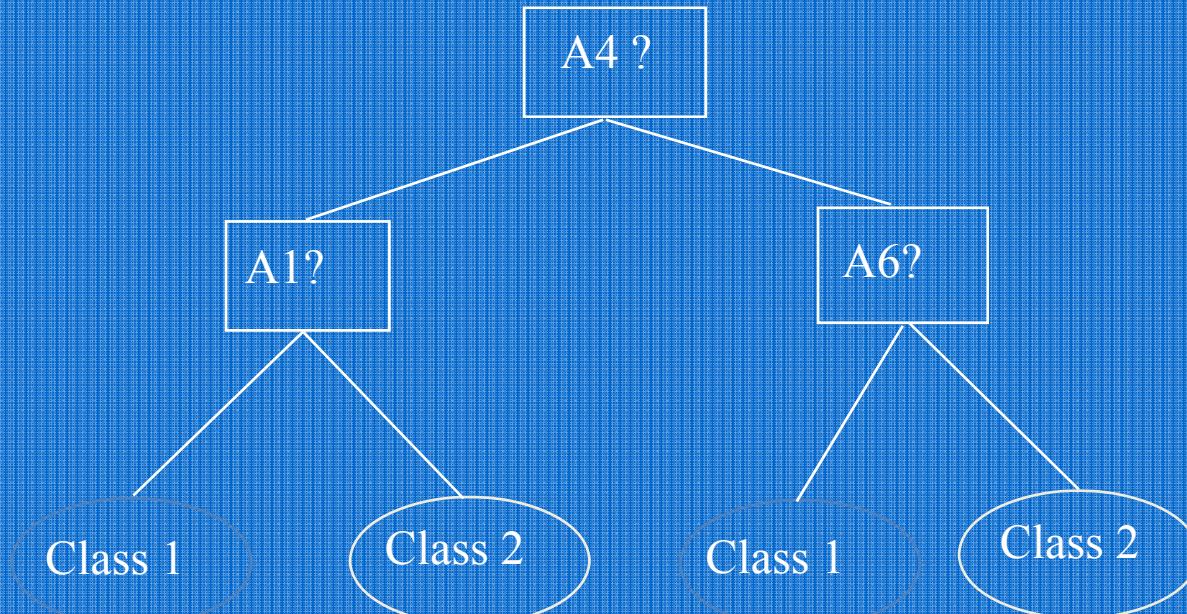
Attribute Subset Selection

- Feature selection (i.e., attribute subset selection):
 - Select a minimum set of features such that the probability distribution of different classes given the values for those features is as close as possible to the original distribution given the values of all features
 - reduce # of patterns in the patterns, easier to understand
- Heuristic methods (due to exponential # of choices):
 - Step-wise forward selection
 - Step-wise backward elimination
 - Combining forward selection and backward elimination
 - Decision-tree induction

Example of Decision Tree Induction

Initial attribute set:

{A1, A2, A3, A4, A5, A6}



-----> Reduced attribute set: {A1, A4, A6}

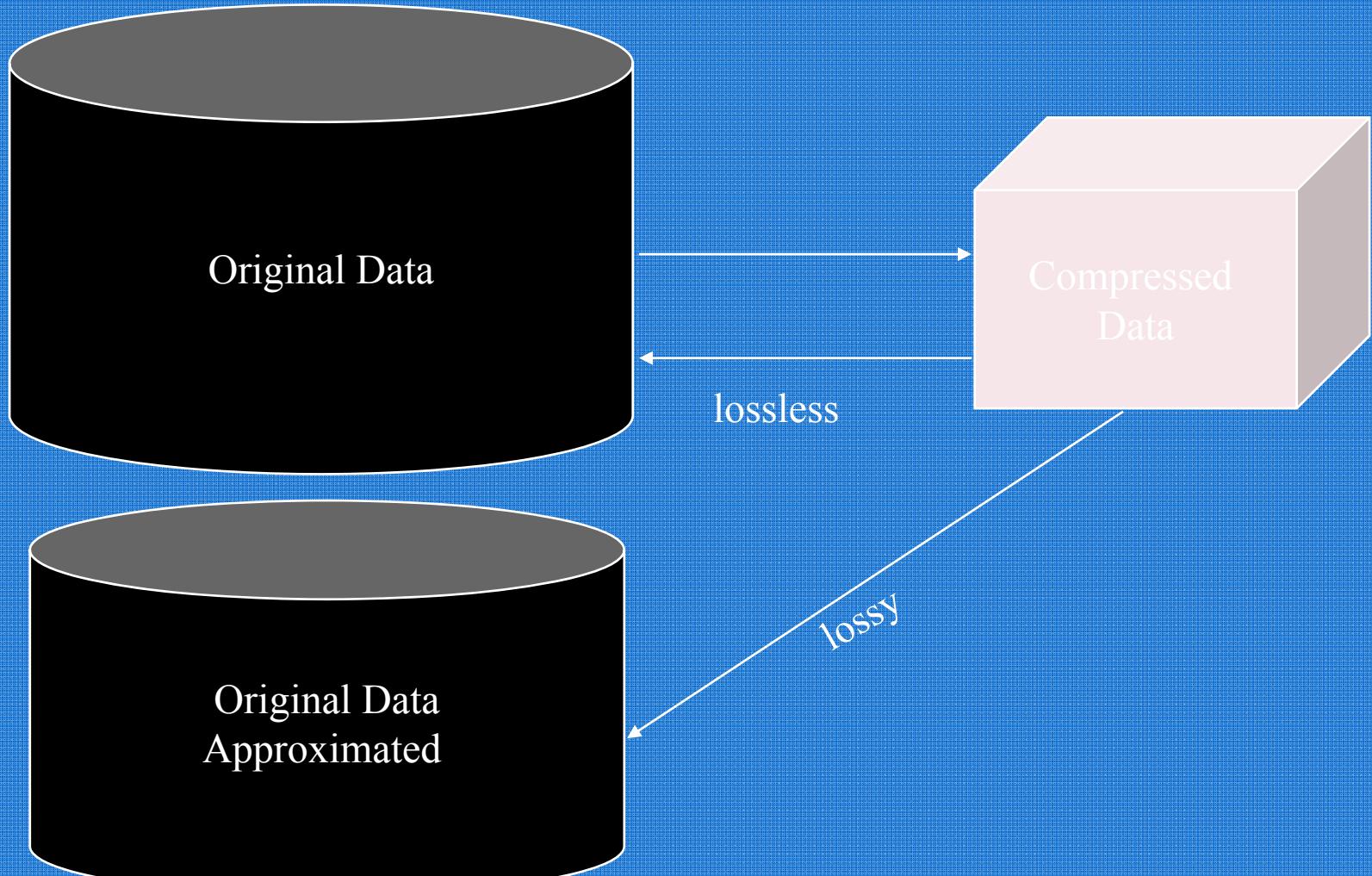
Heuristic Feature Selection Methods

- There are 2^d possible sub-features of d features
- Several heuristic feature selection methods:
 - Best single features under the feature independence assumption: choose by significance tests
 - Best step-wise feature selection:
 - The best single-feature is picked first
 - Then next best feature condition to the first, ...
 - Step-wise feature elimination:
 - Repeatedly eliminate the worst feature
 - Best combined feature selection and elimination
 - Optimal branch and bound:
 - Use feature elimination and backtracking

Data Compression

- String compression
 - There are extensive theories and well-tuned algorithms
 - Typically lossless
 - But only limited manipulation is possible without expansion
- Audio/video compression
 - Typically lossy compression, with progressive refinement
 - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence is not audio
 - Typically short and vary slowly with time

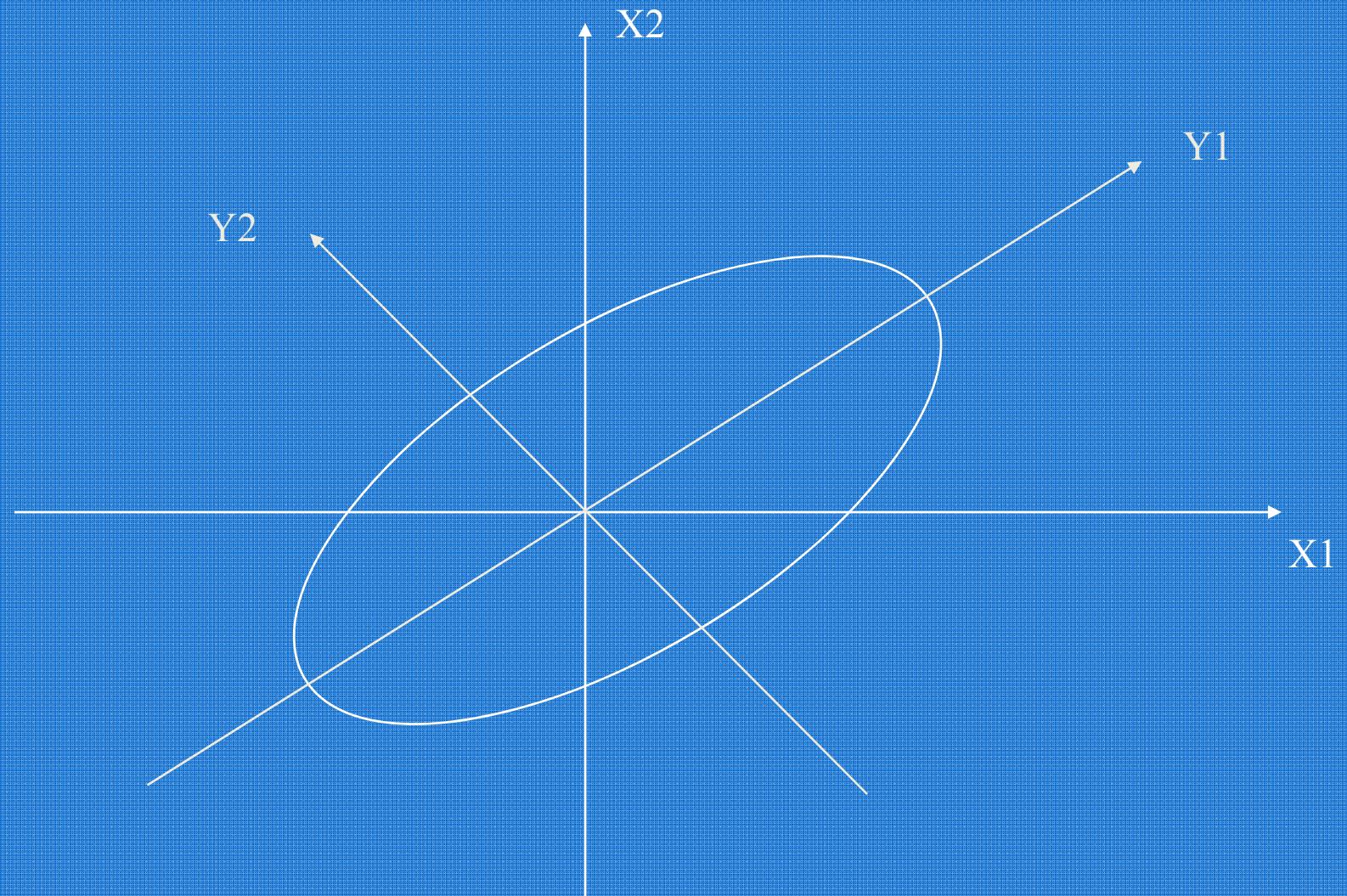
Data Compression



Dimensionality Reduction: Principal Component Analysis (PCA)

- Given N data vectors from n -dimensions, find $k \leq n$ orthogonal vectors (*principal components*) that can be best used to represent data
- Steps
 - Normalize input data: Each attribute falls within the same range
 - Compute k orthonormal (unit) vectors, i.e., *principal components*
 - Each input data (vector) is a linear combination of the k principal component vectors
 - The principal components are sorted in order of decreasing “significance” or strength
 - Since the components are sorted, the size of the data can be reduced by eliminating the weak components, i.e., those with low variance. (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data)
- Works for numeric data only
- Used when the number of dimensions is large

Principal Component Analysis



Data Reduction Method (1): Regression and Log-SRM Linear Models

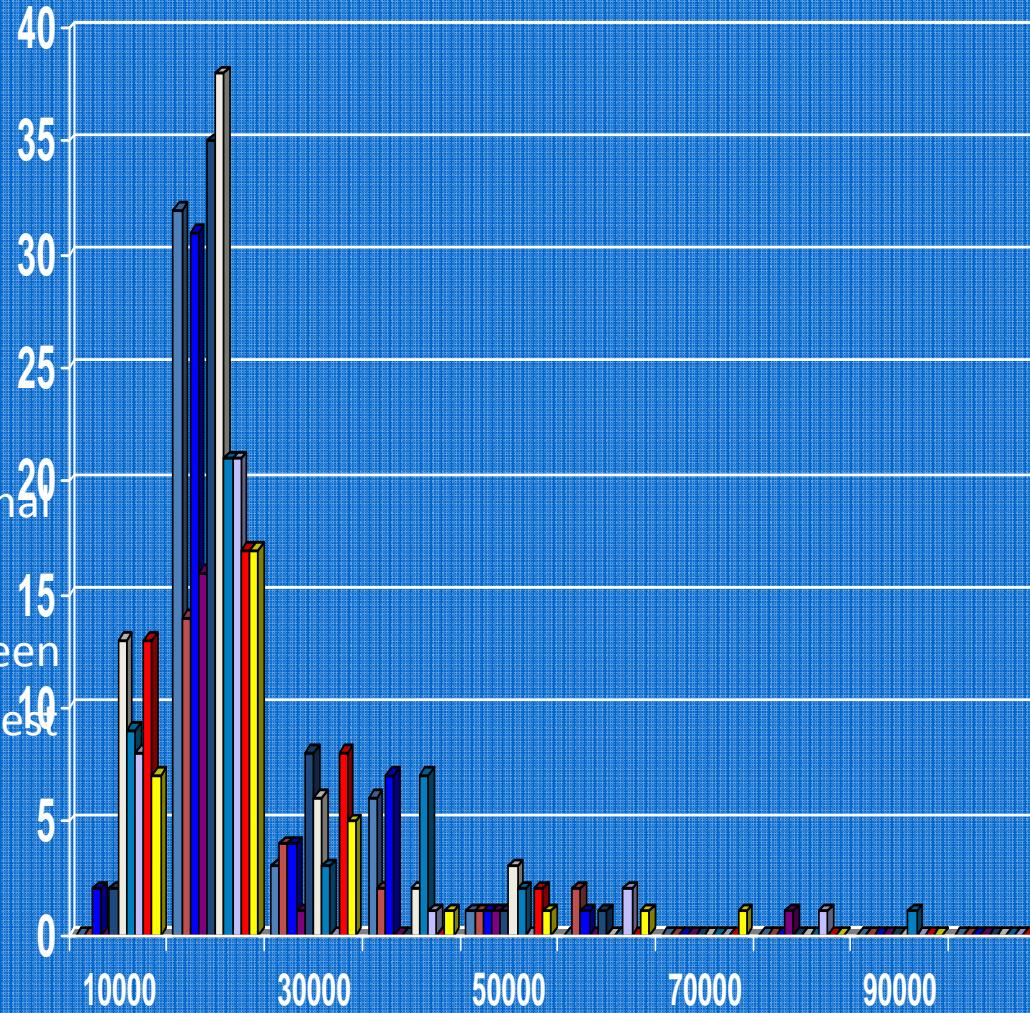
- Linear regression: Data are modeled to fit a straight line
 - Often uses the least-square method to fit the line
- Multiple regression: allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- Log-linear model: approximates discrete multidimensional probability distributions

- Linear regression: $Y = wX + b$
 - Two regression coefficients, w and b , specify the line and are to be estimated by using the data at hand
 - Using the least squares criterion to the known values of $Y_1, Y_2, \dots, X_1, X_2, \dots$
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2.$
 - Many nonlinear functions can be transformed into the above
- Log-linear models:
 - The multi-way table of joint probabilities is approximated by a product of lower-order tables
 - Probability: $p(a, b, c, d) = \alpha_{ab} \beta_{ac} \chi_{ad} \delta_{bcd}$

Data Reduction Method (2): Histograms



- Divide data into buckets and store average (sum) for each bucket
 - Partitioning rules:
 - Equal-width: equal bucket range
 - Equal-frequency (or equal-depth)
 - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)
 - MaxDiff: set bucket boundary between each pair for pairs have the $\beta - 1$ largest differences



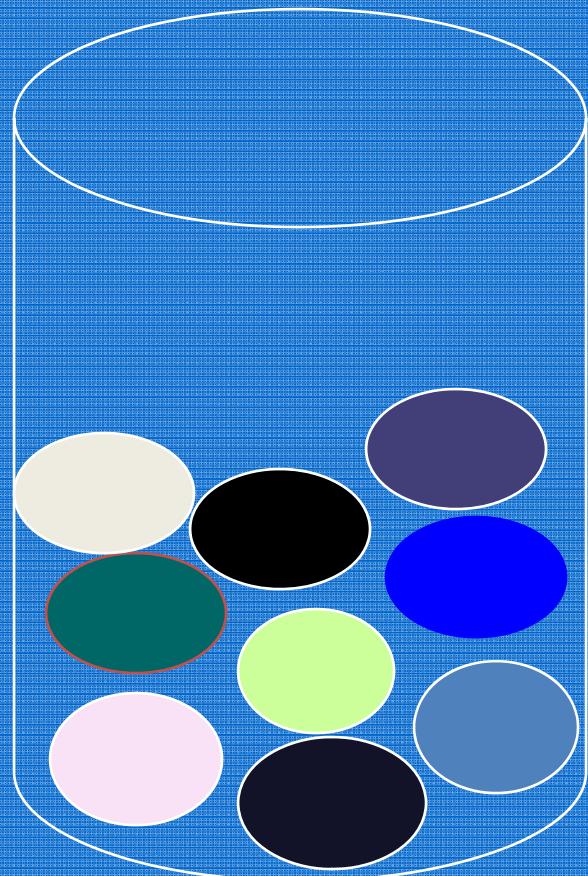
Data Reduction Method (3): Clustering

- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid and diameter) only
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms
- Cluster analysis will be studied in depth in Chapter 7

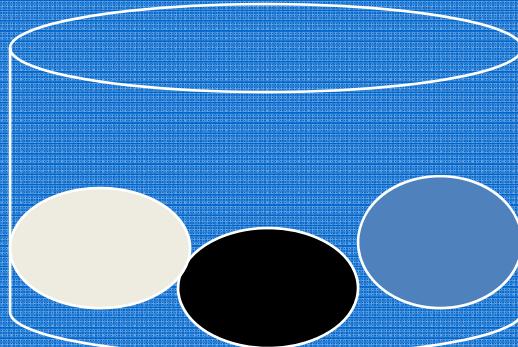
Data Reduction Method (4): Sampling

- Sampling: obtaining a small sample s to represent the whole data set N
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Choose a **representative** subset of the data
 - Simple random sampling may have very 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
 - Note: Sampling may not reduce database I/Os (page at a time)

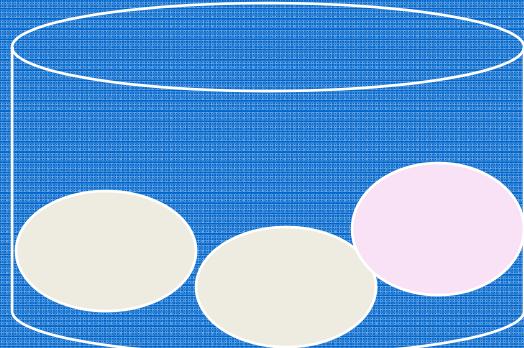
Sampling: with or without Replacement



SRSWOR
(simple random sample without replacement)



SRSWR



Data Preprocessing

- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Discretization

- Three types of attributes:
 - Nominal — values from an unordered set, e.g., color, profession
 - Ordinal — values from an ordered set, e.g., military or academic rank
 - Continuous — real numbers, e.g., integer or real numbers
- Discretization:
 - Divide the range of a continuous attribute into intervals
 - Some classification algorithms only accept categorical attributes.
 - Reduce data size by discretization
 - Prepare for further analysis

Discretization and Concept Hierarchy

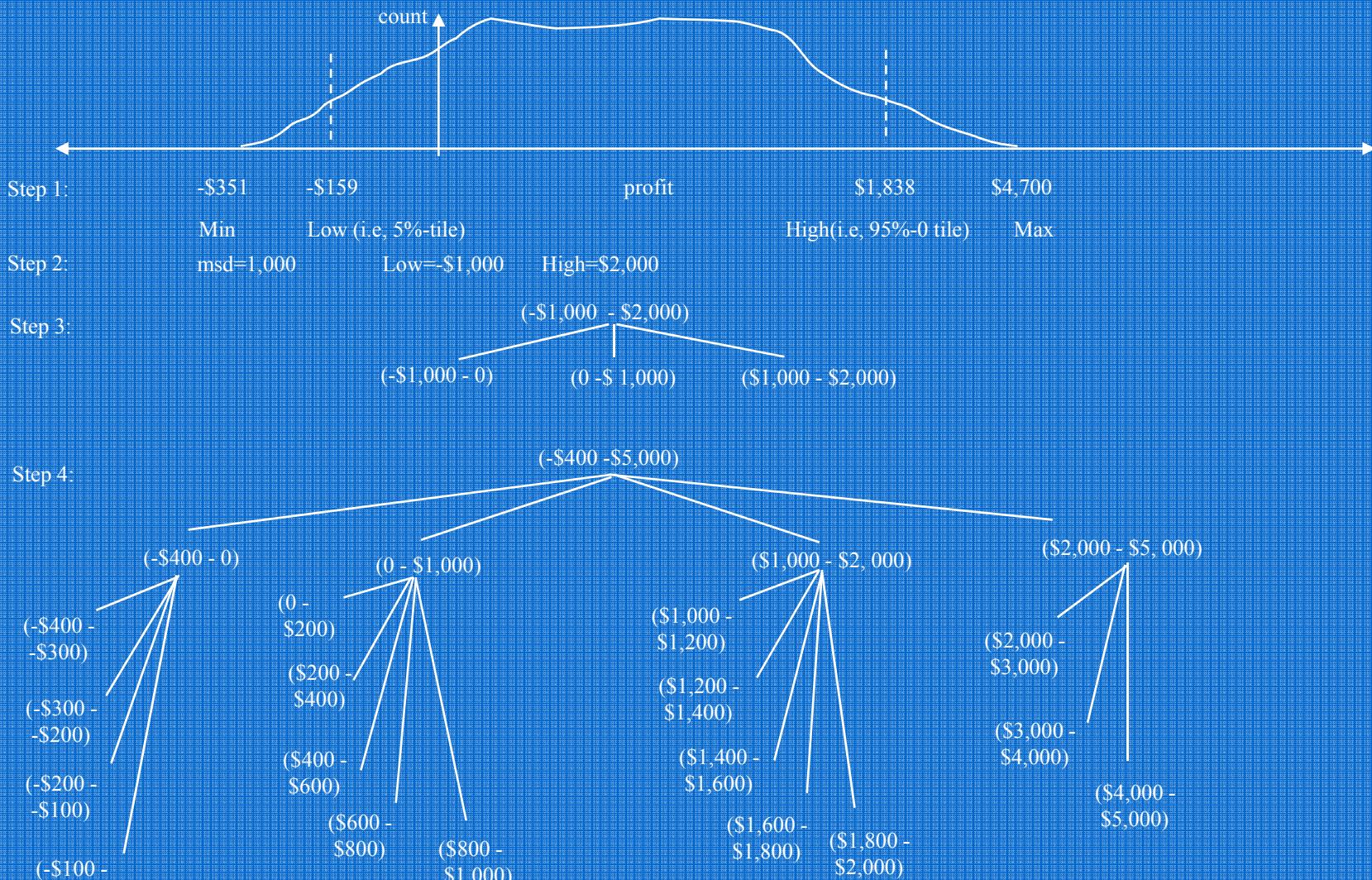
- Discretization
 - Reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals
 - Interval labels can then be used to replace actual data values
 - Supervised vs. unsupervised
 - Split (top-down) vs. merge (bottom-up)
 - Discretization can be performed recursively on an attribute
- Concept hierarchy formation
 - Recursively reduce the data by collecting and replacing low level concepts (such as numeric values for age) by higher level concepts (such as young, middle-aged, or senior)

Discretization and Concept Hierarchy Generation for Numeric Data



- Typical methods: All the methods can be applied recursively
 - Binning (covered above)
 - Top-down split, unsupervised,
 - Histogram analysis (covered above)
 - Top-down split, unsupervised
 - Clustering analysis (covered above)
 - Either top-down split or bottom-up merge, unsupervised
 - Entropy-based discretization: supervised, top-down split
 - Interval merging by χ^2 Analysis: unsupervised, bottom-up merge
 - Segmentation by natural partitioning: top-down split, unsupervised

Example of 3-4-5 Rule



Concept Hierarchy Generation for Categorical Data

- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
 - street < city < state < country
- Specification of a hierarchy for a set of values by explicit data grouping
 - {Urbana, Champaign, Chicago} < Illinois
- Specification of only a partial set of attributes
 - E.g., only street < city, not others
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
 - E.g., for a set of attributes: {street, city, state, country}

Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
 - The attribute with the most distinct values is placed at the lowest level of the hierarchy
 - Exceptions, e.g., weekday, month, quarter, year



- Why preprocess the data?
- Data cleaning
- Data integration and transformation
- Data reduction
- Discretization and concept hierarchy generation
- Summary

Summary

- Data preparation or preprocessing is a big issue for both data warehousing and data mining
- Descriptive data summarization is need for quality data preprocessing
- Data preparation includes
 - Data cleaning and data integration
 - Data reduction and feature selection
 - Discretization
- A lot a methods have been developed but data preprocessing still an active area of research

Review Questions

- How is data warehouse different from a database? How are they similar?
- List the five primitives for specifying a data mining task?
- State the data mining functionalities ?
- Enlist the classification of data mining systems
- Write a note on data mining query Language?
- Describe the steps involved in data mining when viewed as a process of knowledge discovery?
- State the various kinds of frequent pattern?
- Give an example for multidimensional association rule?
- State the need for outlier analysis?
- Are all of the pattern interesting?- Justify
- What are the possible integration schemes included in the integration of data mining system with a database or data ware house system ?

Bibliography

- Data mining concepts and Techniques by Jiawei Han and Micheline Kamber
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley & Sons, 2003

UNIT-II

Closed Patterns and Max-Patterns

- A long pattern contains a combinatorial number of sub-patterns, e.g., $\{a_1, \dots, a_{100}\}$ contains $\binom{100}{1} + \binom{100}{2} + \dots + \binom{100}{0} = 2^{100} - 1 = 1.27*10^{30}$ sub-patterns!
- Solution: *Mine closed patterns and max-patterns instead*
- An itemset X is **closed** if X is *frequent* and there exists *no super-pattern* Y ⊃ X, *with the same support* as X (proposed by Pasquier, et al. @ ICDT'99)
- An itemset X is a **max-pattern** if X is frequent and there exists no frequent super-pattern Y ⊃ X (proposed by Bayardo @ SIGMOD'98)
- Closed pattern is a lossless compression of freq. patterns
 - Reducing the # of patterns and rules

- Exercise. $DB = \{\langle a_1, \dots, a_{100} \rangle, \langle a_1, \dots, a_{50} \rangle\}$
 - $\text{Min_sup} = 1.$
- What is the set of closed itemset?
 - $\langle a_1, \dots, a_{100} \rangle: 1$
 - $\langle a_1, \dots, a_{50} \rangle: 2$
- What is the set of max-pattern?
 - $\langle a_1, \dots, a_{100} \rangle: 1$
- What is the set of all patterns?
 - !!

Chapter 5: Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary

Scalable Methods for Mining Frequent Patterns

- The downward closure property of frequent patterns
 - Any subset of a frequent itemset must be frequent
 - If $\{\text{beer, diaper, nuts}\}$ is frequent, so is $\{\text{beer, diaper}\}$
 - i.e., every transaction having $\{\text{beer, diaper, nuts}\}$ also contains $\{\text{beer, diaper}\}$
- Scalable mining methods: Three major approaches
 - Apriori (Agrawal & Srikant@VLDB'94)
 - Freq. pattern growth (FPgrowth—Han, Pei & Yin @SIGMOD'00)
 - Vertical data format approach (Charm—Zaki & Hsiao @SDM'02)

Apriori: A Candidate Generation-and-Test Approach

- Apriori pruning principle: If there is any itemset which is infrequent, its superset should not be generated/tested!
(Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)
- Method:
 - Initially, scan DB once to get frequent 1-itemset
 - Generate length $(k+1)$ candidate itemsets from length k frequent itemsets
 - Test the candidates against DB
 - Terminate when no frequent or candidate set can be generated

The Apriori Algorithm—An Example

Database TDB

Tid	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

$$\text{Sup}_{\min} = 2$$

C_1

1st scan

Itemset	sup
{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L_1

Itemset	sup
{A}	2
{B}	3
{C}	3
{E}	3

L_2

Itemset	sup
{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

C_2

2nd scan

Itemset	sup
{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

Itemset
{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

Itemset
{B, C, E}

3rd scan

Itemset	sup
{B, C, E}	2

The Apriori Algorithm

- Pseudo-code:

C_k : Candidate itemset of size k

L_k : frequent itemset of size k

$L_1 = \{\text{frequent items}\};$

for ($k = 1; L_k \neq \emptyset; k++$) **do begin**

C_{k+1} = candidates generated from L_k ;

for each transaction t in database do

increment the count of all candidates in C_{k+1}

that are contained in t

L_{k+1} = candidates in C_{k+1} with min_support

end

return $\cup_k L_k$;

Important Details of Apriori

- How to generate candidates?
 - Step 1: self-joining L_k
 - Step 2: pruning
- How to count supports of candidates?
- Example of Candidate-generation
 - $L_3 = \{abc, abd, acd, ace, bcd\}$
 - Self-joining: $L_3 * L_3$
 - $abcd$ from abc and abd
 - $acde$ from acd and ace
 - Pruning:
 - $acde$ is removed because ade is not in L_3
 - $C_4 = \{abcd\}$

How to Generate Candidates?

- Suppose the items in L_{k-1} are listed in an order
- Step 1: self-joining L_{k-1}

insert into C_k

select $p.item_1, p.item_2, \dots, p.item_{k-1}, q.item_{k-1}$

from $L_{k-1} p, L_{k-1} q$

where $p.item_1=q.item_1, \dots, p.item_{k-2}=q.item_{k-2}, p.item_{k-1} < q.item_{k-1}$

- Step 2: pruning

forall *itemsets c in C_k* do

forall *(k-1)-subsets s of c* do

if (s is not in L_{k-1}) then delete c from C_k

How to Count Supports of Candidates?

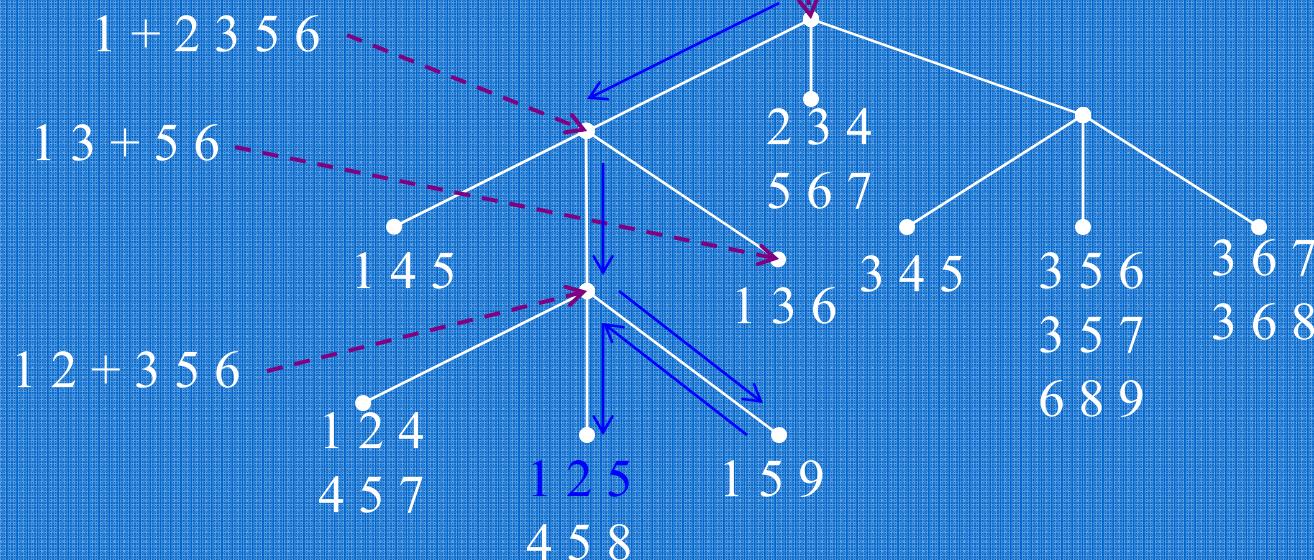
- Why counting supports of candidates a problem?
 - The total number of candidates can be very huge
 - One transaction may contain many candidates
- Method:
 - Candidate itemsets are stored in a *hash-tree*
 - *Leaf node* of hash-tree contains a list of itemsets and counts
 - *Interior node* contains a hash table
 - *Subset function*: finds all the candidates contained in a transaction

Example: Counting Supports of Candidates

Subset function



Transaction: 1 2 3 5 6



Efficient Implementation of Apriori in SQL

- Hard to get good performance out of pure SQL (SQL-92) based approaches alone
- Make use of object-relational extensions like UDFs, BLOBs, Table functions etc.
 - Get orders of magnitude improvement
- S. Sarawagi, S. Thomas, and R. Agrawal. Integrating association rule mining with relational database systems: Alternatives and implications. In SIGMOD'98

Challenges of Frequent Pattern Mining

- Challenges
 - Multiple scans of transaction database
 - Huge number of candidates
 - Tedium workload of support counting for candidates
- Improving Apriori: general ideas
 - Reduce passes of transaction database scans
 - Shrink number of candidates
 - Facilitate support counting of candidates

Partition: Scan Database Only Twice

- Any itemset that is potentially frequent in DB must be frequent in at least one of the partitions of DB
 - Scan 1: partition database and find local frequent patterns
 - Scan 2: consolidate global frequent patterns
- A. Savasere, E. Omiecinski, and S. Navathe. An efficient algorithm for mining association in large databases. In *VLDB'95*

Sampling for Frequent Patterns

- Select a sample of original database, mine frequent patterns within sample using Apriori
- Scan database once to verify frequent itemsets found in sample, only *borders* of closure of frequent patterns are checked
 - Example: check *abcd* instead of *ab*, *ac*, ..., *etc.*
- Scan database again to find missed frequent patterns
- H. Toivonen. Sampling large databases for association rules. In *VLDB'96*

Bottleneck of Frequent-pattern Mining

- Multiple database scans are **costly**
- Mining long patterns needs many passes of scanning and generates lots of candidates
 - To find frequent itemset $i_1 i_2 \dots i_{100}$
 - # of scans: 100
 - # of Candidates: $\binom{100}{1} + \binom{100}{2} + \dots + \binom{100}{0} = 2^{100}-1 = 1.27*10^{30}$!
- Bottleneck: candidate-generation-and-test
- Can we avoid candidate generation?

Mining Frequent Patterns Without Candidate Generation

- Grow long patterns from short ones using local frequent items
 - “abc” is a frequent pattern
 - Get all transactions having “abc”: DB | abc
 - “d” is a local frequent item in DB | abc → abcd is a frequent pattern

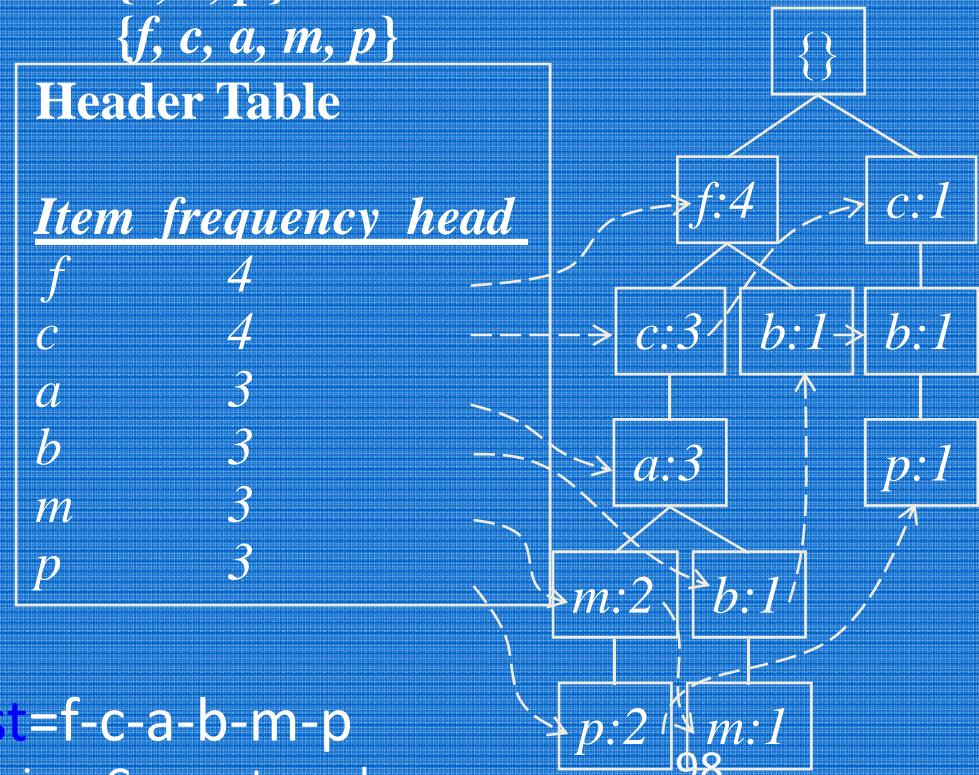
Construct FP-tree from a Transaction Database

<i>TID</i>	<i>Items bought</i>	<i>(ordered) frequent items</i>	<i>min_support = 3</i>
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}	
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}	
300	{b, f, h, j, o, w}	{f, b}	
400	{b, c, k, s, p}	{c, b, p}	
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}	

1. Scan DB once, find frequent 1-itemset (single item pattern)
2. Sort frequent items in frequency descending order, f-list
3. Scan DB again, construct FP-tree

Header Table		
<i>Item</i>	<i>frequency</i>	<i>head</i>
f	4	
c	4	
a	3	
b	3	
m	3	
p	3	

F-list=f-c-a-b-m-p
Data Mining: Concepts and



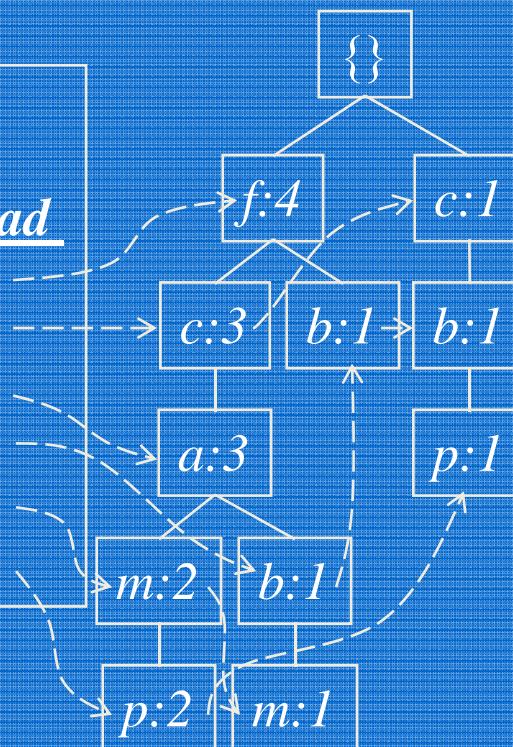
Benefits of the FP-tree Structure

- Completeness
 - Preserve complete information for frequent pattern mining
 - Never break a long pattern of any transaction
- Compactness
 - Reduce irrelevant info—infrequent items are gone
 - Items in frequency descending order: the more frequently occurring, the more likely to be shared
 - Never be larger than the original database (not count node-links and the *count* field)
 - For Connect-4 DB, compression ratio could be over 100

Find Patterns Having P From P-conditional Database

- Starting at the frequent item header table in the FP-tree
- Traverse the FP-tree by following the link of each frequent item p
- Accumulate all of *transformed prefix paths* of item p to form p 's conditional pattern base

Header Table		
	<u>Item frequency</u>	<u>head</u>
f	4	---
c	4	---
a	3	---
b	3	---
m	3	---
p	3	---



Conditional pattern bases

<u>item</u>	<u>cond. pattern base</u>
c	$f:3$
a	$fc:3$
b	$fca:1, f:1, c:1$
m	$fca:2, fcab:1$
p	$fcam:2, cb:1$

Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary

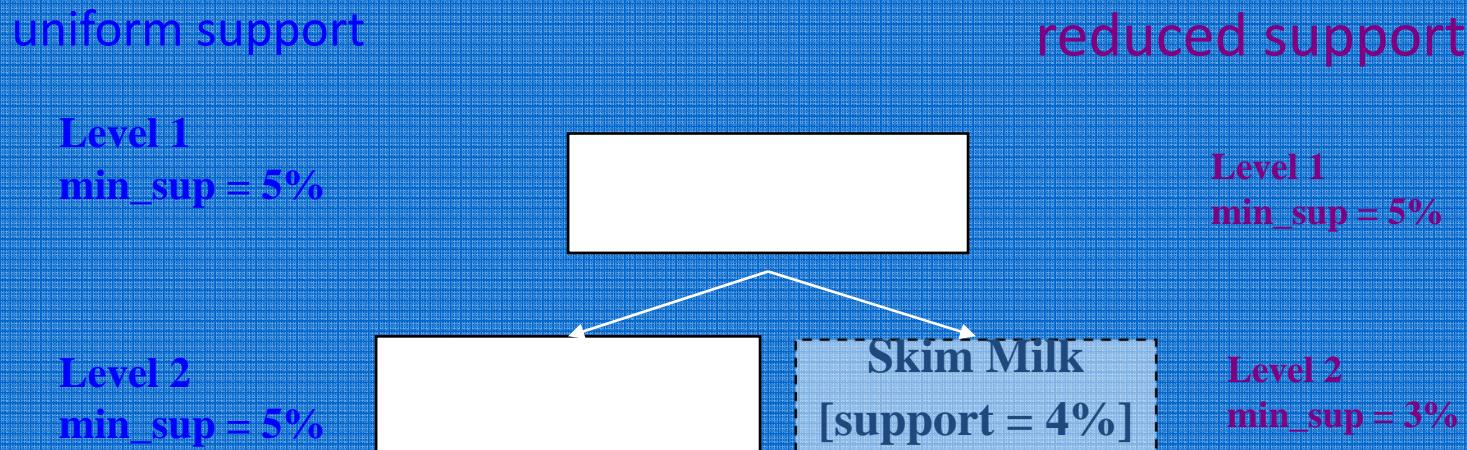


Mining Various Kinds of Association Rules

- Mining multilevel association
- Mining multidimensional association
- Mining quantitative association
- Mining interesting correlation patterns

Mining Multiple-Level Association Rules

- Items often form hierarchies
- Flexible support settings
 - Items at the lower level are expected to have lower support
- Exploration of *shared* multi-level mining (Agrawal & Srikant@VLB'95, Han & Fu@VLDB'95)



Multi-level Association: Redundancy Filtering

- Some rules may be redundant due to “ancestor” relationships between items.
- Example
 - $\text{milk} \Rightarrow \text{wheat bread}$ [support = 8%, confidence = 70%]
 - $2\% \text{ milk} \Rightarrow \text{wheat bread}$ [support = 2%, confidence = 72%]
- We say the first rule is an ancestor of the second rule.
- A rule is redundant if its support is close to the “expected” value, based on the rule’s ancestor.

Mining Multi-Dimensional Association

- Single-dimensional rules:
 $\text{buys}(X, \text{"milk"}) \Rightarrow \text{buys}(X, \text{"bread"})$
- Multi-dimensional rules: ≥ 2 dimensions or predicates
 - Inter-dimension assoc. rules (*no repeated predicates*)
 $\text{age}(X, \text{"19-25"}) \wedge \text{occupation}(X, \text{"student"}) \Rightarrow \text{buys}(X, \text{"coke"})$
 - hybrid-dimension assoc. rules (*repeated predicates*)
 $\text{age}(X, \text{"19-25"}) \wedge \text{buys}(X, \text{"popcorn"}) \Rightarrow \text{buys}(X, \text{"coke"})$
- 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

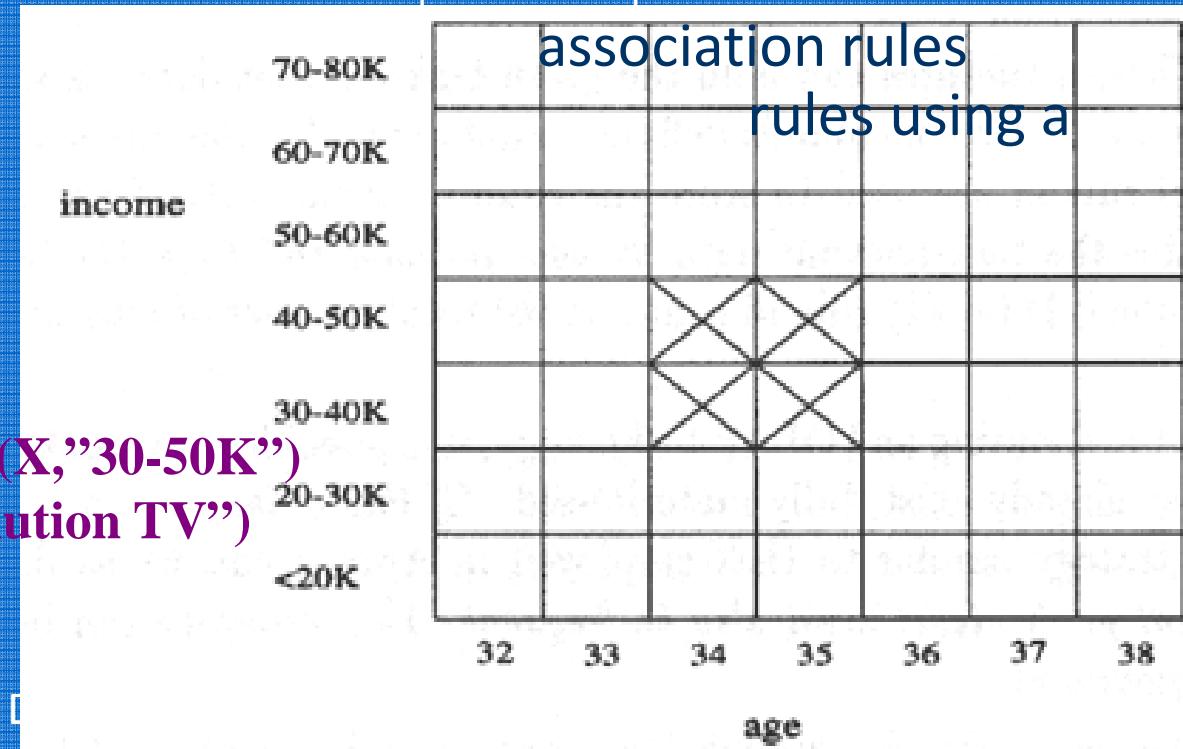
Mining Quantitative Associations

- 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 (quantitative rules, e.g., Agrawal & Srikant@SIGMOD96)
 3. Clustering: Distance-based association (e.g., Yang & Miller@SIGMOD97)
 - one dimensional clustering then association
 4. Deviation: (such as Aumann and Lindell@KDD99)
Sex = female => Wage: mean=\$7/hr (overall mean = \$9)

Quantitative Association Rules

- Proposed by Lent, Swami and Widom ICDE'97
- Numeric attributes are *dynamically* discretized
 - Such that the confidence or compactness of the rules mined is maximized
- 2-D quantitative association rules: $A_{\text{quan1}} \wedge A_{\text{quan2}} \Rightarrow A_{\text{cat}}$
- Cluster *adjacent* to form general 2-D grid
- Example

$\text{age}(X, "34-35") \wedge \text{income}(X, "30-50K")$
 $\Rightarrow \text{buys}(X, \text{"high resolution TV"})$



Mining Other Interesting Patterns

- Flexible support constraints (Wang et al. @ VLDB'02)
 - Some items (e.g., diamond) may occur rarely but are valuable
 - Customized sup_{\min} specification and application
- Top-K closed frequent patterns (Han, et al. @ ICDM'02)
 - Hard to specify sup_{\min} , but top-k with length_{\min} is more desirable
 - Dynamically raise sup_{\min} in FP-tree construction and mining, and select most promising path to mine

Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary



Interestingness Measure: Correlations (Lift)

- $play\ basketball \Rightarrow eat\ cereal$ [40%, 66.7%] is misleading
 - The overall % of students eating cereal is 75% > 66.7%.
- $play\ basketball \Rightarrow not\ eat\ cereal$ [20%, 33.3%] is more accurate, although with lower support and confidence
- Measure of dependent/correlated events: lift

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

	Basketball	Not basketball	Sum (row)
Cereal	2000	1750	3750
Not cereal	1000	250	1250
Sum(col.)	3000	2000	5000

$$lift(B, C) = \frac{2000/5000}{3000/5000 * 3750/5000} = 0.89$$

$$lift(B, \neg C) = \frac{1000/5000}{3000/5000 * 1250/5000} = 1.33$$

Chapter 5: Mining Frequent Patterns, Association and Correlations

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining 
- Summary

Constraint-based (Query-Directed) Mining

- Finding all the patterns in a database autonomously? — unrealistic!
 - The patterns could be too many but not focused!
- Data mining should be an interactive process
 - User directs what to be mined using a data mining query language (or a graphical user interface)
- Constraint-based mining
 - User flexibility: provides constraints on what to be mined
 - System optimization: explores such constraints for efficient mining—constraint-based mining

Constraints in Data Mining

- Knowledge type constraint:
 - classification, association, etc.
- Data constraint — using SQL-like queries
 - find product pairs sold together in stores in **Chicago** in Dec.'02
- Dimension/level constraint
 - in relevance to region, price, brand, customer category
- Rule (or pattern) constraint
 - small sales (price $< \$10$) triggers big sales ($\text{sum} > \200)
- Interestingness constraint
 - strong rules: $\text{min_support} \geq 3\%$, $\text{min_confidence} \geq 60\%$

Constrained Mining vs. Constraint-Based Search

- Constrained mining vs. constraint-based search/reasoning
 - Both are aimed at reducing search space
 - Finding all patterns satisfying constraints vs. finding some (or one) answer in constraint-based search in AI
 - Constraint-pushing vs. heuristic search
 - It is an interesting research problem on how to integrate them
- Constrained mining vs. query processing in DBMS
 - Database query processing requires to find all
 - Constrained pattern mining shares a similar philosophy as pushing selections deeply in query processing

The Apriori Algorithm — Example

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

C_1

Scan D

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

C_2

Scan D

itemset	sup.
{1 2}	1
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_2

L_2

itemset	sup.
{1 3}	2
{2 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

C_3

itemset
{2 3 5}

Scan D

L_3

itemset	sup.
{2 3 5}	2

Naïve Algorithm: Apriori + Constraint

Database D

TID	Items
100	1 3 4
200	2 3 5
300	1 2 3 5
400	2 5

C_1

Scan D

itemset	sup.
{1}	2
{2}	3
{3}	3
{4}	1
{5}	3

L_1

itemset	sup.
{1}	2
{2}	3
{3}	3
{5}	3

L_2

itemset	sup
{1 3}	2
{2 3}	2
{2 5}	3
{3 5}	2

C_2

Scan D

itemset	sup
{1 2}	1
{1 3}	2
{1 5}	1
{2 3}	2
{2 5}	3
{3 5}	2

C_2

itemset
{1 2}
{1 3}
{1 5}
{2 3}
{2 5}
{3 5}

December 26, 2012
{2 3 5}

Scan D

L_3

Data Mining: Concepts and

itemset	sup
{2 3 5}	2

Constraint:

$\text{Sum}\{\text{S.price}\} < 5$

- Basic concepts and a road map
- Efficient and scalable frequent itemset mining methods
- Mining various kinds of association rules
- From association mining to correlation analysis
- Constraint-based association mining
- Summary



Frequent-Pattern Mining: Summary

- Frequent pattern mining—an important task in data mining
- Scalable frequent pattern mining methods
 - Apriori (Candidate generation & test)
 - Projection-based (FPgrowth, CLOSET+, ...)
 - Vertical format approach (CHARM, ...)
- Mining a variety of rules and interesting patterns
- Constraint-based mining
- Mining sequential and structured patterns
- Extensions and applications

Cluster Analysis

1. What is Cluster Analysis? 
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary

What is Cluster Analysis?

- Cluster: a collection of data objects
 - Similar to one another within the same cluster
 - Dissimilar to the objects in other clusters
- Cluster analysis
 - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes
- Typical applications
 - As a stand-alone tool to get insight into data distribution
 - As a preprocessing step for other algorithms

Clustering: Rich Applications and Multidisciplinary Efforts



- Pattern Recognition
- Spatial Data Analysis
 - Create thematic maps in GIS by clustering feature spaces
 - Detect spatial clusters or for other spatial mining tasks
- Image Processing
- Economic Science (especially market research)
- WWW
 - Document classification
 - Cluster Weblog data to discover groups of similar access patterns

Examples of Clustering Applications

- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Land use: Identification of areas of similar land use in an earth observation database
- Insurance: Identifying groups of motor insurance policy holders with a high average claim cost
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along continent faults

Quality: What Is Good Clustering?

- A good clustering method will produce high quality clusters with
 - high intra-class similarity
 - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns

Measure the Quality of Clustering

- Dissimilarity/Similarity metric: Similarity is expressed in terms of a distance function, typically metric: $d(i, j)$
- There is a separate “quality” function that measures the “goodness” of a cluster.
- The definitions of distance functions are usually very different for interval-scaled, boolean, categorical, ordinal ratio, and vector variables.
- Weights should be associated with different variables based on applications and data semantics.
- It is hard to define “similar enough” or “good enough”
 - the answer is typically highly subjective.

Requirements of Clustering in Data Mining

- Scalability
- Ability to deal with different types of attributes
- Ability to handle dynamic data
- Discovery of clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to order of input records
- High dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Data Structures

- Data matrix
 - (two modes)

$$\begin{bmatrix} x_{11} & \dots & x_{1f} & \dots & x_{1p} \\ \dots & \dots & \dots & \dots & \dots \\ x_{i1} & \dots & x_{if} & \dots & x_{ip} \\ \dots & \dots & \dots & \dots & \dots \\ x_{n1} & \dots & x_{nf} & \dots & x_{np} \end{bmatrix}$$

- Dissimilarity matrix
 - (one mode)

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \ddots & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

Type of data in clustering analysis

- Interval-scaled variables
- Binary variables
- Nominal, ordinal, and ratio variables
- Variables of mixed types

Interval-valued variables

- Standardize data
 - Calculate the mean absolute deviation:

$$s_f = \frac{1}{n}(|x_{1f} - m_f| + |x_{2f} - m_f| + \dots + |x_{nf} - m_f|)$$

where $m_f = \frac{1}{n}(x_{1f} + x_{2f} + \dots + x_{nf})$.

- Calculate the standardized measurement (z-score)

$$z_{if} = \frac{x_{if} - m_f}{s_f}$$

- Using mean absolute deviation is more robust than using standard deviation

Similarity and Dissimilarity Between Objects

- Distances are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include: *Minkowski distance*:

$$d(i, j) = \sqrt[q]{(|x_{i1} - x_{j1}|^q + |x_{i2} - x_{j2}|^q + \dots + |x_{ip} - x_{jp}|^q)}$$

where $i = (x_{i1}, x_{i2}, \dots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \dots, x_{jp})$ are two p -dimensional data objects, and q is a positive integer

- If $q = 1$, d is Manhattan distance

$$d(i, j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \dots + |x_{ip} - x_{jp}|$$

Similarity and Dissimilarity Between Objects (Cont.)

- If $q = 2$, d is Euclidean distance:

$$d(i, j) = \sqrt{(|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2)}$$

- Properties

- $d(i, j) \geq 0$
- $d(i, i) = 0$
- $d(i, j) = d(j, i)$
- $d(i, j) \leq d(i, k) + d(k, j)$

- Also, one can use weighted distance, parametric Pearson product moment correlation, or other dissimilarity measures

Binary Variables

- A contingency table for binary data

		Object <i>j</i>		
		1	0	<i>sum</i>
Object <i>i</i>	1	<i>a</i>	<i>b</i>	<i>a+b</i>
	0	<i>c</i>	<i>d</i>	<i>c+d</i>
<i>sum</i>	<i>a+c</i>	<i>b+d</i>	<i>p</i>	

- Distance measure for symmetric binary variables:

$$d(i, j) = \frac{b + c}{a + b + c + d}$$

- Distance measure for asymmetric binary variables:

$$d(i, j) = \frac{b + c}{a + b + c}$$

- Jaccard coefficient (*similarity* measure for *asymmetric* binary variables):

$$\text{sim}_{\text{Jaccard}}(i, j) = \frac{a}{a + b + c}$$

Dissimilarity between Binary Variables

- Example

Name	Gender	Fever	Cough	Test-1	Test-2	Test-3	Test-4
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

- gender is a symmetric attribute
- the remaining attributes are asymmetric binary
- let the values Y and P be set to 1, and the value N be set to 0

$$d(\text{jack}, \text{mary}) = \frac{0 + 1}{2 + 0 + 1} = 0.33$$

$$d(\text{jack}, \text{Jim}) = \frac{1 + 1}{1 + 1 + 1} = 0.67$$

$$d(\text{Jim}, \text{mary}) = \frac{1 + 2}{1 + 1 + 2} = 0.75$$

Nominal Variables

- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
 - m : # of matches, p : total # of variables

$$d(i, j) = \frac{p - m}{p}$$

- Method 2: use a large number of binary variables
 - creating a new binary variable for each of the M nominal states

Ordinal Variables

- An ordinal variable can be discrete or continuous
- Order is important, e.g., rank
- Can be treated like interval-scaled

— replace x_{if} by their rank

$$r_{if} \in \{1, \dots, M_f\}$$

— map the range of each variable onto $[0, 1]$ by replacing i -th object in the f -th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

— compute the dissimilarity using methods for interval-scaled variables

Ratio-Scaled Variables

- Ratio-scaled variable: a positive measurement on a nonlinear scale, approximately at exponential scale, such as Ae^{Bt} or Ae^{-Bt}
- Methods:
 - treat them like interval-scaled variables—*not a good choice!* (why?—the scale can be distorted)
 - apply logarithmic transformation
$$y_{if} = \log(x_{if})$$
 - treat them as continuous ordinal data treat their rank as interval-scaled

Variables of Mixed Types

- A database may contain all the six types of variables
 - symmetric binary, asymmetric binary, nominal, ordinal, interval and ratio
- One may use a weighted formula to combine their effects

$$d_{ij} = \frac{\sum_{f=1}^p \delta_{ij}^{(f)} d_{ij}^{(f)}}{\sum_{f=1}^p \delta_{ij}^{(f)}}$$

– f is binary or nominal:

$d_{ij}^{(f)} = 0$ if $x_{if} = x_{jf}$, or $d_{ij}^{(f)} = 1$ otherwise

- f is interval-based: use the normalized distance
- f is ordinal or ratio-scaled
 - compute ranks r_{if} and
 - and treat z_{if} as interval-scaled

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

Vector Objects

- Vector objects: keywords in documents, gene features in micro-arrays, etc.
- Broad applications: information retrieval, biologic taxonomy
- Cosine measure

$$s(\vec{X}, \vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{|\vec{X}| |\vec{Y}|},$$

\vec{X}^t is a transposition of vector \vec{X} , $|\vec{X}|$ is the Euclidean normal of vector \vec{X} ,

- A variant: Tanimoto

$$s(\vec{X}, \vec{Y}) = \frac{\vec{X}^t \cdot \vec{Y}}{\vec{X}^t \cdot \vec{X} + \vec{Y}^t \cdot \vec{Y} - \vec{X}^t \cdot \vec{Y}},$$

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Major Clustering Approaches (I)

- Partitioning approach:
 - Construct various partitions and then evaluate them by some criterion, e.g., minimizing the sum of square errors
 - Typical methods: k-means, k-medoids, CLARANS
- Hierarchical approach:
 - Create a hierarchical decomposition of the set of data (or objects) using some criterion
 - Typical methods: Diana, Agnes, BIRCH, ROCK, CAMELEON
- Density-based approach:
 - Based on connectivity and density functions
 - Typical methods: DBSCAN, OPTICS, DenClue

Major Clustering Approaches (II)

- Grid-based approach:
 - based on a multiple-level granularity structure
 - Typical methods: STING, WaveCluster, CLIQUE
- Model-based:
 - A model is hypothesized for each of the clusters and tries to find the best fit of that model to each other
 - Typical methods: EM, SOM, COBWEB
- Frequent pattern-based:
 - Based on the analysis of frequent patterns
 - Typical methods: pCluster
- User-guided or constraint-based:
 - Clustering by considering user-specified or application-specific constraints
 - Typical methods: COD (obstacles), constrained clustering

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Partitioning Algorithms: Basic Concept

- Partitioning method: Construct a partition of a database D of n objects into a set of k clusters, s.t., min sum of squared distance

$$\sum_{m=1}^k \sum_{t_{mi} \in K_m} (C_m - t_{mi})^2$$

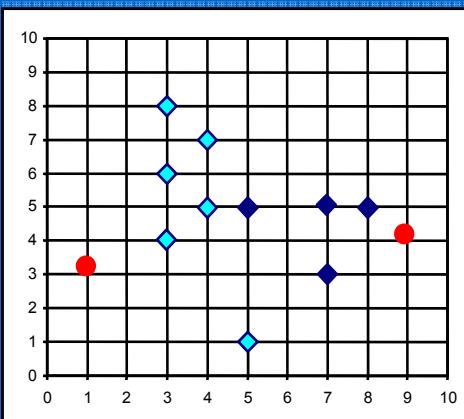
- Given a k , find a partition of k *clusters* that optimizes the chosen partitioning criterion
 - Global optimal: exhaustively enumerate all partitions
 - Heuristic methods: *k-means* and *k-medoids* algorithms
 - *k-means* (MacQueen'67): Each cluster is represented by the center of the cluster
 - *k-medoids* or PAM (Partition around medoids) (Kaufman & Rousseeuw'87): Each cluster is represented by one of the objects in the cluster

The *K-Means* Clustering Method

- Given k , the *k-means* algorithm is implemented in four steps:
 - Partition objects into k nonempty subsets
 - Compute seed points as the centroids of the clusters of the current partition (the centroid is the center, i.e., *mean point*, of the cluster)
 - Assign each object to the cluster with the nearest seed point
 - Go back to Step 2, stop when no more new assignment

The *K*-Means Clustering Method

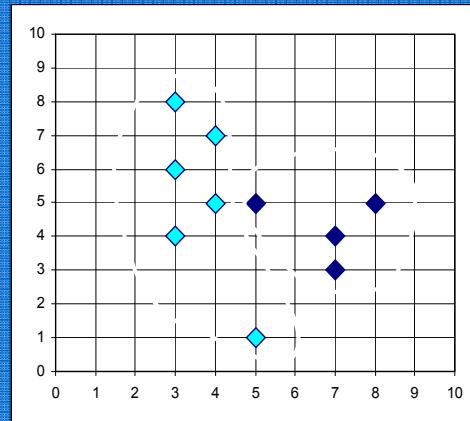
- Example



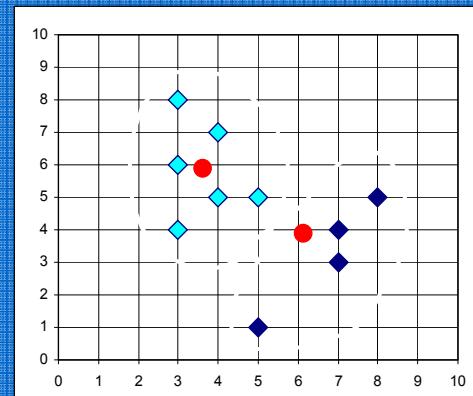
K=2

Arbitrarily choose K object as initial cluster center

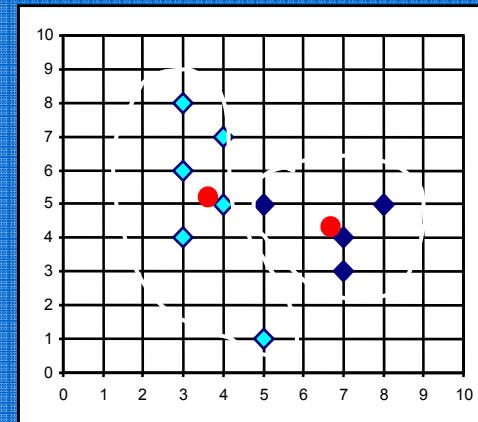
Assign each objects to most similar center



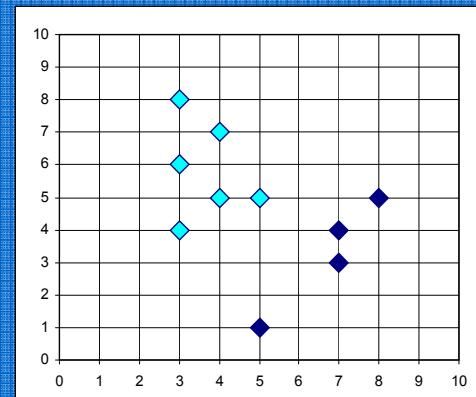
↑ reassign



Update the cluster means



↓ reassign



Update the cluster means

Comments on the *K-Means* Method

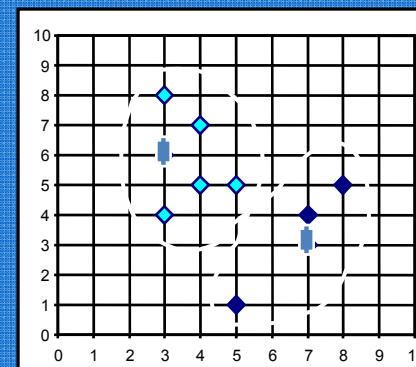
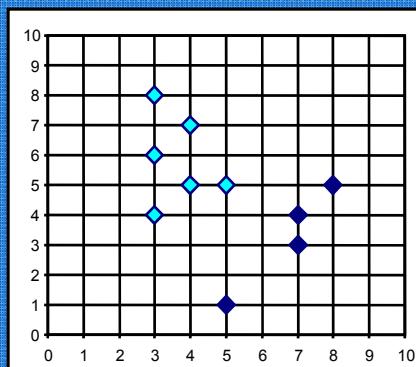
- Strength: *Relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, $k, t \ll n$.
 - Comparing: PAM: $O(k(n-k)^2)$, CLARA: $O(ks^2 + k(n-k))$
- Comment: Often terminates at a *local optimum*. The *global optimum* may be found using techniques such as: *deterministic annealing* and *genetic algorithms*
- Weakness
 - Applicable only when *mean* is defined, then what about categorical data?
 - Need to specify k , the *number* of clusters, in advance
 - Unable to handle noisy data and *outliers*
 - Not suitable to discover clusters with *non-convex shapes*

Variations of the *K-Means* Method

- A few variants of the *k-means* which differ in
 - Selection of the initial k means
 - Dissimilarity calculations
 - Strategies to calculate cluster means
- Handling categorical data: *k-modes* (Huang'98)
 - Replacing means of clusters with modes
 - Using new dissimilarity measures to deal with categorical objects
 - Using a frequency-based method to update modes of clusters
 - A mixture of categorical and numerical data: *k-prototype* method

What Is the Problem of the K-Means Method?

- The k-means algorithm is sensitive to outliers !
 - Since an object with an extremely large value may substantially distort the distribution of the data.
- K-Medoids: Instead of taking the **mean** value of the object in a cluster as a reference point, **medoids** can be used, which is the **most centrally located** object in a cluster.



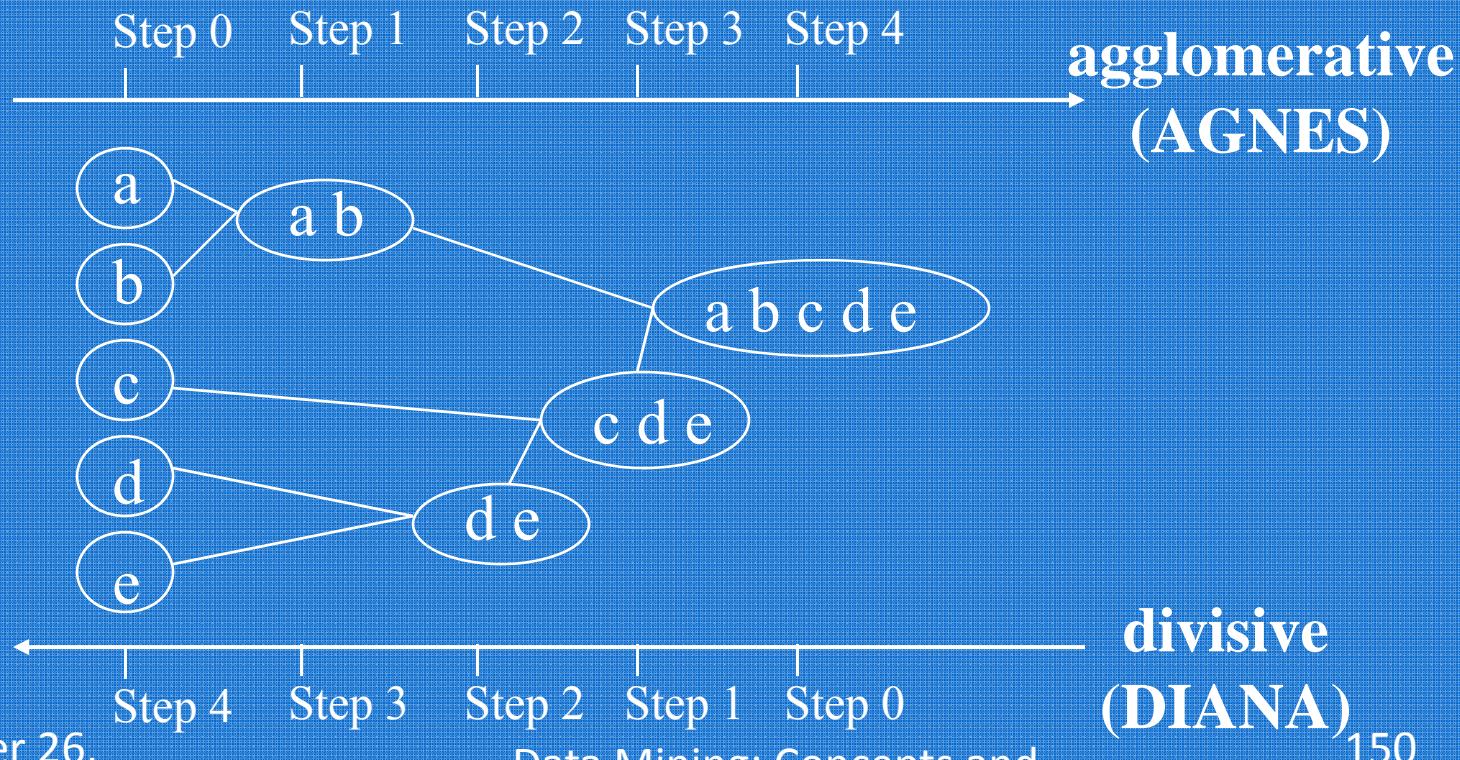
Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Hierarchical Clustering

- Use distance matrix as clustering criteria. This method does not require the number of clusters k as an input, but needs a termination condition



Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary

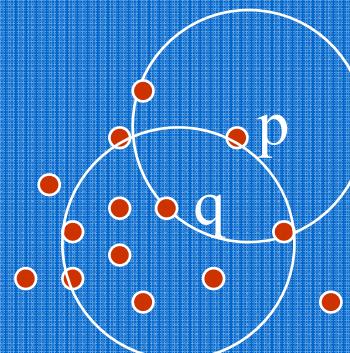


Density-Based Clustering Methods

- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
 - Discover clusters of arbitrary shape
 - Handle noise
 - One scan
 - Need density parameters as termination condition
- Several interesting studies:
 - DBSCAN: Ester, et al. (KDD'96)
 - OPTICS: Ankerst, et al (SIGMOD'99).
 - DENCLUE: Hinneburg & D. Keim (KDD'98)
 - CLIQUE: Agrawal, et al. (SIGMOD'98) (more grid-based)

Density-Based Clustering: Basic Concepts

- Two parameters:
 - Eps : Maximum radius of the neighbourhood
 - $MinPts$: Minimum number of points in an Eps -neighbourhood of that point
- $N_{Eps}(p)$: $\{q \text{ belongs to } D \mid dist(p,q) \leq Eps\}$
- **Directly density-reachable**: A point p is directly density-reachable from a point q w.r.t. Eps , $MinPts$ if
 - p belongs to $N_{Eps}(q)$
 - core point condition:
 $|N_{Eps}(q)| \geq MinPts$



MinPts = 5

Eps = 1 cm

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Grid-Based Clustering Method

- Using multi-resolution grid data structure
- Several interesting methods
 - **STING** (a STatistical INformation Grid approach) by Wang, Yang and Muntz (1997)
 - **WaveCluster** by Sheikholeslami, Chatterjee, and Zhang (VLDB'98)
 - A multi-resolution clustering approach using wavelet method
 - **CLIQUE**: Agrawal, et al. (SIGMOD'98)
 - On high-dimensional data (thus put in the section of clustering high-dimensional data)

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary

Model-Based Clustering

- What is model-based clustering?
 - Attempt to optimize the fit between the given data and some mathematical model
 - Based on the assumption: Data are generated by a mixture of underlying probability distribution
- Typical methods
 - Statistical approach
 - EM (Expectation maximization), AutoClass
 - Machine learning approach
 - COBWEB, CLASSIT
 - Neural network approach
 - SOM (Self-Organizing Feature Map)

Self-Organizing Feature Map (SOM)

- SOMs, also called topological ordered maps, or Kohonen Self-Organizing Feature Map (KSOMs)
- It maps all the points in a high-dimensional source space into a 2 to 3-d target space, s.t., the distance and proximity relationship (i.e., topology) are preserved as much as possible
- Similar to k-means: cluster centers tend to lie in a low-dimensional manifold in the feature space
- Clustering is performed by having several units competing for the current object
 - The unit whose weight vector is closest to the current object wins
 - The winner and its neighbors learn by having their weights adjusted
- SOMs are believed to resemble processing that can occur in the brain
- Useful for visualizing high-dimensional data in 2- or 3-D space

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Clustering High-Dimensional Data

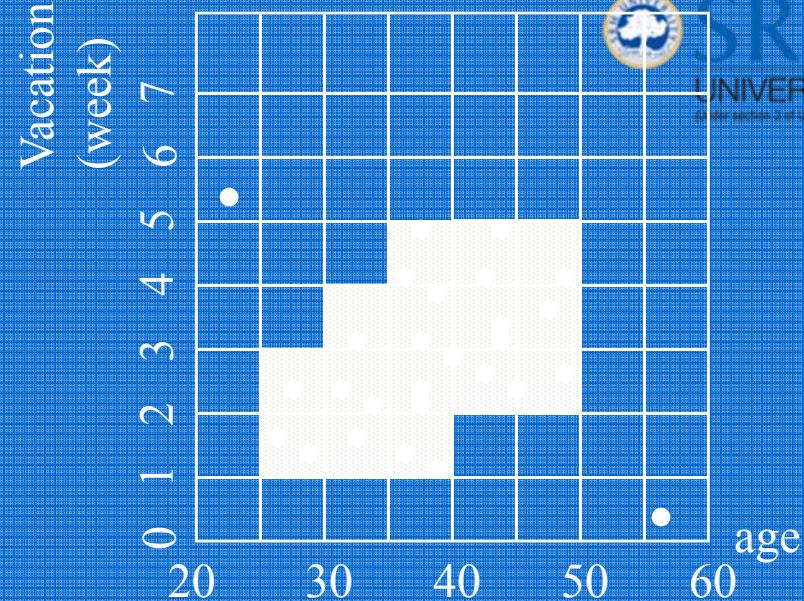
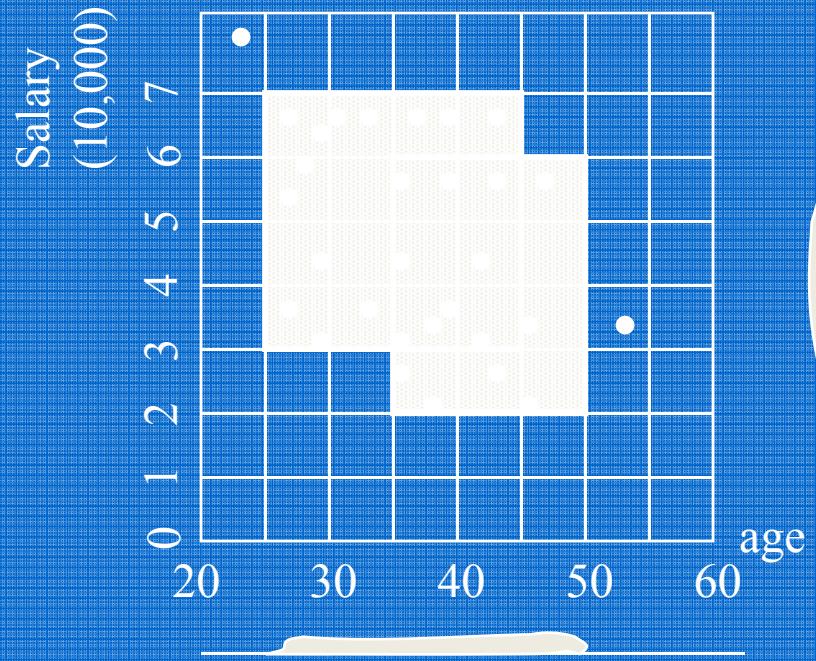
- Clustering high-dimensional data
 - Many applications: text documents, DNA micro-array data
 - Major challenges:
 - Many irrelevant dimensions may mask clusters
 - Distance measure becomes meaningless—due to equi-distance
 - Clusters may exist only in some subspaces
- Methods
 - Feature transformation: only effective if most dimensions are relevant
 - PCA & SVD useful only when features are highly correlated/redundant
 - Feature selection: wrapper or filter approaches
 - useful to find a subspace where the data have nice clusters
 - Subspace-clustering: find clusters in all the possible subspaces
 - CLIQUE, ProClus, and frequent pattern-based clustering

CLIQUE (Clustering In QUEst)

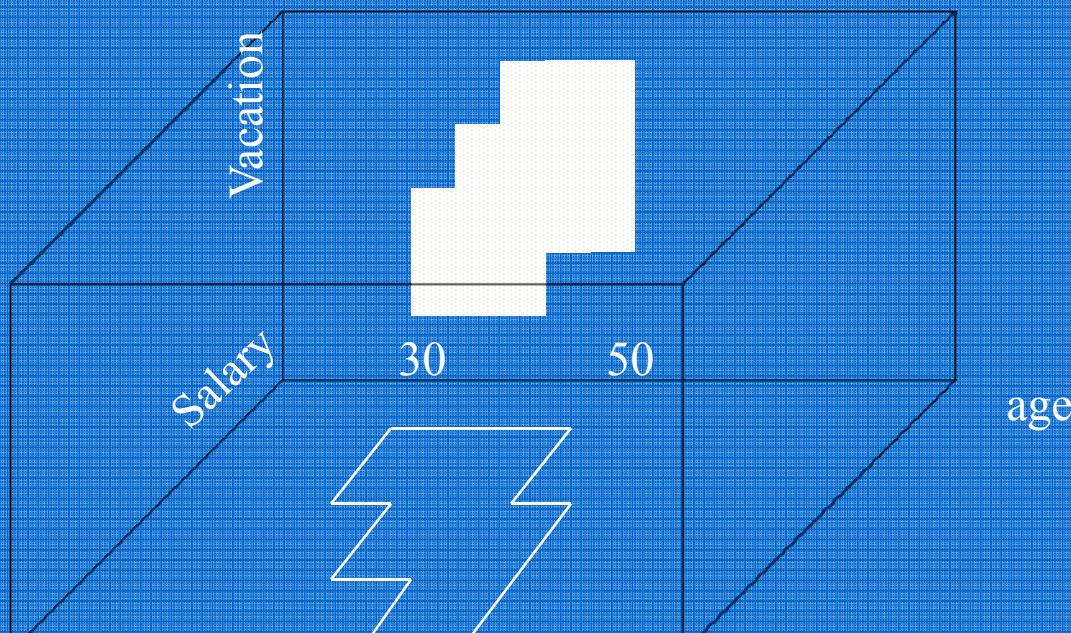
- Agrawal, Gehrke, Gunopulos, Raghavan (SIGMOD'98)
- Automatically identifying subspaces of a high dimensional data space that allow better clustering than original space
- CLIQUE can be considered as both density-based and grid-based
 - It partitions each dimension into the same number of equal length interval
 - It partitions an m-dimensional data space into non-overlapping rectangular units
 - A unit is dense if the fraction of total data points contained in the unit exceeds the input model parameter
 - A cluster is a maximal set of connected dense units within a subspace

CLIQUE: The Major Steps

- Partition the data space and find the number of points that lie inside each cell of the partition.
- Identify the subspaces that contain clusters using the Apriori principle
- Identify clusters
 - Determine dense units in all subspaces of interests
 - Determine connected dense units in all subspaces of interests.
- Generate minimal description for the clusters
 - Determine maximal regions that cover a cluster of connected dense units for each cluster
 - Determination of minimal cover for each cluster



$\tau = 3$

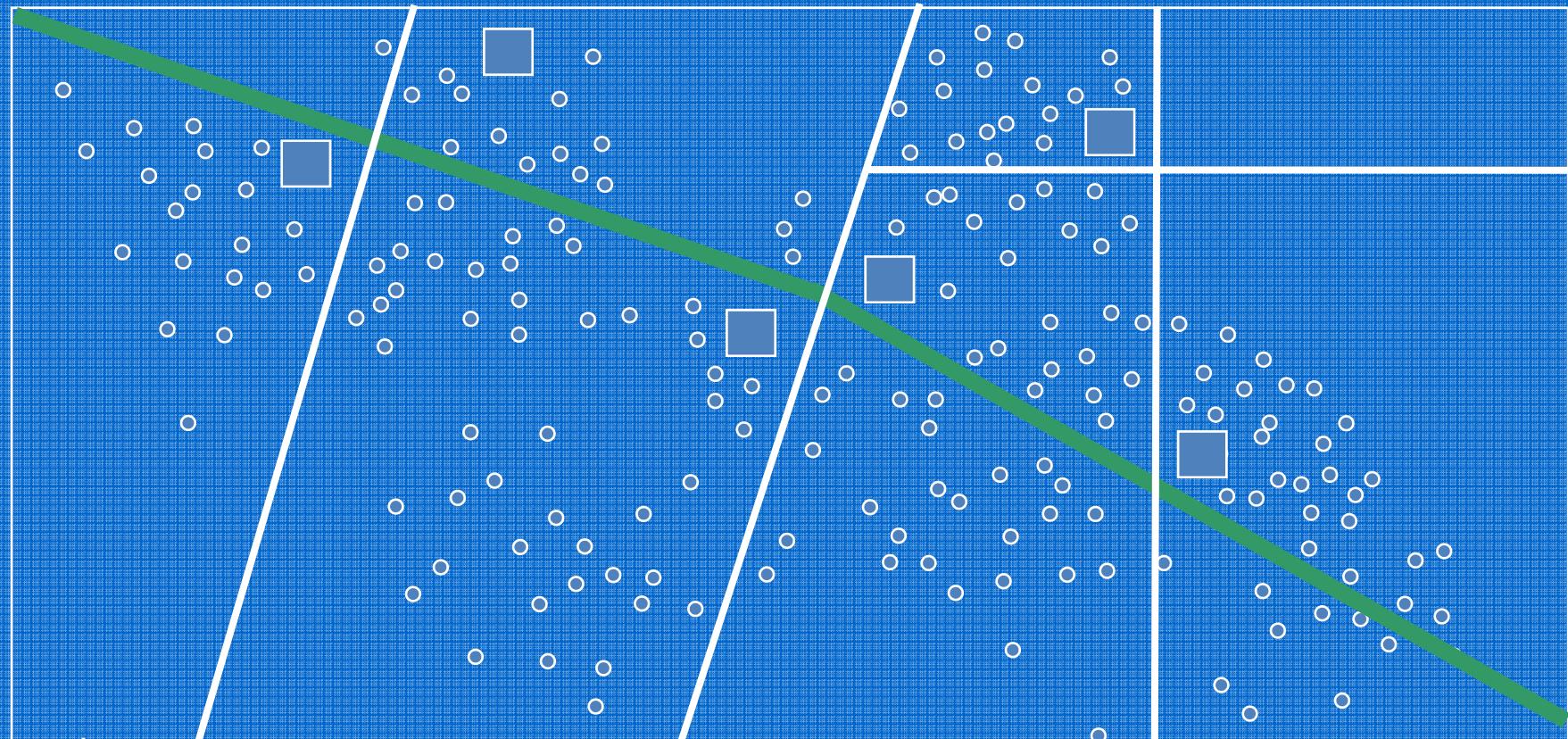


Strength and Weakness of *CLIQUE*

- Strength
 - *automatically* finds subspaces of the highest dimensionality such that high density clusters exist in those subspaces
 - *insensitive* to the order of records in input and does not presume some canonical data distribution
 - scales *linearly* with the size of input and has good scalability as the number of dimensions in the data increases
- Weakness
 - The accuracy of the clustering result may be degraded at the expense of simplicity of the method

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



Cluster Analysis

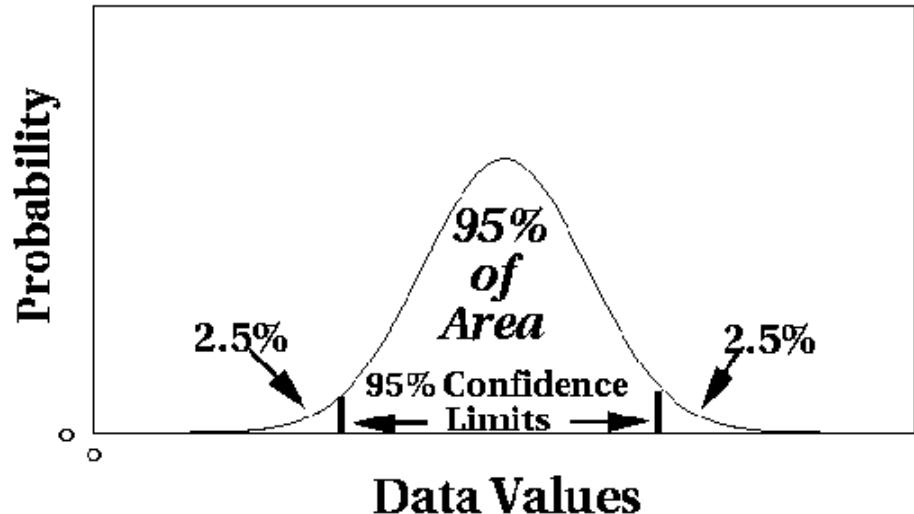
1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



What Is Outlier Discovery?

- What are outliers?
 - The set of objects are considerably dissimilar from the remainder of the data
 - Example: Sports: Michael Jordon, Wayne Gretzky, ...
- Problem: Define and find outliers in large data sets
- Applications:
 - Credit card fraud detection
 - Telecom fraud detection
 - Customer segmentation
 - Medical analysis

Outlier Discovery: Statistical Approaches



- ↗ Assume a model underlying distribution that generates data set (e.g. normal distribution)
- Use discordancy tests depending on
 - data distribution
 - distribution parameter (e.g., mean, variance)
 - number of expected outliers
- Drawbacks
 - most tests are for single attribute
 - In many cases, data distribution may not be known

Outlier Discovery: Distance-Based Approach

- Introduced to counter the main limitations imposed by statistical methods
 - We need multi-dimensional analysis without knowing data distribution
- Distance-based outlier: A DB(p , D)-outlier is an object O in a dataset T such that at least a fraction p of the objects in T lies at a distance greater than D from O
- Algorithms for mining distance-based outliers
 - Index-based algorithm
 - Nested-loop algorithm
 - Cell-based algorithm

Cluster Analysis

1. What is Cluster Analysis?
2. Types of Data in Cluster Analysis
3. A Categorization of Major Clustering Methods
4. Partitioning Methods
5. Hierarchical Methods
6. Density-Based Methods
7. Grid-Based Methods
8. Model-Based Methods
9. Clustering High-Dimensional Data
10. Constraint-Based Clustering
11. Outlier Analysis
12. Summary



Summary

- Cluster analysis groups objects based on their similarity and has wide applications
- Measure of similarity can be computed for various types of data
- Clustering algorithms can be categorized into partitioning methods, hierarchical methods, density-based methods, grid-based methods, and model-based methods
- Outlier detection and analysis are very useful for fraud detection, etc. and can be performed by statistical, distance-based or deviation-based approaches
- There are still lots of research issues on cluster analysis

Review Questions

- State the need for market basket analysis?
- What are the two conditions that make association rule interesting?
- State the two step process of association rule mining?
- Define Apriori property?
- List the techniques to improve the efficiency of Apriori
- What is clustering analysis?
- Give the typical requirements of clustering in data mining?
- What is the difference between symmetric and asymmetric binary variables?
- State the types of data in cluster analysis?

Bibliography

- Data mining concepts and Techniques by Jiawei Han and Micheline Kamber
- R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications
- R. Agrawal and R. Srikant. Fast algorithms for mining association rules. VLDB'94

UNIT-III

Classification and prediction

Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



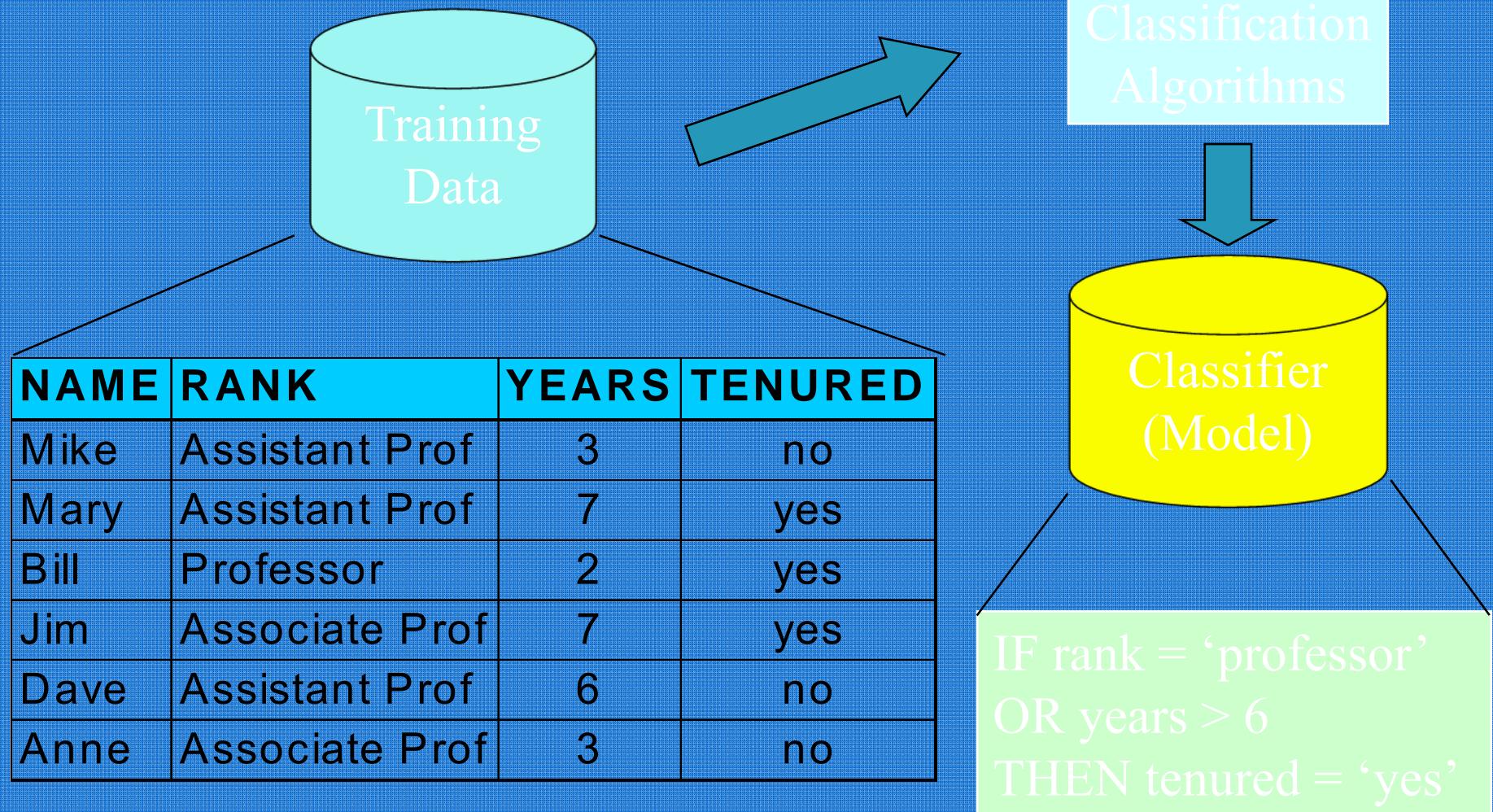
Classification vs. Prediction

- 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
- Prediction
 - models continuous-valued functions, i.e., predicts unknown or missing values
- Typical applications
 - Credit approval
 - Target marketing
 - Medical diagnosis
 - Fraud detection

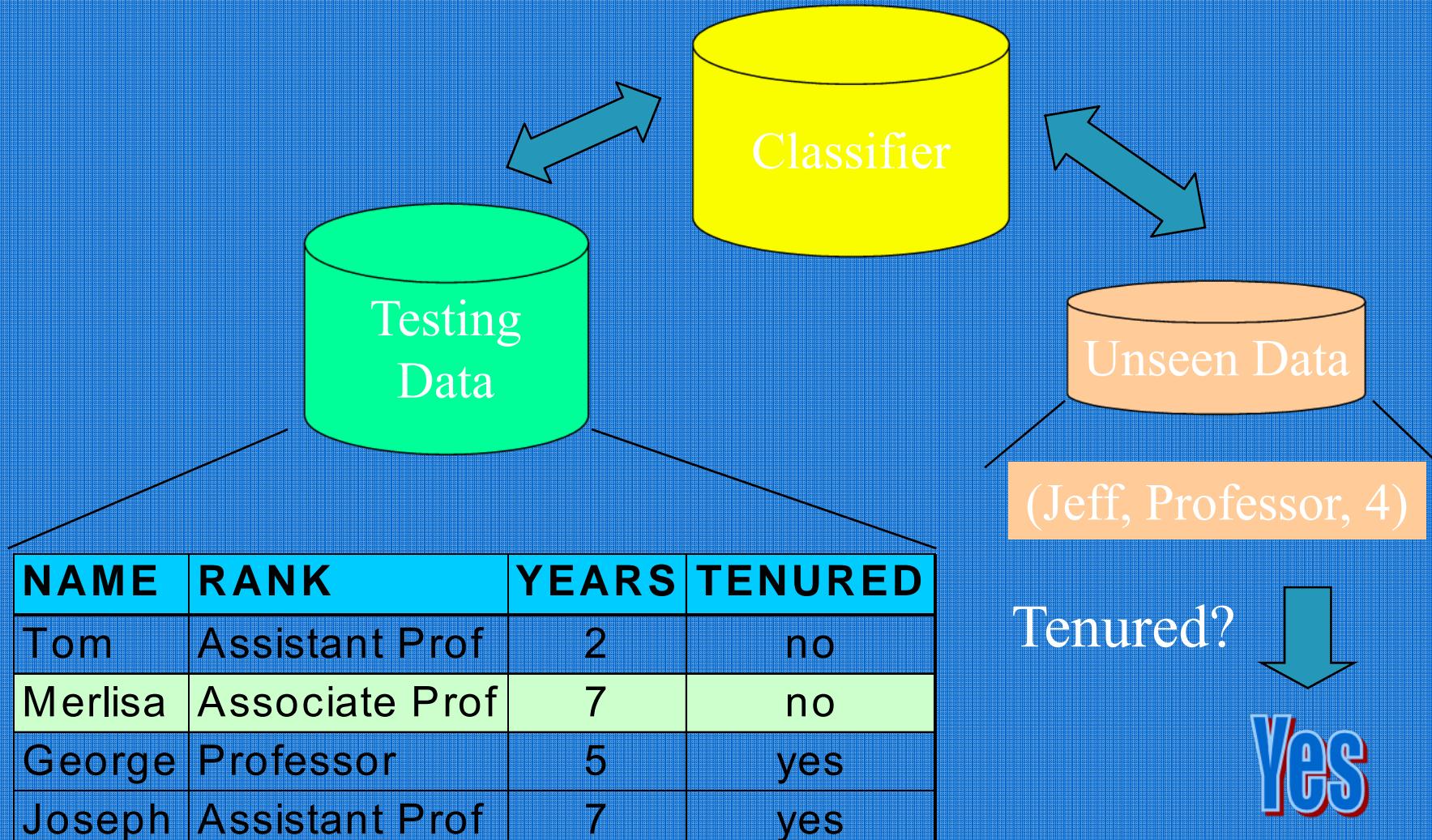
Classification—A Two-Step Process

- Model construction: describing a set of predetermined classes
 - Each tuple/sample is assumed to belong to a predefined class, as determined by the **class label attribute**
 - The set of tuples used for model construction is **training set**
 - The model is represented as classification rules, decision trees, or mathematical formulae
- Model usage: for classifying future or unknown objects
 - Estimate accuracy of the model
 - The known label of test sample is compared with the classified result from the model
 - Accuracy rate is the percentage of test set samples that are correctly classified by the model
 - Test set is independent of training set, otherwise over-fitting will occur
 - If the accuracy is acceptable, use the model to **classify** data tuples whose class labels are not known

Process (1): Model Construction



Process (2): Using the Model in Prediction



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

Chapter 6. Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



Issues: Data Preparation

- Data cleaning
 - Preprocess data in order to reduce noise and handle missing values
- Relevance analysis (feature selection)
 - Remove the irrelevant or redundant attributes
- Data transformation
 - Generalize and/or normalize data

Issues: Evaluating Classification Methods

- Accuracy
 - classifier accuracy: predicting class label
 - predictor accuracy: guessing value of predicted attributes
- Speed
 - time to construct the model (training time)
 - time to use the model (classification/prediction time)
- Robustness: handling noise and missing values
- Scalability: efficiency in disk-resident databases
- Interpretability
 - understanding and insight provided by the model
- Other measures, e.g., goodness of rules, such as decision tree size or compactness of classification rules

Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary

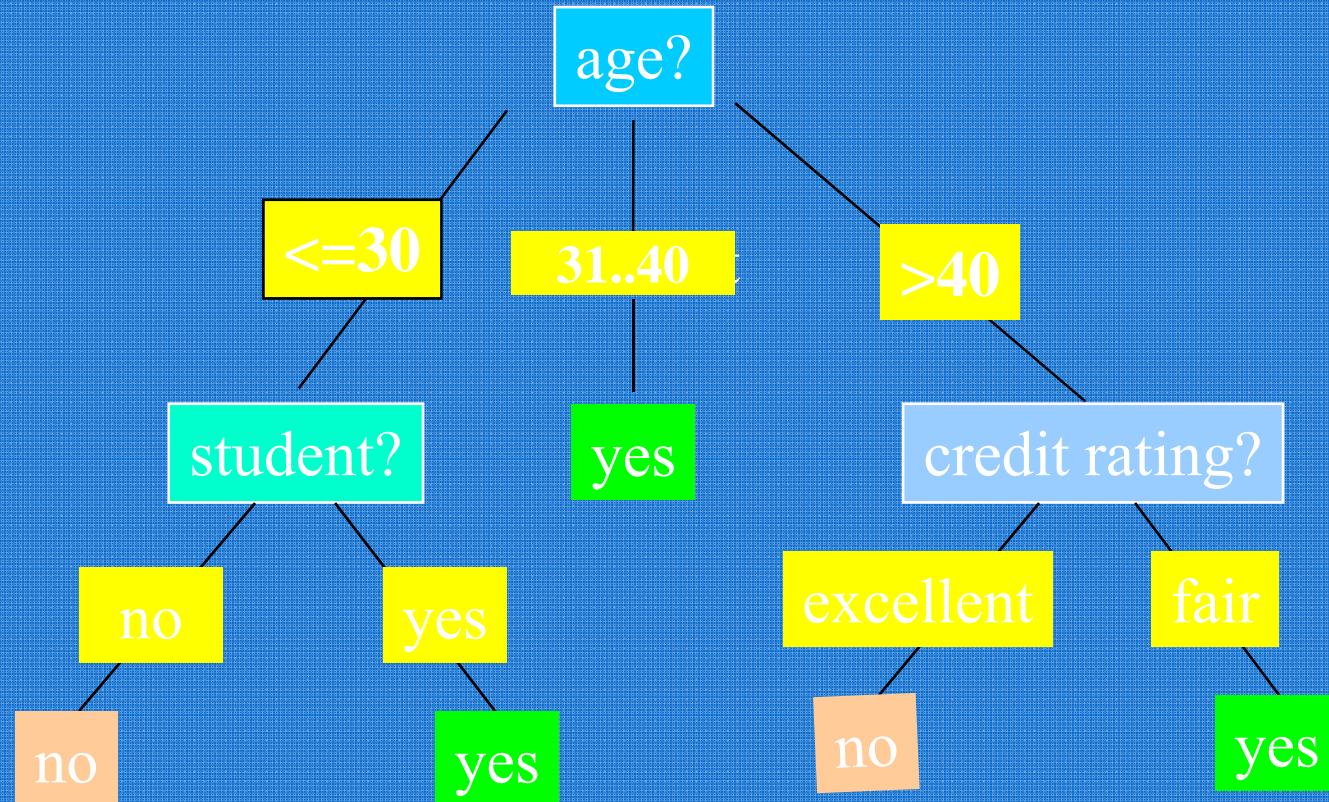


Decision Tree Induction: Training Dataset

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

This follows an example of Quinlan's ID3 (Playing Tennis)

Output: A Decision Tree for “buys_computer”



Algorithm for Decision Tree Induction

- Basic algorithm (a greedy algorithm)
 - Tree is constructed in a **top-down recursive divide-and-conquer manner**
 - At start, all the training examples are at the root
 - Attributes are categorical (if continuous-valued, they are discretized in advance)
 - Examples are partitioned recursively based on selected attributes
 - Test attributes are selected on the basis of a heuristic or statistical measure (e.g., **information gain**)
- Conditions for stopping partitioning
 - All samples for a given node belong to the same class
 - There are no remaining attributes for further partitioning – **majority voting** is employed for classifying the leaf
 - There are no samples left

Classification in Large Databases

- Classification—a classical problem extensively studied by statisticians and machine learning researchers
- Scalability: Classifying data sets with millions of examples and hundreds of attributes with reasonable speed
- Why decision tree induction in data mining?
 - 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

Data Cube-Based Decision-Tree Induction

- Integration of generalization with decision-tree induction
(Kamber et al.'97)
- Classification at primitive concept levels
 - E.g., precise temperature, humidity, outlook, etc.
 - Low-level concepts, scattered classes, bushy classification-trees
 - Semantic interpretation problems
- Cube-based multi-level classification
 - Relevance analysis at multi-levels
 - Information-gain analysis with dimension + level

Classification and Prediction

- What is classification? What is prediction?
 - Issues regarding classification and prediction
 - Classification by decision tree induction
 - Bayesian classification
 - Rule-based classification
 - Classification by back propagation
- 
- Support Vector Machines (SVM)
 - Associative classification
 - Lazy learners (or learning from your neighbors)
 - Other classification methods
 - Prediction
 - Accuracy and error measures
 - Ensemble methods
 - Model selection
 - Summary

Bayesian Classification: Why?

- A statistical classifier: performs *probabilistic prediction*, i.e., predicts class membership probabilities
- Foundation: Based on Bayes' Theorem.
- Performance: A simple Bayesian classifier, *naïve Bayesian classifier*, has comparable performance with decision tree and selected neural network classifiers
- Incremental: Each training example can incrementally increase/decrease the probability that a hypothesis is correct — prior knowledge can be combined with observed data
- Standard: Even when Bayesian methods are computationally intractable, they can provide a standard of optimal decision making against which other methods can be measured

Bayesian Theorem: Basics

- Let \mathbf{X} be a data sample (“*evidence*”): class label is unknown
- Let H be a *hypothesis* that \mathbf{X} belongs to class C
- Classification is to determine $P(H|\mathbf{X})$, the probability that the hypothesis holds given the observed data sample \mathbf{X}
- $P(H)$ (*prior probability*), the initial probability
 - E.g., \mathbf{X} will buy computer, regardless of age, income, ...
- $P(\mathbf{X})$: probability that sample data is observed
- $P(\mathbf{X}|H)$ (*posteriori probability*), the probability of observing the sample \mathbf{X} , given that the hypothesis holds
 - E.g., Given that \mathbf{X} will buy computer, the prob. that \mathbf{X} is 31..40, medium income

Bayesian Theorem

- Given training data \mathbf{X} , *posteriori probability of a hypothesis H*, $P(H|\mathbf{X})$, follows the Bayes theorem

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})}$$

- Informally, this can be written as
posteriori = likelihood x prior/evidence
- Predicts \mathbf{X} belongs to C_2 iff the probability $P(C_i|\mathbf{X})$ is the highest among all the $P(C_k|\mathbf{X})$ for all the k classes
- Practical difficulty: require initial knowledge of many probabilities, significant computational cost

Towards Naïve Bayesian Classifier

- Let D be a training set of tuples and their associated class labels, and each tuple is represented by an n -D attribute vector $\mathbf{X} = (x_1, x_2, \dots, x_n)$
- Suppose there are m classes C_1, C_2, \dots, C_m .
- Classification is to derive the maximum posteriori, i.e., the maximal $P(C_i | \mathbf{X})$
- This can be derived from Bayes' theorem

$$P(C_i | \mathbf{X}) = \frac{P(\mathbf{X} | C_i) P(C_i)}{P(\mathbf{X})}$$

- Since $P(X)$ is constant for all classes, only

$$P(C_i | \mathbf{X}) = P(\mathbf{X} | C_i) P(C_i)$$

needs to be maximized

Naïve Bayesian Classifier: Training Dataset

Class:

C1:buys_computer = 'yes'
C2:buys_computer = 'no'

Data sample

X = (age <=30,
Income = medium,
Student = yes
Credit_rating = Fair)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31...40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31...40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31...40	medium	no	excellent	yes
31...40	high	yes	fair	yes
>40	medium	no	excellent	no

Naïve Bayesian Classifier: An Example

- $P(C_i)$: $P(\text{buys_computer} = \text{"yes"}) = 9/14 = 0.643$
 $P(\text{buys_computer} = \text{"no"}) = 5/14 = 0.357$

- Compute $P(X|C_i)$ for each class

$$P(\text{age} = \text{"}<=30\text{"} | \text{buys_computer} = \text{"yes"}) = 2/9 = 0.222$$

$$P(\text{age} = \text{"}<= 30\text{"} | \text{buys_computer} = \text{"no"}) = 3/5 = 0.6$$

$$P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"yes"}) = 4/9 = 0.444$$

$$P(\text{income} = \text{"medium"} | \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$$

$$P(\text{student} = \text{"yes"} | \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$$

$$P(\text{student} = \text{"yes"} | \text{buys_computer} = \text{"no"}) = 1/5 = 0.2$$

$$P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"yes"}) = 6/9 = 0.667$$

$$P(\text{credit_rating} = \text{"fair"} | \text{buys_computer} = \text{"no"}) = 2/5 = 0.4$$

- $X = (\text{age} \leq 30, \text{income} = \text{medium}, \text{student} = \text{yes}, \text{credit_rating} = \text{fair})$

$$P(X|C_i) : P(X|\text{buys_computer} = \text{"yes"}) = 0.222 \times 0.444 \times 0.667 \times 0.667 = 0.044$$

$$P(X|\text{buys_computer} = \text{"no"}) = 0.6 \times 0.4 \times 0.2 \times 0.4 = 0.019$$

$$P(X|C_i)*P(C_i) : P(X|\text{buys_computer} = \text{"yes"}) * P(\text{buys_computer} = \text{"yes"}) = 0.028$$

$$P(X|\text{buys_computer} = \text{"no"}) * P(\text{buys_computer} = \text{"no"}) = 0.007$$

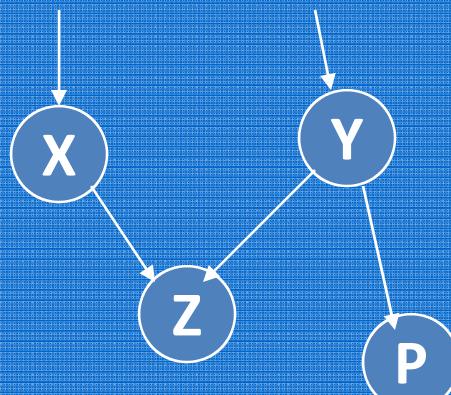
Therefore, X belongs to class ("buys_computer = yes")

Naïve Bayesian Classifier: Comments

- Advantages
 - Easy to implement
 - Good results obtained in most of the cases
- Disadvantages
 - Assumption: class conditional independence, therefore loss of accuracy
 - Practically, dependencies exist among variables
 - E.g., hospitals: patients: Profile: age, family history, etc.
Symptoms: fever, cough etc., Disease: lung cancer, diabetes, etc.
 - Dependencies among these cannot be modeled by Naïve Bayesian Classifier
- How to deal with these dependencies?
 - Bayesian Belief Networks

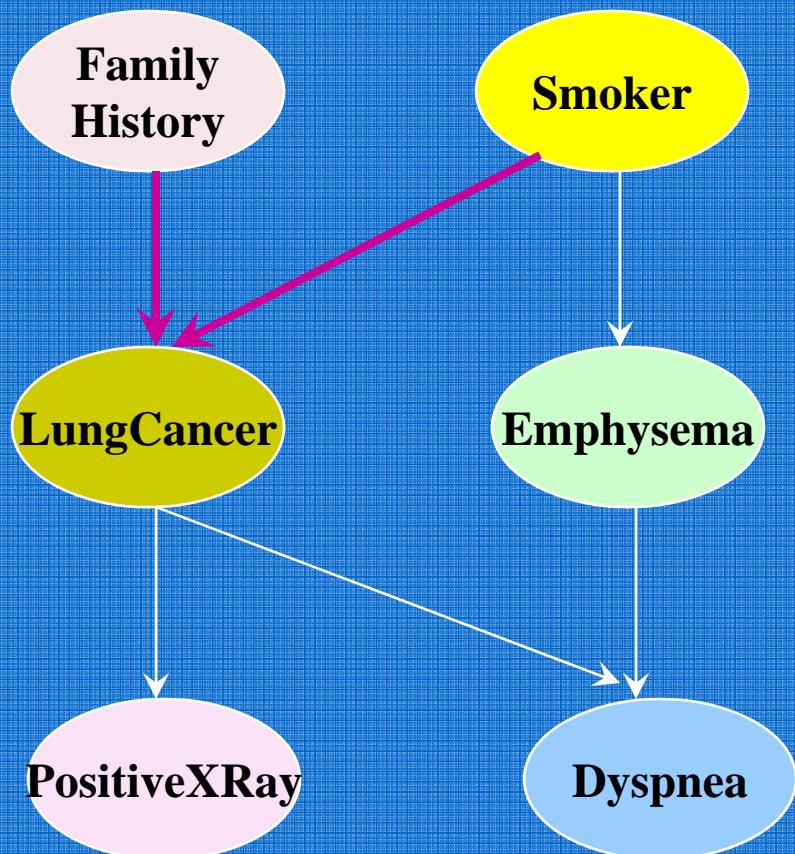
Bayesian Belief Networks

- Bayesian belief network allows a *subset* of the variables conditionally independent
- A graphical model of causal relationships
 - Represents dependency among the variables
 - Gives a specification of joint probability distribution



- Nodes: random variables
- Links: dependency
- X and Y are the parents of Z, and Y is the parent of P
- No dependency between Z and P
- Has no loops or cycles

Bayesian Belief Network: An Example



Bayesian Belief Networks

The **conditional probability table (CPT)** for variable LungCancer:

	(FH, S)	(FH, ~S)	(~FH, S)	(~FH, ~S)
LC	0.8	0.5	0.7	0.1
~LC	0.2	0.5	0.3	0.9

CPT shows the conditional probability for each possible combination of its parents

Derivation of the probability of a particular combination of values of X, from CPT:

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | Parents(Y_i))$$

Training Bayesian Networks

- Several scenarios:
 - Given both the network structure and all variables observable: *learn only the CPTs*
 - Network structure known, some hidden variables: *gradient descent* (greedy hill-climbing) method, analogous to neural network learning
 - Network structure unknown, all variables observable: search through the model space to *reconstruct network topology*
 - Unknown structure, all hidden variables: No good algorithms known for this purpose
- Ref. D. Heckerman: Bayesian networks for data mining

Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



Using IF-THEN Rules for Classification

- Represent the knowledge in the form of **IF-THEN** rules

R: IF *age* = youth AND *student* = yes THEN *buys_computer* = yes

- Rule antecedent/precondition vs. rule consequent

- Assessment of a rule: *coverage* and *accuracy*

- n_{covers} = # of tuples covered by R

- n_{correct} = # of tuples correctly classified by R

$$\text{coverage}(R) = n_{\text{covers}} / |D| \quad /* D: training data set */$$

$$\text{accuracy}(R) = n_{\text{correct}} / n_{\text{covers}}$$

- If more than one rule is triggered, need **conflict resolution**

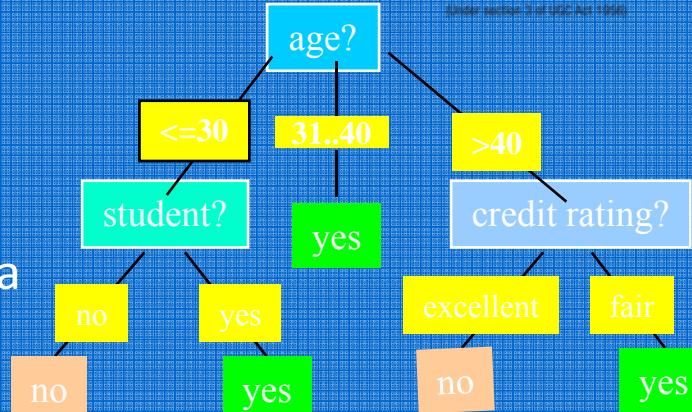
- Size ordering: assign the highest priority to the triggering rules that has the “toughest” requirement (i.e., with the *most attribute test*)

- Class-based ordering: decreasing order of *prevalence* or *misclassification cost per class*

- Rule-based ordering (**decision list**): rules are organized into one long priority list, according to some measure of rule quality or by experts

Rule Extraction from a Decision Tree

- Rules are easier to understand than large trees
- One rule is created for each path from the root to a leaf
- Each attribute-value pair along a path forms a conjunction: the leaf holds the class prediction
- Rules are mutually exclusive and exhaustive
- Example: Rule extraction from our *buys_computer* decision-tree



IF *age* = young AND *student* = no

THEN *buys_computer* = no

IF *age* = young AND *student* = yes

THEN *buys_computer* = yes

IF *age* = mid-age

THEN *buys_computer* = yes

IF *age* = old AND *credit_rating* = excellent THEN *buys_computer* = yes

IF *age* = young AND *credit_rating* = fair THEN *buys_computer* = no

Rule Extraction from the Training Data

- Sequential covering algorithm: Extracts rules directly from training data
- Typical sequential covering algorithms: FOIL, AQ, CN2, RIPPER
- Rules are learned *sequentially*, each for a given class C_i will cover many tuples of C_i but none (or few) of the tuples of other classes
- Steps:
 - Rules are learned one at a time
 - Each time a rule is learned, the tuples covered by the rules are removed
 - The process repeats on the remaining tuples unless *termination condition*, e.g., when no more training examples or when the quality of a rule returned is below a user-specified threshold
- Comp. w. decision-tree induction: learning a set of rules *simultaneously*

Classification and Prediction

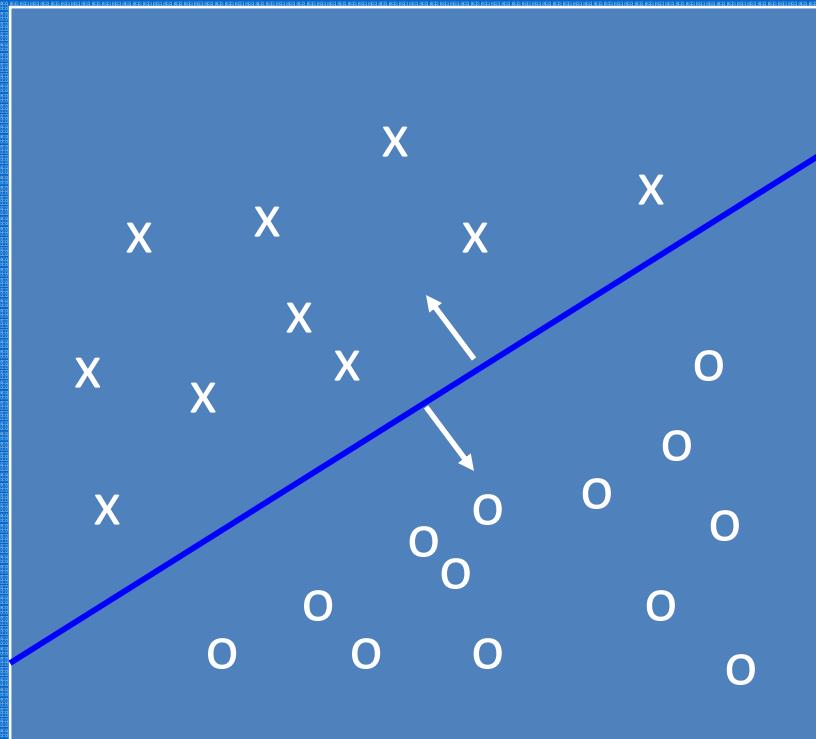
- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



Classification: A Mathematical Mapping

- Classification:
 - predicts categorical class labels
- E.g., Personal homepage classification
 - $x_i = (x_1, x_2, x_3, \dots)$, $y_i = +1$ or -1
 - x_1 : # of a word “homepage”
 - x_2 : # of a word “welcome”
- Mathematically
 - $x \in X = \mathbb{R}^n$, $y \in Y = \{+1, -1\}$
 - We want a function $f: X \rightarrow Y$

Linear Classification



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

Discriminative Classifiers

- Advantages
 - prediction accuracy is generally high
 - As compared to Bayesian methods – in general
 - robust, works when training examples contain errors
 - fast evaluation of the learned target function
 - Bayesian networks are normally slow
- Criticism
 - long training time
 - difficult to understand the learned function (weights)
 - Bayesian networks can be used easily for pattern discovery
 - not easy to incorporate domain knowledge
 - Easy in the form of priors on the data or distributions

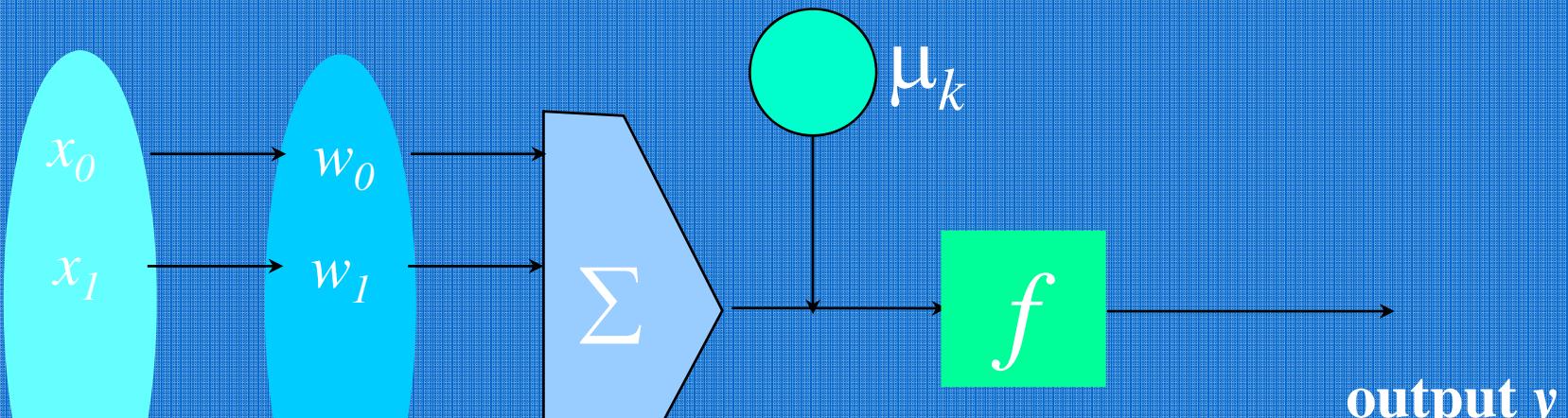
Classification by Backpropagation

- Backpropagation: A **neural network** learning algorithm
- Started by psychologists and neurobiologists to develop and test computational analogues of neurons
- A neural network: A set of connected input/output units where each connection has a **weight** associated with it
- During the learning phase, the **network learns by adjusting the weights** so as to be able to predict the correct class label of the input tuples
- Also referred to as **connectionist learning** due to the connections between units

Neural Network as a Classifier

- Weakness
 - Long training time
 - Require a number of 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 a wide array of real-world data
 - Algorithms are inherently parallel
 - Techniques have recently been developed for the extraction of rules from trained neural networks

A Neuron (= a perceptron)



Input vector x	weight vector w	weighted sum	Activation function
------------------------------------	-------------------------------------	---------------------	----------------------------

For Example

$$y = \text{sign}\left(\sum_{i=0}^n w_i x_i + \mu_k\right)$$

- The n -dimensional input vector x is mapped into variable y by means of the scalar product and a nonlinear function mapping

A Multi-Layer Feed-Forward Neural Network

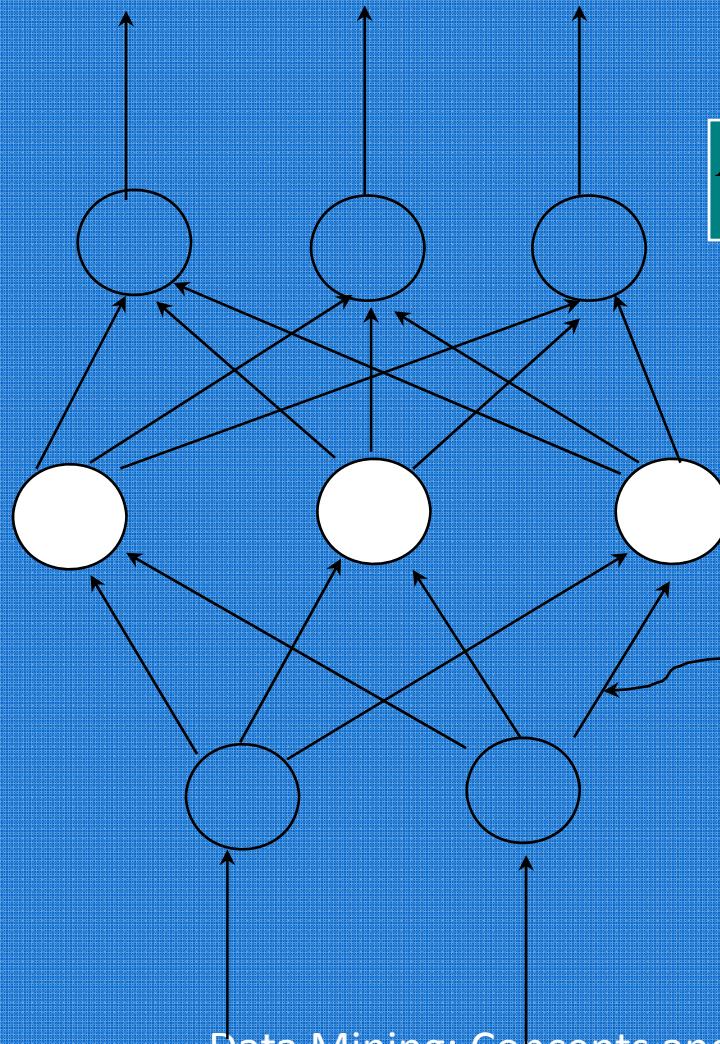
Output vector

Output layer

Hidden layer

Input layer

Input vector: X



$$Err_j = O_j(1 - O_j) \sum_k Err_k w_{jk}$$

$$\theta_j = \theta_j + (l) Err_j$$

$$w_{ij} = w_{ij} + (l) Err_j O_i$$

$$Err_j = O_j(1 - O_j)(T_j - O_j)$$

$$w_{ij}$$

$$O_j = \frac{1}{1 + e^{-I_j}}$$

$$I_j = \sum_i w_{ij} O_i + \theta_j$$

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** in that 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

Defining a Network Topology

- First decide the **network topology**: # of units in the *input layer*, # of *hidden layers* (if > 1), # of units in *each hidden layer*, and # of units in the *output layer*
- Normalizing the input values for each attribute measured in the training tuples to [0.0—1.0]
- One **input** unit per domain value, each initialized to 0
- **Output**, if for classification and more than two classes, one output unit per class is used
- Once a network has been trained and its accuracy is **unacceptable**, repeat the training process with a *different network topology* or a *different set of initial weights*

Backpropagation

- Iteratively process a set of training tuples & compare the network's prediction with the actual known target value
- For each training tuple, the weights are modified to **minimize the mean squared error** between the network's prediction and the actual target value
- Modifications are made in the "**backwards**" direction: from the output layer, through each hidden layer down to the first hidden layer, hence "**backpropagation**"
- Steps
 - Initialize weights (to small random #s) and biases in the network
 - Propagate the inputs forward (by applying activation function)
 - Backpropagate the error (by updating weights and biases)
 - Terminating condition (when error is very small, etc.)

Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



Associative Classification

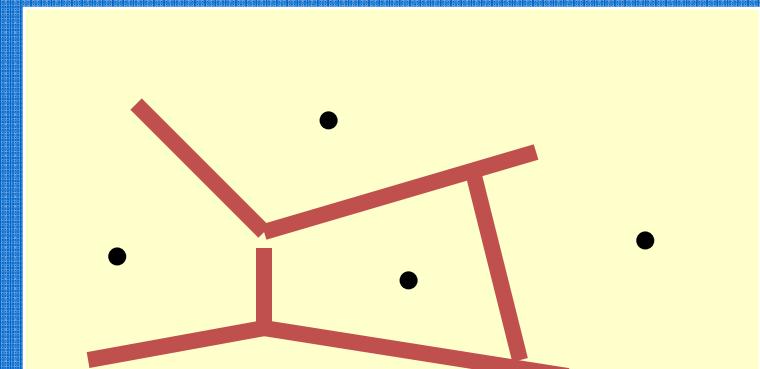
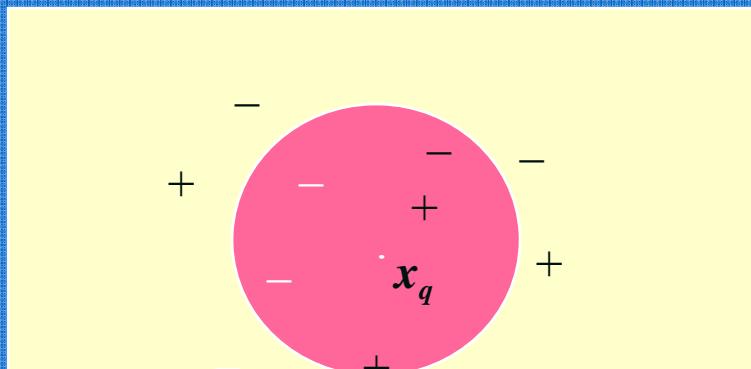
- Associative classification
 - Association rules are generated and analyzed for use in classification
 - Search for strong associations between frequent patterns (conjunctions of attribute-value pairs) and class labels
 - Classification: Based on evaluating a set of rules in the form of
$$P_1 \wedge p_2 \dots \wedge p_l \rightarrow "A_{\text{class}} = C" \text{ (conf, sup)}$$
- Why effective?
 - It explores highly confident associations among multiple attributes and may overcome some constraints introduced by decision-tree induction, which considers only one attribute at a time
 - In many studies, associative classification has been found to be more accurate than some traditional classification methods, such as C4.5

Typical Associative Classification Methods

- CBA (Classification By Association: Liu, Hsu & Ma, KDD'98)
 - Mine association possible rules in the form of
 - Cond-set (a set of attribute-value pairs) → class label
 - Build classifier: Organize rules according to decreasing precedence based on confidence and then support
- CMAR (Classification based on Multiple Association Rules: Li, Han, Pei, ICDM'01)
 - Classification: Statistical analysis on multiple rules
- CPAR (Classification based on Predictive Association Rules: Yin & Han, SDM'03)
 - Generation of predictive rules (FOIL-like analysis)
 - High efficiency, accuracy similar to CMAR
- RCBT (Mining top- k covering rule groups for gene expression data, Cong et al. SIGMOD'05)
 - Explore high-dimensional classification, using top- k rule groups
 - Achieve high classification accuracy and high run-time efficiency

The k -Nearest Neighbor Algorithm

- All instances correspond to points in the n-D space
- The nearest neighbor are defined in terms of Euclidean distance, $\text{dist}(X_1, X_2)$
- Target function could be discrete- or real- valued
- For discrete-valued, k -NN returns the most common value among the k training examples nearest to x_q
- Voronoi diagram: the decision surface induced by 1-NN for a typical set of training examples



Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



What Is Prediction?

- (Numerical) prediction is similar to classification
 - construct a model
 - use model to predict continuous or ordered value for a given input
- Prediction is different from classification
 - Classification refers to predict categorical class label
 - Prediction models continuous-valued functions
- Major method for prediction: regression
 - model the relationship between one or more *independent* or **predictor** variables and a *dependent* or **response** variable
- Regression analysis
 - Linear and multiple regression
 - Non-linear regression
 - Other regression methods: generalized linear model, Poisson regression, log-linear models, regression trees

Linear Regression

- Linear regression: involves a response variable y and a single predictor variable x

$$y = w_0 + w_1 x$$

where w_0 (y -intercept) and w_1 (slope) are regression coefficients

- Method of least squares: estimates the best-fitting straight line

$$w_1 = \frac{\sum_{i=1}^{|D|} (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{|D|} (x_i - \bar{x})^2} \quad w_0 = \bar{y} - w_1 \bar{x}$$

- Multiple linear regression: involves more than one predictor variable
 - Training data is of the form $(\mathbf{X}_1, y_1), (\mathbf{X}_2, y_2), \dots, (\mathbf{X}_{|D|}, y_{|D|})$
 - Ex. For 2-D data, we may have: $y = w_0 + w_1 x_1 + w_2 x_2$
 - Solvable by extension of least square method or using SAS, S-Plus
 - Many nonlinear functions can be transformed into the above

Nonlinear Regression

- Some nonlinear models can be modeled by a polynomial function
- A polynomial regression model can be transformed into linear regression model. For example,

$$y = w_0 + w_1 x + w_2 x^2 + w_3 x^3$$

convertible to linear with new variables: $x_2 = x^2$, $x_3 = x^3$

$$y = w_0 + w_1 x + w_2 x_2 + w_3 x_3$$

- Other functions, such as power function, can also be transformed to linear model
- Some models are intractable nonlinear (e.g., sum of exponential terms)
 - possible to obtain least square estimates through extensive calculation on more complex formulae

Other Regression-Based Models

- Generalized linear model:
 - Foundation on which linear regression can be applied to modeling categorical response variables
 - Variance of y is a function of the mean value of y , not a constant
 - Logistic regression: models the prob. of some event occurring as a linear function of a set of predictor variables
 - Poisson regression: models the data that exhibit a Poisson distribution
- Log-linear models: (for categorical data)
 - Approximate discrete multidimensional prob. distributions
 - Also useful for data compression and smoothing
- Regression trees and model trees
 - Trees to predict continuous values rather than class labels

Regression Trees and Model Trees

- Regression tree: proposed in CART system (Breiman et al. 1984)
 - CART: Classification And Regression Trees
 - Each leaf stores a *continuous-valued prediction*
 - It is the *average value of the predicted attribute* for the training tuples that reach the leaf
- Model tree: proposed by Quinlan (1992)
 - Each leaf holds a regression model—a multivariate linear equation for the predicted attribute
 - A more general case than regression tree
- Regression and model trees tend to be more accurate than linear regression when the data are not represented well by a simple linear model

Predictive Modeling in Multidimensional Databases

- Predictive modeling: Predict data values or construct generalized linear models based on the database data
- One can only predict value ranges or category distributions
- Method outline:
 - Minimal generalization
 - Attribute relevance analysis
 - Generalized linear model construction
 - Prediction
- Determine the major factors which influence the prediction
 - Data relevance analysis: uncertainty measurement, entropy analysis, expert judgement, etc.
- Multi-level prediction: drill-down and roll-up analysis

Boosting

- Analogy: Consult several doctors, based on a combination of weighted diagnoses—weight assigned based on the previous diagnosis accuracy
- How boosting works?
 - Weights are assigned to each training tuple
 - A series of k classifiers is iteratively learned
 - After a classifier M_i is learned, the weights are updated to allow the subsequent classifier, M_{i+1} , to pay more attention to the training tuples that were misclassified by M_i
 - The final M^* combines the votes of each individual classifier, where the weight of each classifier's vote is a function of its accuracy
- The boosting algorithm can be extended for the prediction of continuous values
- Comparing with bagging: boosting tends to achieve greater accuracy, but it also risks overfitting the model to misclassified data

Classification and Prediction

- What is classification? What is prediction?
- Issues regarding classification and prediction
- Classification by decision tree induction
- Bayesian classification
- Rule-based classification
- Classification by back propagation
- Support Vector Machines (SVM)
- Associative classification
- Lazy learners (or learning from your neighbors)
- Other classification methods
- Prediction
- Accuracy and error measures
- Ensemble methods
- Model selection
- Summary



Summary (I)

- Classification and prediction are two forms of data analysis that can be used to extract models describing important data classes or to predict future data trends.
- Effective and scalable methods have been developed for decision trees induction, Naive Bayesian classification, Bayesian belief network, rule-based classifier, Backpropagation, Support Vector Machine (SVM), associative classification, nearest neighbor classifiers, and case-based reasoning, and other classification methods such as genetic algorithms, rough set and fuzzy set approaches.
- Linear, nonlinear, and generalized linear models of regression can be used for prediction. Many nonlinear problems can be converted to linear problems by performing transformations on the predictor variables. Regression trees and model trees are also used for prediction.

Summary (II)

- Stratified k-fold cross-validation is a recommended method for accuracy estimation. Bagging and boosting can be used to increase overall accuracy by learning and combining a series of individual models.
- Significance tests and ROC curves are useful for model selection
- There have been numerous comparisons of the different classification and prediction methods, and the matter remains a research topic
- No single method has been found to be superior over all others for all data sets
- Issues such as accuracy, training time, robustness, interpretability, and scalability must be considered and can involve trade-offs, further complicating the quest for an overall superior method

Review Questions

- How does classification works?
- How is prediction different form classification ?
- Define Data cleaning?
- List the criteria involved in comparing and evaluating the classification and prediction methods?
- What are Bayesian classifier?
- State Bayes theorem
- Define Back propagation and how does it work?
- State Rule pruning?
- What if we would like to predict a continuous value ,rather than a categorical label?
- State linear regression?
- State polynomial regression?
- Give a note on bootstrap method?
- What is boosting ?State why it may improve the accuracy of decision tree induction?

Bibliography

- Data mining concepts and Techniques by Jiawei Han and Micheline Kamber
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley & Sons, 2003

UNIT IV

Data Mining: Concepts and
Techniques

- Mining data streams
- Mining time-series data
- Mining sequence patterns in transactional databases
- Mining sequence patterns in biological data



Mining Data Streams



- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues

Characteristics of Data Streams

- Data Streams
 - Data streams—continuous, ordered, changing, fast, huge amount
 - Traditional DBMS—data stored in finite, persistent data sets
- Characteristics
 - Huge volumes of continuous data, possibly infinite
 - Fast changing and requires fast, real-time response
 - Data stream captures nicely our data processing needs of today
 - Random access is expensive—single scan algorithm (*can only have one look*)
 - Store only the summary of the data seen thus far
 - Most stream data are at pretty low-level or multi-dimensional in nature, needs multi-level and multi-dimensional processing

Stream Data Applications

- Telecommunication calling records
- Business: credit card transaction flows
- Network monitoring and traffic engineering
- Financial market: stock exchange
- Engineering & industrial processes: power supply & manufacturing
- Sensor, monitoring & surveillance: video streams, RFIDs
- Security monitoring
- Web logs and Web page click streams
- Massive data sets (even saved but random access is too expensive)

DBMS versus DSMS

- Persistent relations
- One-time queries
- Random access
- “Unbounded” disk store
- Only current state matters
- No real-time services
- Relatively low update rate
- Data at any granularity
- Assume precise data
- Access plan determined by query processor, physical DB design
- Transient streams
- Continuous queries
- Sequential access
- Bounded main memory
- Historical data is important
- Real-time requirements
- Possibly multi-GB arrival rate
- Data at fine granularity
- Data stale/imprecise
- Unpredictable/variable data arrival and characteristics

Ack. From Motwani's PODS tutorial slides

Mining Data Streams

- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues

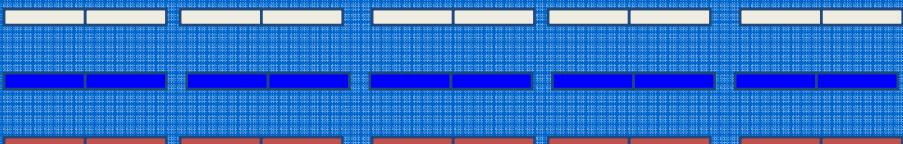


Architecture: Stream Query Processing

SDMS (Stream Data Management System)

Continuous Query

Multiple streams



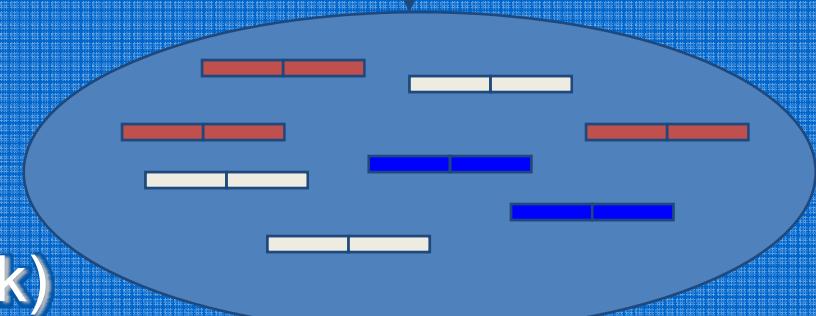
Scratch Space

(Main memory and/or Disk)

User/Application

Stream Query Processor

Results



Challenges of Stream Data Processing

- Multiple, continuous, rapid, time-varying, ordered streams
- Main memory computations
- Queries are often continuous
 - Evaluated continuously as stream data arrives
 - Answer updated over time
- Queries are often complex
 - Beyond element-at-a-time processing
 - Beyond stream-at-a-time processing
 - Beyond relational queries (scientific, data mining, OLAP)
- Multi-level/multi-dimensional processing and data mining
 - Most stream data are at low-level or multi-dimensional in nature

Processing Stream Queries

- Query types
 - One-time query vs. continuous query (being evaluated continuously as stream continues to arrive)
 - Predefined query vs. ad-hoc query (issued on-line)
- Unbounded memory requirements
 - For real-time response, main memory algorithm should be used
 - Memory requirement is unbounded if one will join future tuples
- Approximate query answering
 - With bounded memory, it is not always possible to produce exact answers
 - High-quality approximate answers are desired
 - Data reduction and synopsis construction methods
 - Sketches, random sampling, histograms, wavelets, etc.

Methodologies for Stream Data Processing

- Major challenges
 - Keep track of a large universe, e.g., pairs of IP address, not ages
- Methodology
 - Synopses (trade-off between accuracy and storage)
 - Use *synopsis data structure*, much smaller ($O(\log^k N)$ space) than their base data set ($O(N)$ space)
 - Compute an *approximate answer* within a *small error range* (factor ϵ of the actual answer)
- Major methods
 - Random sampling
 - Histograms
 - Sliding windows
 - Multi-resolution model
 - Sketches
 - Radomized algorithms

Stream Data Mining vs. Stream Querying

- Stream mining—A more challenging task in many cases
 - It shares most of the difficulties with stream querying
 - But often requires less “precision”, e.g., no join, grouping, sorting
 - Patterns are hidden and more general than querying
 - It may require exploratory analysis
 - Not necessarily continuous queries
- Stream data mining tasks
 - Multi-dimensional on-line analysis of streams
 - Mining outliers and unusual patterns in stream data
 - Clustering data streams
 - Classification of stream data

Mining Data Streams

- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues



Challenges for Mining Dynamics in Data Streams

- Most stream data are at pretty low-level or multi-dimensional in nature: needs ML/MD processing
- Analysis requirements
 - Multi-dimensional trends and unusual patterns
 - Capturing important changes at multi-dimensions/levels
 - Fast, real-time detection and response
 - Comparing with data cube: Similarity and differences
- Stream (data) cube or stream OLAP: Is this feasible?
 - Can we implement it efficiently?

A Stream Cube Architecture

- A tilted time frame
 - Different time granularities
 - second, minute, quarter, hour, day, week, ...
- Critical layers
 - Minimum interest layer (m-layer)
 - Observation layer (o-layer)
 - User: watches at o-layer and occasionally needs to drill-down down to m-layer
- Partial materialization of stream cubes
 - Full materialization: too space and time consuming
 - No materialization: slow response at query time
 - Partial materialization: what do we mean “partial”?

Mining Data Streams

- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues



Frequent Patterns for Stream Data

- Frequent pattern mining is valuable in stream applications
 - e.g., network intrusion mining (Dokas, et al'02)
- Mining precise freq. patterns in stream data: unrealistic
 - Even store them in a compressed form, such as FPtree
- How to mine frequent patterns with good approximation?
 - Approximate frequent patterns (Manku & Motwani VLDB'02)
 - Keep only current frequent patterns? No changes can be detected
- Mining evolution freq. patterns (C. Giannella, J. Han, X. Yan, P.S. Yu, 2003)
 - Use tilted time window frame
 - Mining evolution and dramatic changes of frequent patterns
- Space-saving computation of frequent and top-k elements (Metwally, Agrawal, and El Abbadi, ICDT'05)

Mining Approximate Frequent Patterns

- Mining precise freq. patterns in stream data: **unrealistic**
 - Even store them in a compressed form, such as FPtree
- Approximate answers are often sufficient (e.g., trend/pattern analysis)
 - Example: a router is interested in all flows:
 - whose **frequency** is at least **1% (σ)** of the entire traffic stream seen so far
 - and feels that **$1/10$ of σ ($\varepsilon = 0.1\%$) error** is comfortable
- How to mine frequent patterns with good approximation?
 - Lossy Counting Algorithm (Manku & Motwani, VLDB'02)
 - Major ideas: not tracing items until it becomes frequent
 - Adv: guaranteed error bound
 - Disadv: keep a large set of traces

Mining Data Streams

- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues



Classification for Dynamic Data Streams

- Decision tree induction for stream data classification
 - VFDT (Very Fast Decision Tree)/CVFDT (Domingos, Hulten, Spencer, KDD00/KDD01)
- Is decision-tree good for modeling fast changing data, e.g., stock market analysis?
- Other stream classification methods
 - Instead of decision-trees, consider other models
 - Naïve Bayesian
 - Ensemble (Wang, Fan, Yu, Han. KDD'03)
 - K-nearest neighbors (Aggarwal, Han, Wang, Yu. KDD'04)
 - Tilted time framework, incremental updating, dynamic maintenance, and model construction
 - Comparing of models to find changes

Hoeffding Tree

- With high probability, classifies tuples the same
- Only uses small sample
 - Based on Hoeffding Bound principle
- Hoeffding Bound (Additive Chernoff Bound)

r: random variable

R: range of r

n: # independent observations

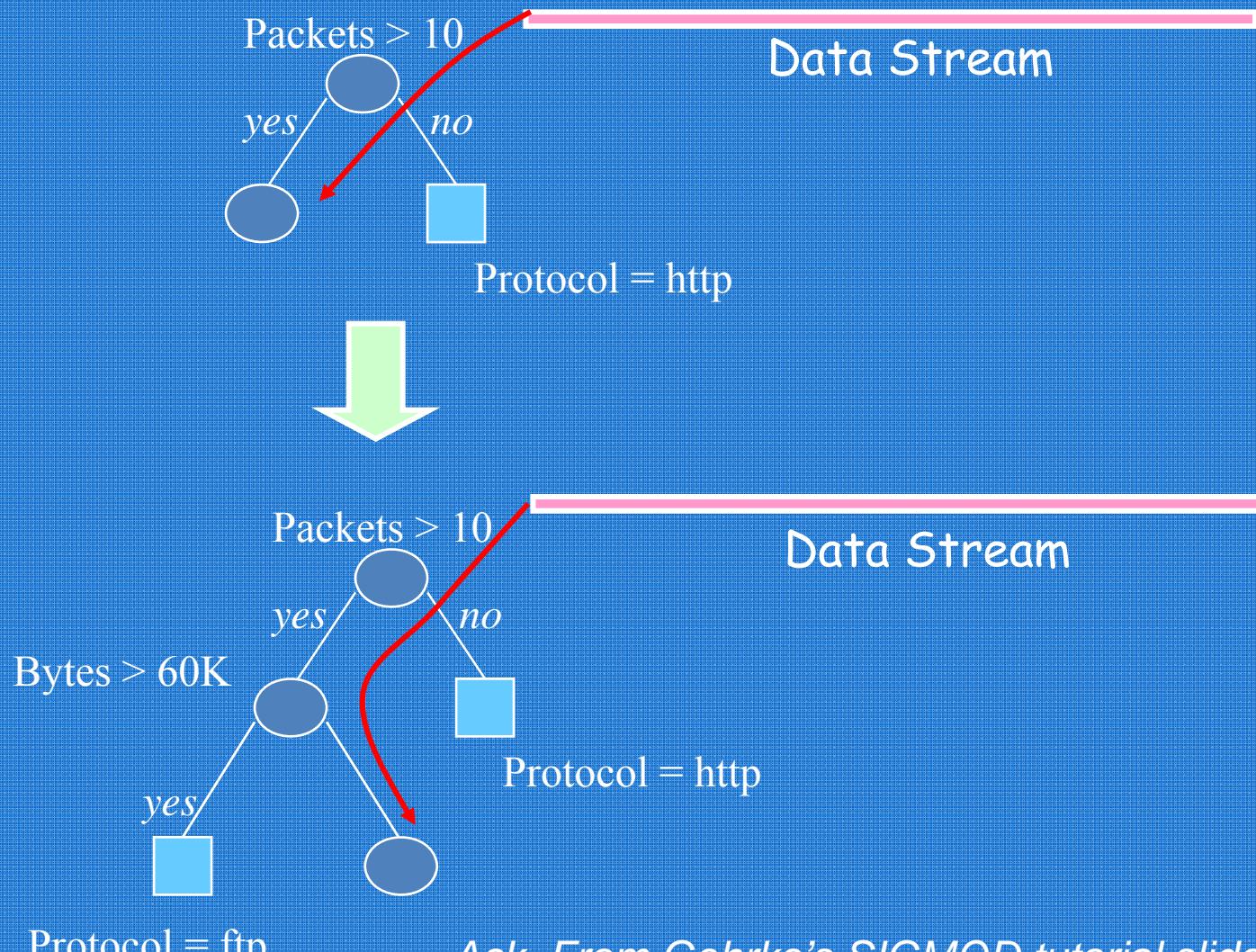
Mean of r is at least $r_{\text{avg}} - \varepsilon$, with probability $1 - d$

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/d)}{2n}}$$

Hoeffding Tree Algorithm

- Hoeffding Tree Input
 - S: sequence of examples
 - X: attributes
 - G(): evaluation function
 - d: desired accuracy
- Hoeffding Tree Algorithm
 - for each example in S
 - retrieve $G(X_a)$ and $G(X_b)$ //two highest $G(X_i)$
 - if ($G(X_a) - G(X_b) > \varepsilon$)
 - split on X_a
 - recurse to next node
 - break

Decision-Tree Induction with Data Streams



Hoeffding Tree: Strengths and Weaknesses

- Strengths
 - Scales better than traditional methods
 - Sublinear with sampling
 - Very small memory utilization
 - Incremental
 - Make class predictions in parallel
 - New examples are added as they come
- Weakness
 - Could spend a lot of time with ties
 - Memory used with tree expansion
 - Number of candidate attributes

Ensemble of Classifiers Algorithm

- H. Wang, W. Fan, P. S. Yu, and J. Han, “Mining Concept-Drifting Data Streams using Ensemble Classifiers”, KDD'03.
- Method (derived from the ensemble idea in classification)
 - train K classifiers from K chunks
 - for each subsequent chunk
 - train a new classifier
 - test other classifiers against the chunk
 - assign weight to each classifier
 - select top K classifiers

Mining Data Streams

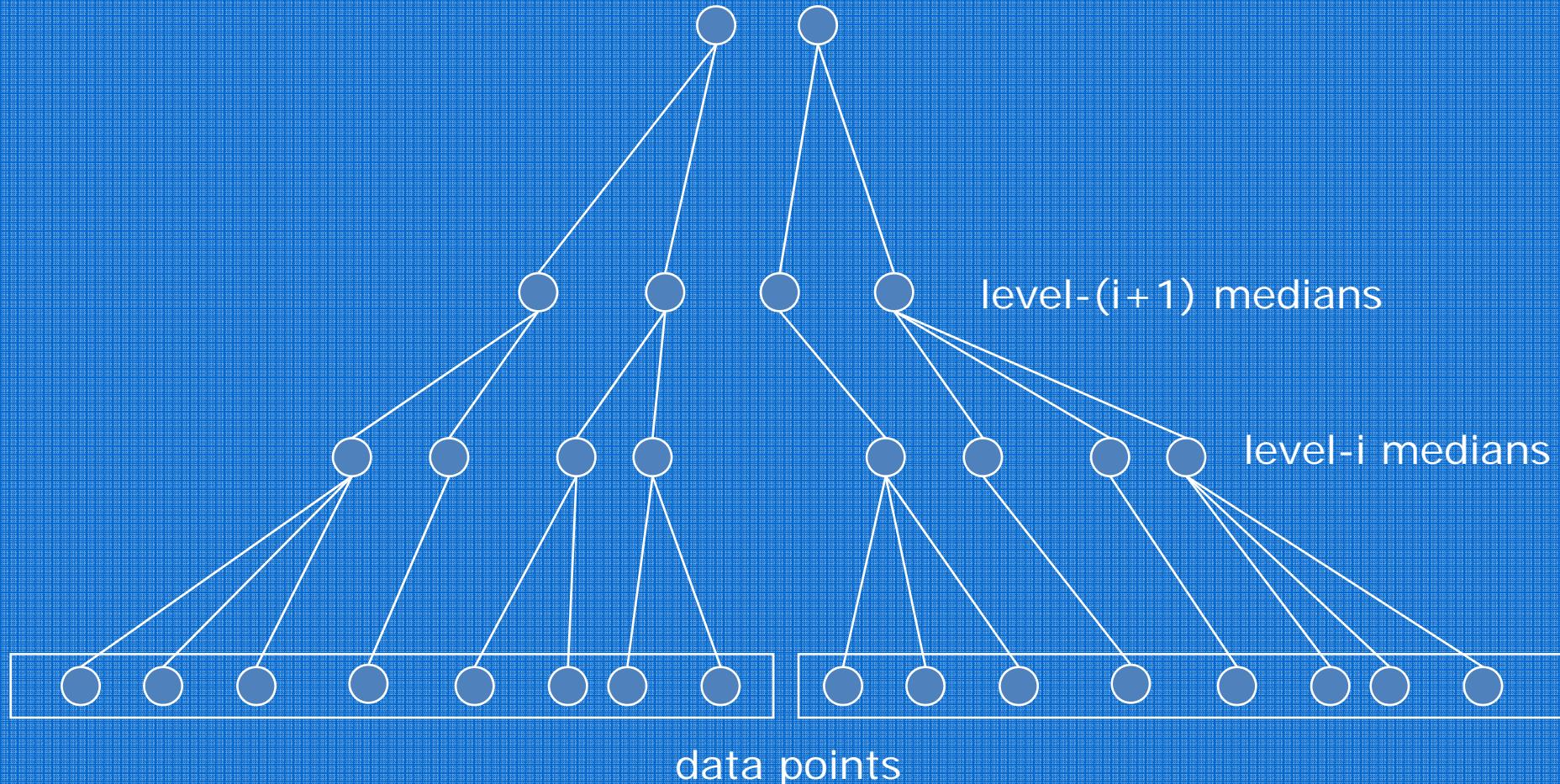
- What is stream data? Why Stream Data Systems?
- Stream data management systems: Issues and solutions
- Stream data cube and multidimensional OLAP analysis
- Stream frequent pattern analysis
- Stream classification
- Stream cluster analysis
- Research issues



Clustering Data Streams [GMMO01]

- Base on the k-median method
 - Data stream points from metric space
 - Find k clusters in the stream s.t. the sum of distances from data points to their closest center is minimized
- Constant factor approximation algorithm
 - In small space, a simple two step algorithm:
 1. For each set of M records, S_i , find $O(k)$ centers in S_1, \dots, S_i
 - Local clustering: Assign each point in S_i to its closest center
 2. Let S' be centers for S_1, \dots, S_i with each center weighted by number of points assigned to it
 - Cluster S' to find k centers

Hierarchical Clustering Tree



Hierarchical Tree and Drawbacks

- Method:
 - maintain at most m level- i medians
 - On seeing m of them, generate $O(k)$ level- $(i+1)$ medians of weight equal to the sum of the weights of the intermediate medians assigned to them
- Drawbacks:
 - Low quality for evolving data streams (register only k centers)
 - Limited functionality in discovering and exploring clusters over different portions of the stream over time

Summary: Stream Data Mining

- Stream data mining: A rich and on-going research field
 - Current research focus in database community:
 - DSMS system architecture, continuous query processing, supporting mechanisms
 - Stream data mining and stream OLAP analysis
 - Powerful tools for finding general and unusual patterns
 - Effectiveness, efficiency and scalability: lots of open problems
 - Our philosophy on stream data analysis and mining
 - A multi-dimensional stream analysis framework
 - Time is a special dimension: Tilted time frame
 - What to compute and what to save?—Critical layers
 - partial materialization and precomputation
 - Mining dynamics of stream data

Mining time-series data

Mining Stream, Time-Series, and Sequence Data

- Mining data streams
- Mining time-series data
- Mining sequence patterns in transactional databases
- Mining sequence patterns in biological data



Time-Series and Sequential Pattern Mining

- Regression and trend analysis—A statistical approach
- Similarity search in time-series analysis
- Sequential Pattern Mining
- Markov Chain
- Hidden Markov Model

Mining Time-Series Data

- Time-series database
 - Consists of sequences of values or events changing with time
 - Data is recorded at regular intervals
 - Characteristic time-series components
 - Trend, cycle, seasonal, irregular
- Applications
 - Financial: stock price, inflation
 - Industry: power consumption
 - Scientific: experiment results
 - Meteorological: precipitation

Categories of Time-Series Movements

- Categories of Time-Series Movements
 - Long-term or trend movements (trend curve): general direction in which a time series is moving over a long interval of time
 - Cyclic movements or cycle variations: long term oscillations about a trend line or curve
 - e.g., business cycles, may or may not be periodic
 - Seasonal movements or seasonal variations
 - i.e, almost identical patterns that a time series appears to follow during corresponding months of successive years.
 - Irregular or random movements
- Time series analysis: decomposition of a time series into these four basic movements
 - Additive Modal: $TS = T + C + S + I$
 - Multiplicative Modal: $TS = T \times C \times S \times I$

Estimation of Trend Curve

- The freehand method
 - Fit the curve by looking at the graph
 - Costly and barely reliable for large-scaled data mining
- The least-square method
 - Find the curve minimizing the sum of the squares of the deviation of points on the curve from the corresponding data points
- The moving-average method

Trend Discovery in Time-Series (1): Estimation of Seasonal Variations

- Seasonal index
 - Set of numbers showing the relative values of a variable during the months of the year
 - E.g., if the sales during October, November, and December are 80%, 120%, and 140% of the average monthly sales for the whole year, respectively, then 80, 120, and 140 are seasonal index numbers for these months
- Deseasonalized data
 - Data adjusted for seasonal variations for better trend and cyclic analysis
 - Divide the original monthly data by the seasonal index numbers for the corresponding months

Trend Discovery in Time-Series (2)

- Estimation of cyclic variations
 - If (approximate) periodicity of cycles occurs, cyclic index can be constructed in much the same manner as seasonal indexes
- Estimation of irregular variations
 - By adjusting the data for trend, seasonal and cyclic variations
- With the systematic analysis of the trend, cyclic, seasonal, and irregular components, it is possible to make long- or short-term predictions with reasonable quality

- Regression and trend analysis—A statistical approach
- Similarity search in time-series analysis
- Sequential Pattern Mining
- Markov Chain
- Hidden Markov Model



Similarity Search in Time-Series Analysis

- Normal database query finds exact match
- Similarity search finds data sequences that differ only slightly from the given query sequence
- Two categories of similarity queries
 - Whole matching: find a sequence that is similar to the query sequence
 - Subsequence matching: find all pairs of similar sequences
- Typical Applications
 - Financial market
 - Market basket data analysis
 - Scientific databases
 - Medical diagnosis

Data Transformation

- Many techniques for signal analysis require the data to be in the frequency domain
- Usually data-independent transformations are used
 - The transformation matrix is determined a priori
 - discrete Fourier transform (DFT)
 - discrete wavelet transform (DWT)
- The distance between two signals in the time domain is the same as their Euclidean distance in the frequency domain

Mining sequence patterns in transactional databases

Mining Stream, Time-Series, and Sequence Data

- Mining data streams
- Mining time-series data
- Mining sequence patterns in transactional databases
- Mining sequence patterns in biological data



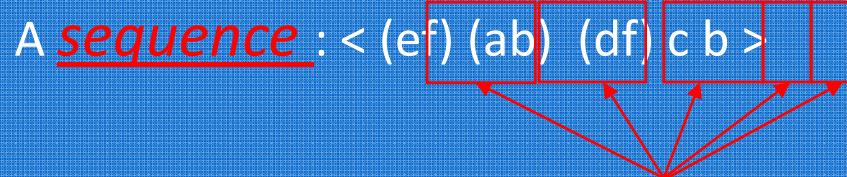
- Transaction databases, time-series databases vs. sequence databases
- Frequent patterns vs. (frequent) sequential patterns
- Applications of sequential pattern mining
 - Customer shopping sequences:
 - First buy computer, then CD-ROM, and then digital camera, within 3 months.
 - Medical treatments, natural disasters (e.g., earthquakes), science & eng. processes, stocks and markets, etc.
 - Telephone calling patterns, Weblog click streams
 - DNA sequences and gene structures

What Is Sequential Pattern Mining?

- Given a set of sequences, find the complete set of *frequent* subsequences

A *sequence database*

SID	sequence
10	<a(<u>abc</u>)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(<u>ab</u>)(df) <u>cb</u> >
40	<eg(af)cbc>



An element may contain a set of items.
Items within an element are unordered
and we list them alphabetically.

<a(bc)dc> is a *subsequence* of
<a(abc)(ac)d(cf)>

Given *support threshold* $min_sup = 2$, <(ab)c> is a *sequential pattern*

Challenges on Sequential Pattern Mining

- A huge number of possible sequential patterns are hidden in databases
- A mining algorithm should
 - find the complete set of patterns, when possible, satisfying the minimum support (frequency) threshold
 - be highly efficient, scalable, involving only a small number of database scans
 - be able to incorporate various kinds of user-specific constraints

Sequential Pattern Mining Algorithms

- Concept introduction and an initial Apriori-like algorithm
 - Agrawal & Srikant. Mining sequential patterns, ICDE'95
- Apriori-based method: **GSP** (Generalized Sequential Patterns: Srikant & Agrawal @ EDBT'96)
- Pattern-growth methods: **FreeSpan** & **PrefixSpan** (Han et al. @ KDD'00; Pei, et al. @ ICDE'01)
- Vertical format-based mining: **SPADE** (Zaki @ Machine Learning'00)
- Constraint-based sequential pattern mining (SPIRIT: Garofalakis, Rastogi, Shim @ VLDB'99; Pei, Han, Wang @ CIKM'02)
- Mining closed sequential patterns: **CloSpan** (Yan, Han & Afshar @ SDM'03)

The Apriori Property of Sequential Patterns

- A basic property: Apriori (Agrawal & Sirkant'94)
 - If a sequence S is not frequent
 - Then none of the super-sequences of S is frequent
 - E.g, $\langle hb \rangle$ is infrequent \rightarrow so do $\langle hab \rangle$ and $\langle (ah)b \rangle$

Seq. ID	Sequence
10	$\langle (bd)cb(ac) \rangle$
20	$\langle (bf)(ce)b(fg) \rangle$
30	$\langle (ah)(bf)abf \rangle$
40	$\langle (be)(ce)d \rangle$
50	$\langle a(bd)bcb(ade) \rangle$

Given *support threshold min_sup*
 $=2$

The SPADE Algorithm

- SPADE (Sequential Pattern Discovery using Equivalent Class) developed by Zaki 2001
- A vertical format sequential pattern mining method
- A sequence database is mapped to a large set of
 - Item: <SID, EID>
- Sequential pattern mining is performed by
 - growing the subsequences (patterns) one item at a time by Apriori candidate generation

The SPADE Algorithm

SID	EID	Items
1	1	a
1	2	abc
1	3	ac
1	4	d
1	5	cf
2	1	ad
2	2	c
2	3	bc
2	4	ae
3	1	ef
3	2	ab
3	3	df
3	4	c
3	5	b
4	1	e
4	2	g
4	3	af
4	4	c
4	5	b
4	6	c

a		b		...	
SID	EID	SID	EID	...	
1	1	1	2		
1	2	2	3		
1	3	3	2		
2	1	3	5		
2	4	4	5		
3	2				
4	3				

ab		ba		...	
SID	EID (a)	EID(b)	SID	EID (b)	EID(a)
1	1	2	1	2	3
2	1	3	2	3	4
3	2	5			
4	3	5			

aba		...	
SID	EID (a)	EID(b)	EID(a)
1	1	2	3
2	1	3	4

Mining sequence patterns in biological data

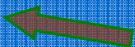
Mining Stream, Time-Series, and Sequence Data

- Mining data streams
- Mining time-series data
- Mining sequence patterns in transactional databases
- Mining sequence patterns in biological data



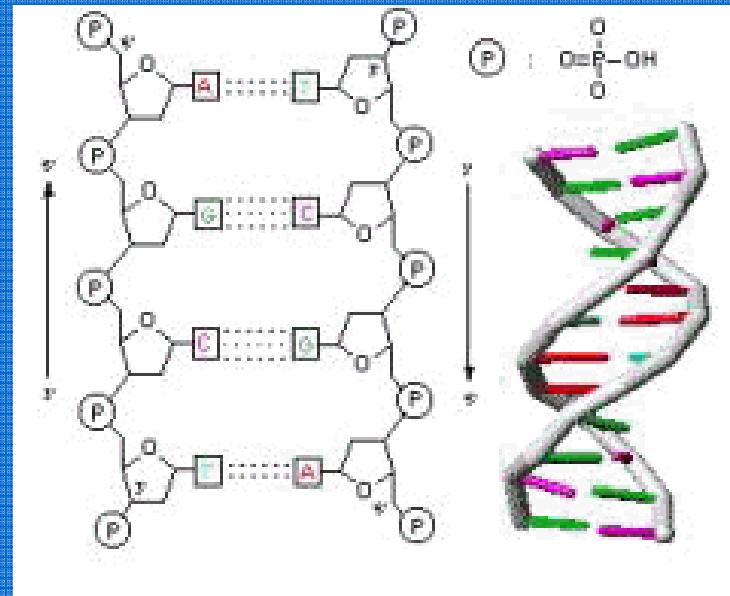
Mining Sequence Patterns in Biological Data

- A brief introduction to biology and bioinformatics
- Alignment of biological sequences
- Hidden Markov model for biological sequence analysis
- Summary



Biology Fundamentals (1): DNA Structure

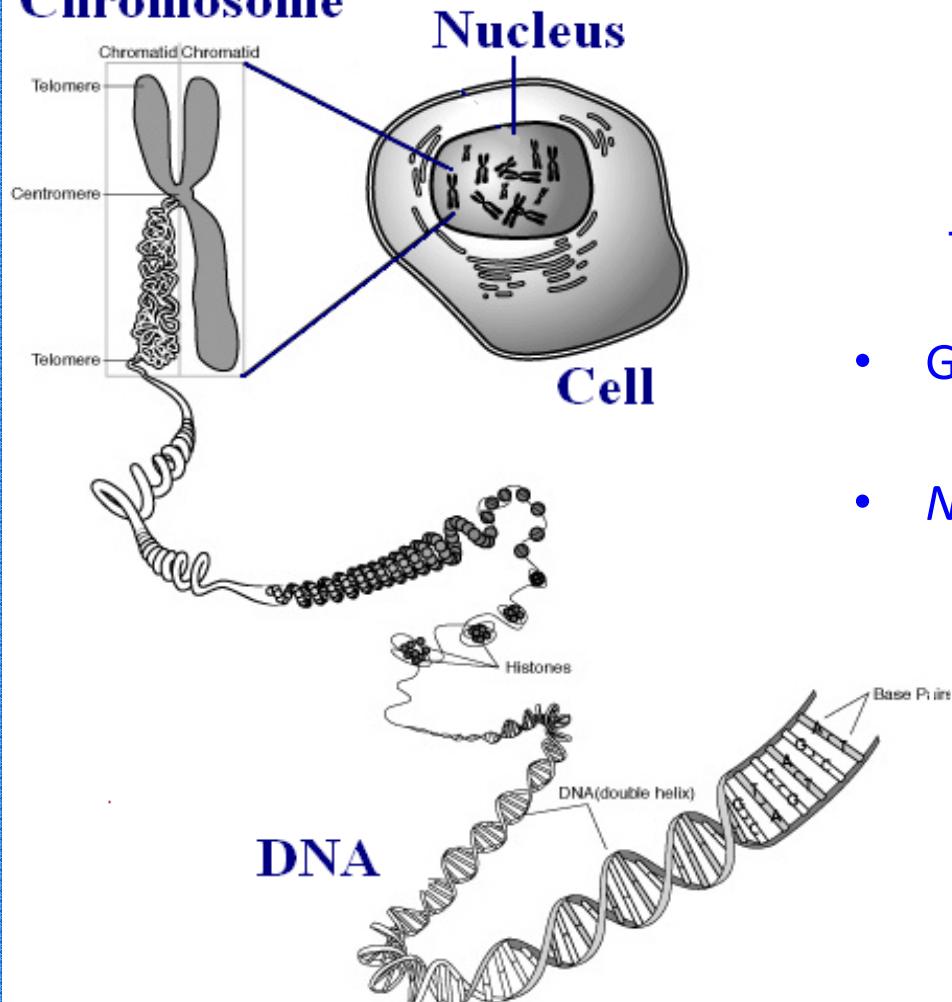
- DNA: helix-shaped molecule whose constituents are two parallel strands of nucleotides
- DNA is usually represented by sequences of these four nucleotides
- This assumes only one strand is considered; the second strand is always derivable from the first by pairing A's with T's and C's with G's and vice-versa



- Nucleotides (bases)
 - Adenine (A)
 - Cytosine (C)
 - Guanine (G)
 - Thymine (T)

Biology Fundamentals (2): Genes

Chromosome



- **Gene:** Contiguous subparts of single strand DNA that are templates for producing *proteins*. Genes can appear in either of the DNA strand.
- **Chromosomes:** compact chains of coiled DNA
- **Genome:** The *set of all genes* in a given organism.
- **Noncoding part:** The function of DNA material between genes is largely unknown. Certain intergenic regions of DNA are known to play a major role in gene regulation (controls the production of proteins and their possible interactions with DNA).

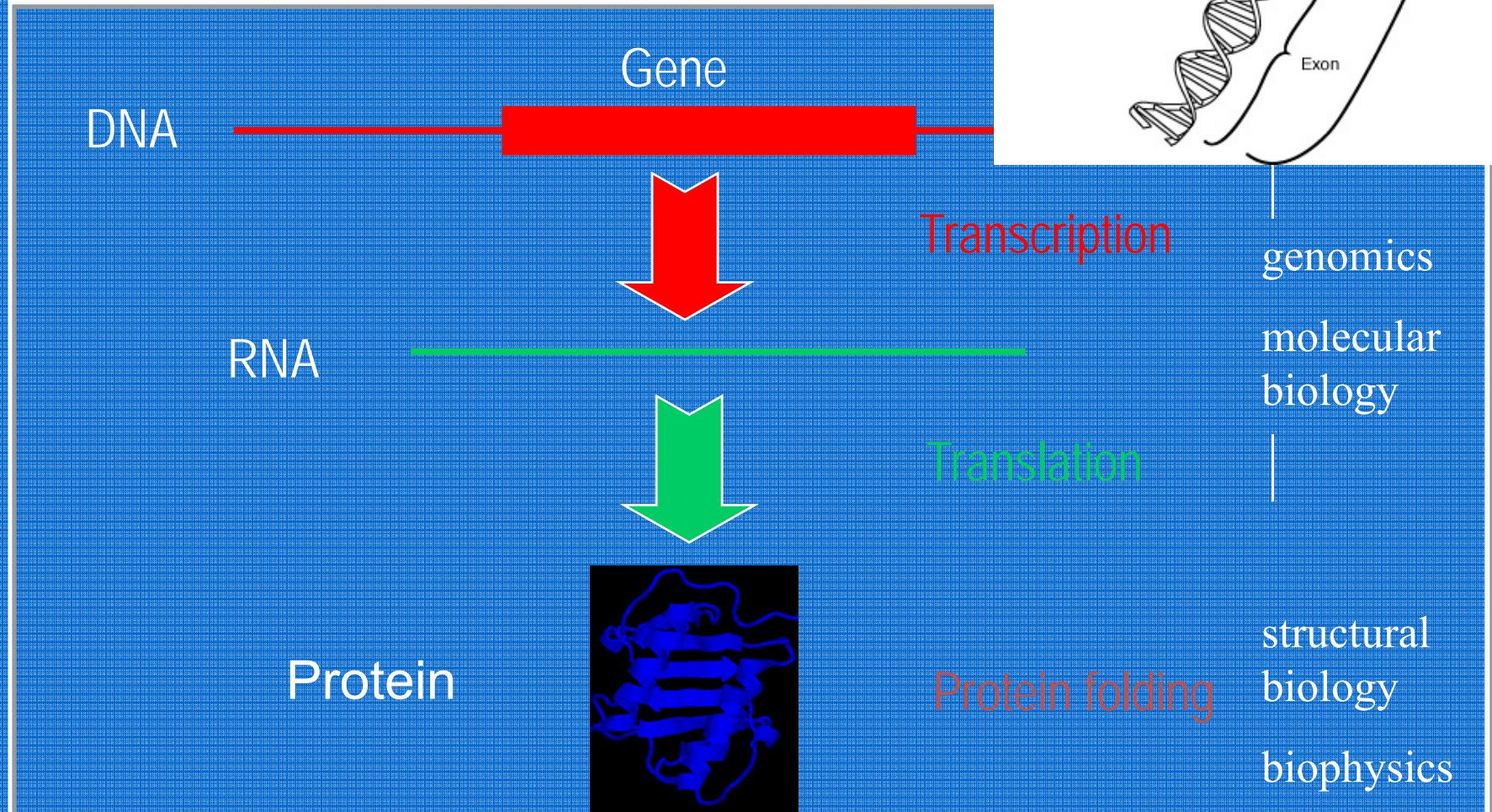
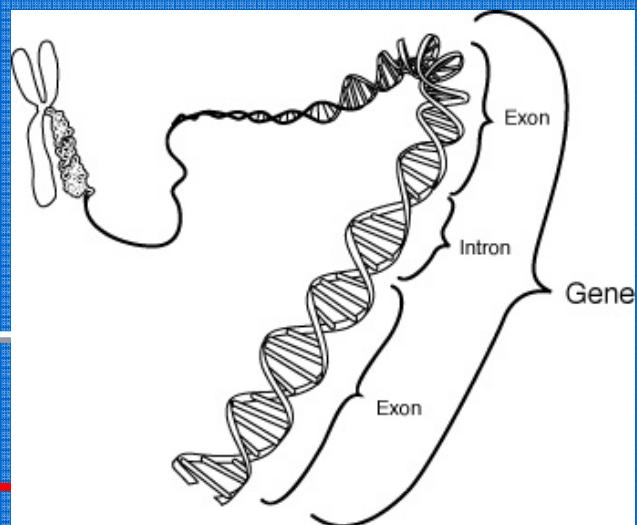
Biology Fundamentals (3): Transcription

- Proteins: Produced from DNA using 3 operations or transformations: *transcription*, *splicing* and *translation*
 - In *eukaryotes* (cells with nucleus): genes are only a minute part of the total DNA
 - In *prokaryotes* (cells without nucleus): the phase of splicing does not occur (no pre-RNA generated)
- DNA is capable of replicating itself (*DNA-polymerase*)
- *Center dogma*: The capability of DNA for replication and undergoing the three (or two) transformations
- Genes are *transcribed* into pre-RNA by a complex ensemble of molecules (*RNA-polymerase*). During transcription T is substituted by the letter U (for *uracil*).
- Pre-RNA can be represented by alternations off sequence segments called *exons* and *introns*. The exons represents the parts of pre-RNA that will be *expressed*, i.e., translated into proteins.

Biology Fundamentals (4): Proteins

- *Splicing* (by spliceosome—an ensemble of proteins): concatenates the exons and excises introns to form mRNA (or simply RNA)
- *Translation* (by ribosomes—an ensemble of RNA and proteins)
 - Repeatedly considers a *triplet* of consecutive nucleotides (called *codon*) in RNA and produces one corresponding amino acid
 - In RNA, there is one special codon called *start codon* and a few others called *stop codons*
- An **Open Reading Frame (ORF)**: a sequence of codons starting with a start codon and ending with an end codon. The ORF is thus a sequence of nucleotides that is used by the ribosome to produce the sequence of amino acid that makes up a protein.
- There are basically **20 amino acids** (A, L, V, S, ...) but in certain rare situations, others can be added to that list.

Biological Information: From Genes to Proteins



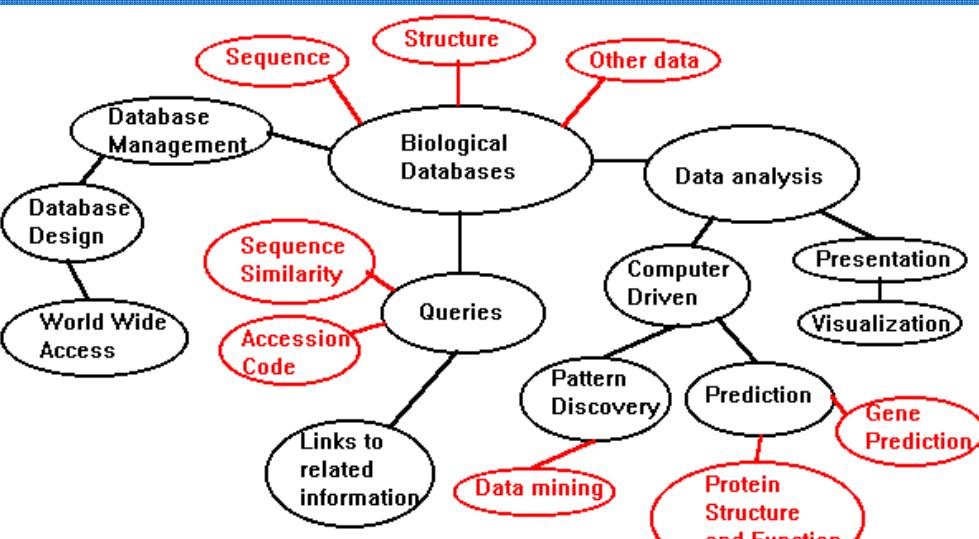
Biology Fundamentals (5): 3D Structure

- Since there are 64 different codons and 20 amino acids, the “table look-up” for translating each codon into an amino acid is redundant: multiple codons can produce the same amino acid
- The table used by nature to perform translation is called the *genetic code*
- Due to the *redundancy* of the genetic code, certain nucleotide changes in DNA may not alter the resulting protein
- Once a protein is produced, it folds into a unique structure in 3D space, with 3 types of components: *α-helices*, *β-sheets* and *coils*.
- The *secondary* structure of a protein is its sequence of amino acids, annotated to distinguish the boundary of each component
- The *tertiary* structure is its 3D representation

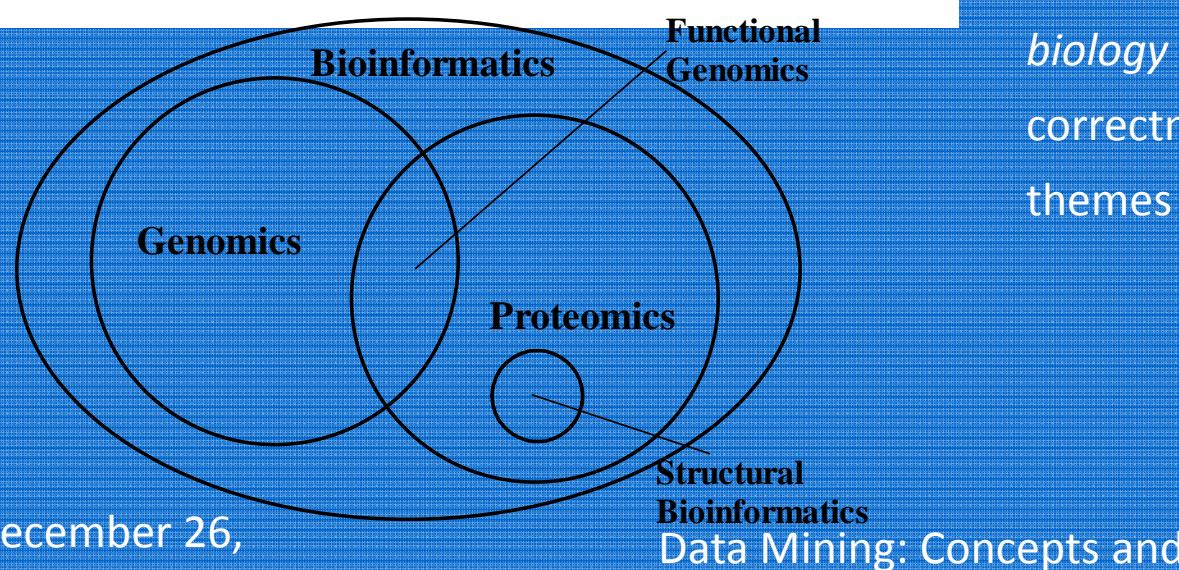
Biological Data Available

- Vast majority of data are *sequence of symbols* (*nucleotides—genomic data, but also good amount on amino acids*).
- Next in volume: *microarray* experiments and also *protein-array* data
- Comparably small: 3D structure of proteins (*PDB*)
- *NCBI* (National Center for Biotechnology Information) server:
 - Total 26B bp: 3B bp human genome, then several bacteria (e.g., E. Coli), higher organisms: yeast, worm, fruit fly, mouse, and plants
 - The largest known genes has ~20million bp and the largest protein consists of ~34k amino acids
 - PDB has a catalogue of only 45k proteins, specified by their 3D structure (i.e, need to infer protein shape from sequence data)

Bioinformatics



- Computational management and analysis of biological information
- Interdisciplinary Field (Molecular Biology, Statistics, Computer Science, Genomics, Genetics, Databases, Chemistry, Radiology ...)
- Bioinformatics vs. *computational biology* (more on algorithm correctness, complexity and other themes central to theoretical CS)



Data Mining & Bioinformatics : Why?

- Many biological processes are not well-understood
- Biological knowledge is **highly complex, imprecise, descriptive, and experimental**
- Biological data is abundant and information-rich
 - Genomics & proteomics data (sequences), microarray and protein-arrays, protein database (PDB), bio-testing data
 - Huge data banks, rich literature, openly accessible
 - Largest and richest scientific data sets in the world
- Mining: gain biological **insight (data/information → knowledge)**
 - Mining for correlations, linkages between disease and gene sequences, protein networks, classification, clustering, outliers, ...
 - Find correlations among linkages in literature and heterogeneous databases

Data Mining & Bioinformatics: How (1)

- Data Integration: Handling heterogeneous, distributed bio-data
 - Build Web-based, interchangeable, integrated, multi-dimensional genome databases
 - Data cleaning and data integration methods becomes crucial
 - Mining correlated information across multiple databases itself becomes a data mining task
 - Typical studies: mining database structures, information extraction from data, reference reconciliation, document classification, clustering and correlation discovery algorithms, ...

Data Mining & Bioinformatics: How (2)

- Master and exploration of existing data mining tools
 - Genomics, proteomics, and functional genomics (functional networks of genes and proteins)
- What are the current bioinformatics tools aiming for?
 - Inferring a protein's shape and function from a given sequence of amino acids
 - Finding all the genes and proteins in a given genome
 - Determining sites in the protein structure where drug molecules can be attached

Data Mining & Bioinformatics – How (3)

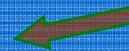
- Research and development of new tools for bioinformatics
 - Similarity search and comparison between classes of genes (e.g., diseased and healthy) by finding and comparing frequent patterns
 - Identify sequential patterns that play roles in various diseases
 - New clustering and classification methods for micro-array data and protein-array data analysis
 - Mining, indexing and similarity search in sequential and structured (e.g., graph and network) data sets
 - Path analysis: linking genes/proteins to different disease development stages
 - Develop pharmaceutical interventions that target the different stages separately
 - High-dimensional analysis and OLAP mining
 - Visualization tools and genetic/proteomic data analysis

Algorithms Used in Bioinformatics

- Comparing sequences: Comparing large numbers of long sequences, allow insertion/deletion/mutations of symbols
- Constructing evolutionary (phylogenetic) trees: Comparing seq. of diff. organisms, & build trees based on their degree of similarity (evolution)
- Detecting patterns in sequences
 - Search for genes in DNA or subcomponents of a seq. of amino acids
- Determining 3D structures from sequences
 - E.g., infer RNA shape from seq. & protein shape from amino acid seq.
- Inferring cell regulation:
 - Cell modeling from experimental (say, microarray) data
- Determining protein function and metabolic pathways: Interpret human annotations for protein function and develop graph db that can be queried
- Assembling DNA fragments (provided by sequencing machines)
- Using script languages: script on the Web to analyze data and applications

Mining Sequence Patterns in Biological Data

- A brief introduction to biology and bioinformatics
- Alignment of biological sequences
- Hidden Markov model for biological sequence analysis
- Summary



Comparing Sequences

- All living organisms are related to evolution
- **Alignment:** Lining up sequences to achieve the maximal level of identity
- Two sequences are *homologous* if they share a common ancestor
- Sequences to be compared: either nucleotides (DNA/RNA) or amino acids (proteins)
 - Nucleotides: identical
 - Amino acids: identical, or if one can be derived from the other by substitutions that are likely to occur in nature
- **Local vs. global alignments:** Local—only portions of the sequences are aligned. Global—align over the entire length of the sequences
 - Use gap “–” to indicate preferable not to align two symbols
- **Percent identity:** ratio between the number of columns containing identical symbols vs. the number of symbols in the longest sequence
- **Score of alignment:** summing up the matches and counting gaps as negative

Sequence Alignment: Problem Definition

- Goal:
 - Given two or more input sequences
 - Identify similar sequences with long conserved subsequences
- Method:
 - Use substitution matrices (probabilities of substitutions of nucleotides or amino-acids and probabilities of insertions and deletions)
 - *Optimal* alignment problem: NP-hard
 - Heuristic method to find *good* alignments

Pair-Wise Sequence Alignment

- Example

HEAGAWGHEE
PAWHEAE

HEAGAWGHE-E
| | | | |
P-A--W-HEAE

HEAGAWGHE-E
| | | | |
--P-AW-HEAE

- Which one is better? → **Scoring alignments**
- To compare two sequence alignments, calculate a score
 - PAM (Percent Accepted Mutation) or BLOSUM (Blocks Substitution Matrix) (*substitution*) matrices: Calculate matches and mismatches, considering amino acid substitution
 - Gap penalty: Initiating a gap
 - Gap extension penalty: Extending a gap

Pair-wise Sequence Alignment: Scoring Matrix

	A	E	G	H	W
A	5	-1	0	-2	-3
E	-1	6	-3	0	-3
H	-2	0	-2	10	-3
P	-1	-1	-2	-2	-4
W	-3	-3	-3	-3	15

Exercise: Calculate for

- Gap penalty: -8
- Gap extension: -8

HEAGAWGHE – E

| | | | |

--P-AW-HEAE

$$\begin{aligned}
(-8) &+ (-8) + (-1) + 5 + 15 + (-8) \\
&+ 10 + 6 + (-8) + 6 = 9
\end{aligned}$$

HEAGAWGHE – E

| | | | |

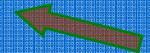
P-A--W-HEAE

Heuristic Alignment Algorithms

- Motivation: Complexity of alignment algorithms: $O(nm)$
 - Current protein DB: 100 million base pairs
 - Matching each sequence with a 1,000 base pair query takes about 3 hours!
- Heuristic algorithms aim at speeding up at the price of possibly missing the best scoring alignment
- Two well known programs
 - BLAST: Basic Local Alignment Search Tool
 - FASTA: Fast Alignment Tool
 - Both find high scoring local alignments between a query sequence and a target database
 - Basic idea: first locate high-scoring short stretches and then extend them

Mining Sequence Patterns in Biological Data

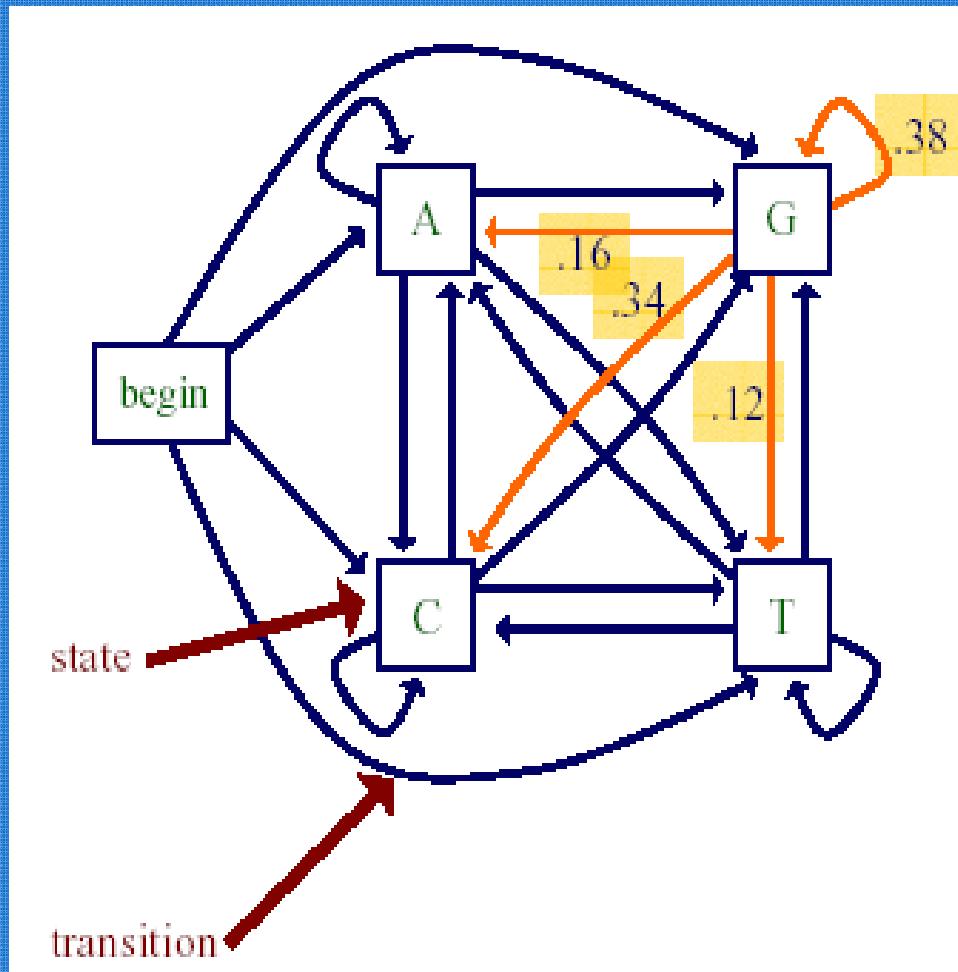
- A brief introduction to biology and bioinformatics
- Alignment of biological sequences
- Hidden Markov model for biological sequence analysis
- Summary



Motivation for Markov Models in Computational Biology

- There are many cases in which we would like to represent the statistical regularities of some class of sequences
 - genes
 - various regulatory sites in DNA (e.g., where RNA polymerase and transcription factors bind)
 - proteins in a given family
- Markov models are well suited to this type of task

A Markov Chain Model



Transition probabilities

- $\Pr(x_i=a | x_{i-1}=g)=0.16$
- $\Pr(x_i=c | x_{i-1}=g)=0.34$
- $\Pr(x_i=g | x_{i-1}=g)=0.38$
- $\Pr(x_i=t | x_{i-1}=g)=0.12$

$$\sum \Pr(x_i | x_{i-1} = g) = 1$$

Definition of Markov Chain Model

- A Markov chain model is defined by
 - a set of states
 - some states emit symbols
 - other states (e.g., the begin state) are silent
 - a set of transitions with associated probabilities
 - the transitions emanating from a given state define a distribution over the possible next states

Markov Chain Models: Properties

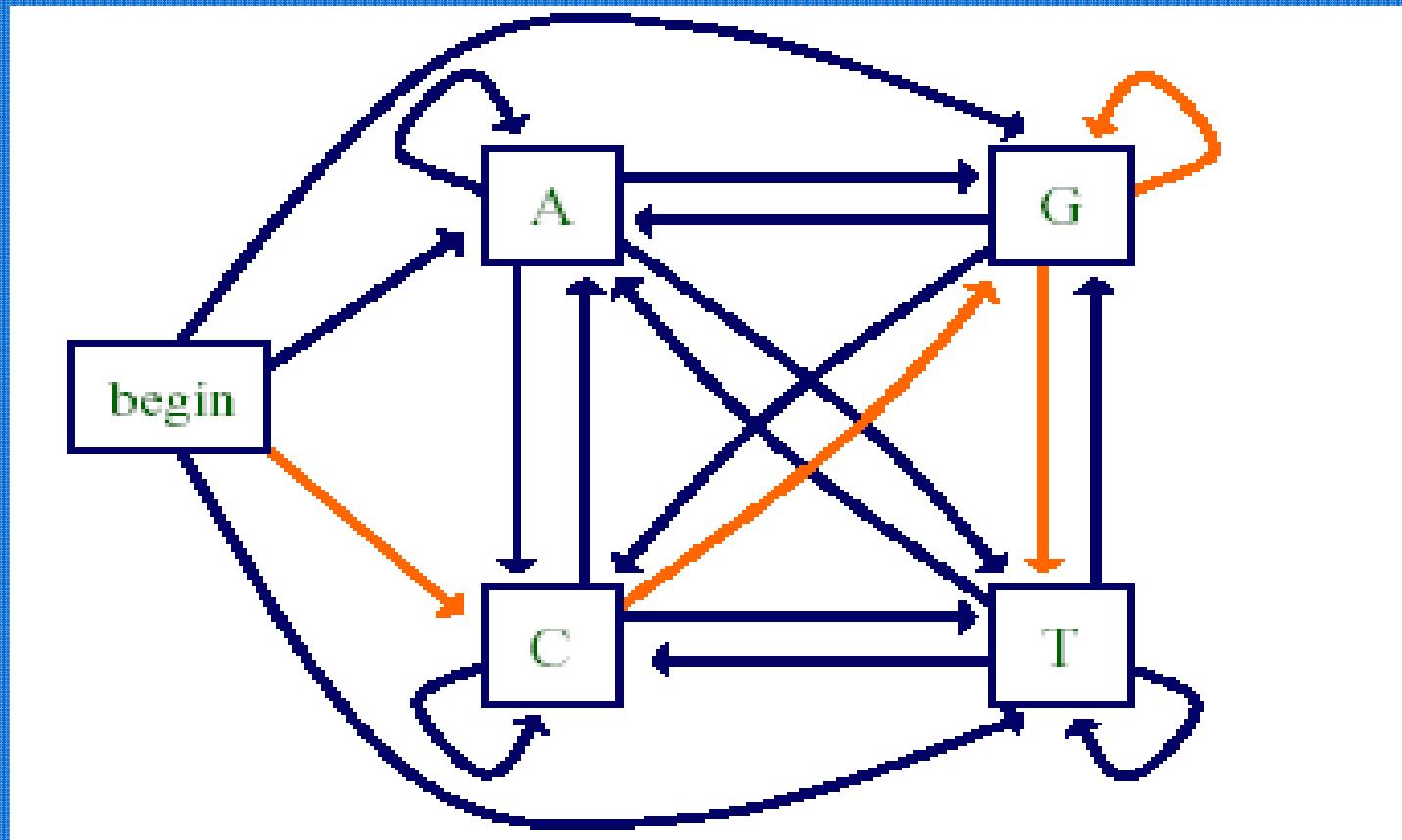
- Given some sequence x of length L , we can ask how probable the sequence is given our model
- For any probabilistic model of sequences, we can write this probability as

$$\begin{aligned}\Pr(x) &= \Pr(x_L, x_{L-1}, \dots, x_1) \\ &= \Pr(x_L / x_{L-1}, \dots, x_1) \Pr(x_{L-1} | x_{L-2}, \dots, x_1) \dots \Pr(x_1)\end{aligned}$$

- key property of a (1st order) Markov chain: the probability of each x_i depends only on the value of x_{i-1}

$$\begin{aligned}\Pr(x) &= \Pr(x_L / x_{L-1}) \Pr(x_{L-1} | x_{L-2}) \dots \Pr(x_2 | x_1) \Pr(x_1) \\ &= \Pr(x_1) \prod_{i=2}^L \Pr(x_i | x_{i-1})\end{aligned}$$

The Probability of a Sequence for a Markov Chain Model

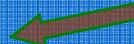


$$\Pr(\text{cggt}) = \Pr(c)\Pr(g|c)\Pr(g|g)\Pr(t|g)$$

- Learning
 - correct path known for each training sequence -> simple maximum likelihood or Bayesian estimation
 - correct path not known -> Forward-Backward algorithm + ML or Bayesian estimation
- Classification
 - simple Markov model -> calculate probability of sequence along single path for each model
 - hidden Markov model -> Forward algorithm to calculate probability of sequence along all paths for each model
- Segmentation
 - hidden Markov model -> Viterbi algorithm to find most probable path for sequence

Mining Sequence Patterns in Biological Data

- A brief introduction to biology and bioinformatics
- Alignment of biological sequences
- Hidden Markov model for biological sequence analysis
- Summary



Summary: Mining Biological Data

- Biological sequence analysis compares, aligns, indexes, and analyzes biological sequences (sequence of nucleotides or amino acids)
- Biosequence analysis can be partitioned into two essential tasks:
 - pair-wise sequence alignment and multiple sequence alignment
- Dynamic programming approach (notably, BLAST) has been popularly used for sequence alignments
- Markov chains and hidden Markov models are probabilistic models in which the probability of a state depends only on that of the previous state
 - Given a sequence of symbols, x , the **forward** algorithm finds the probability of obtaining x in the model
 - The **Viterbi** algorithm finds the most probable path (corresponding to x) through the model
 - The **Baum-Welch** learns or adjusts the model parameters (transition and emission probabilities) to best explain a set of training sequences.

Graph mining

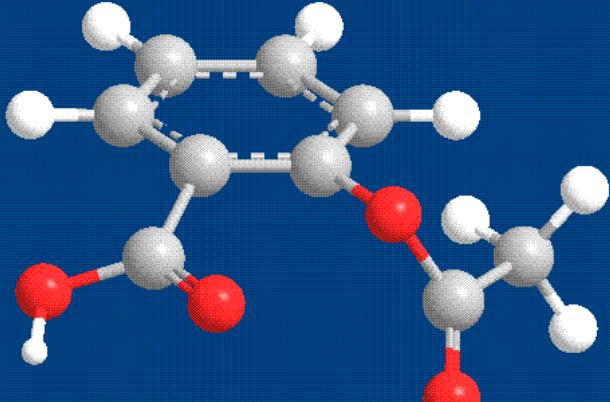


- Methods for Mining Frequent Subgraphs
- Mining Variant and Constrained Substructure Patterns
- Applications:
 - Graph Indexing
 - Similarity Search
 - Classification and Clustering
- Summary

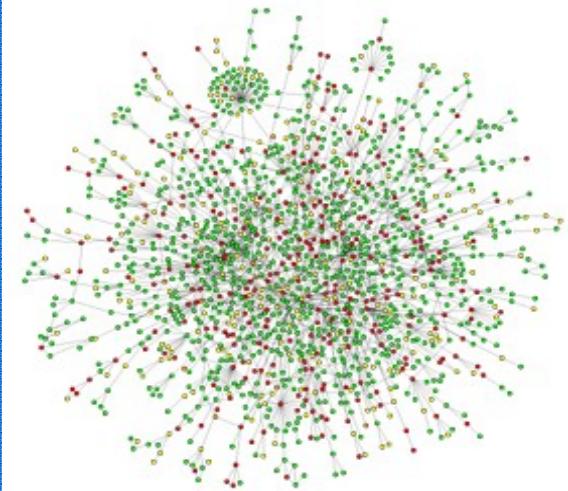
Why Graph Mining?

- Graphs are ubiquitous
 - Chemical compounds (Cheminformatics)
 - Protein structures, biological pathways/networks (Bioinformatics)
 - Program control flow, traffic flow, and workflow analysis
 - XML databases, Web, and social network analysis
- Graph is a general model
 - Trees, lattices, sequences, and items are degenerated graphs
- Diversity of graphs
 - Directed vs. undirected, labeled vs. unlabeled (edges & vertices), weighted, with angles & geometry (topological vs. 2-D/3-D)
- Complexity of algorithms: many problems are of high complexity

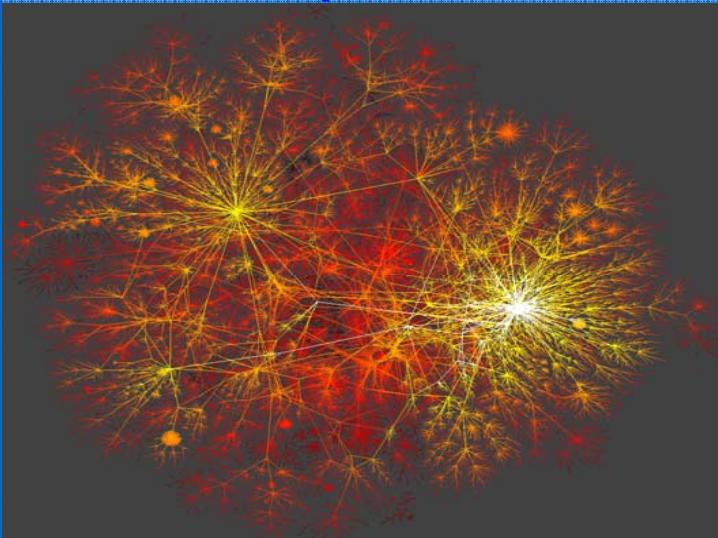
Graph, Graph, Everywhere



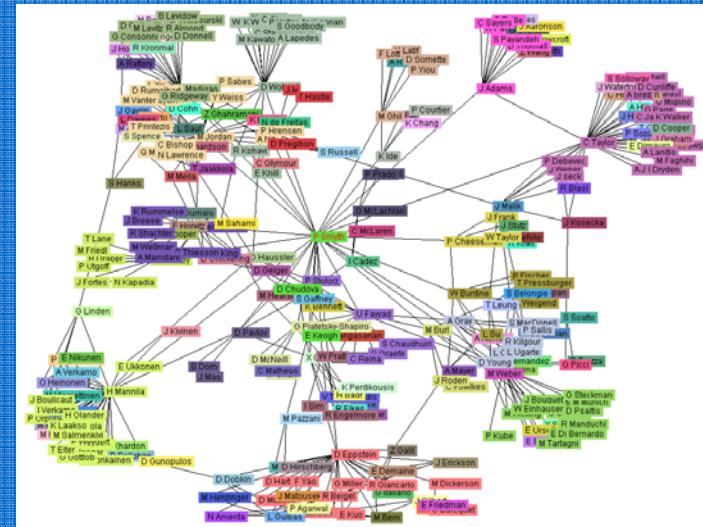
Aspirin



Yeast protein interaction network



Internet



Co-author network
317

from H. Jeong et al Nature 411, 41 (2001)

Graph Pattern Mining

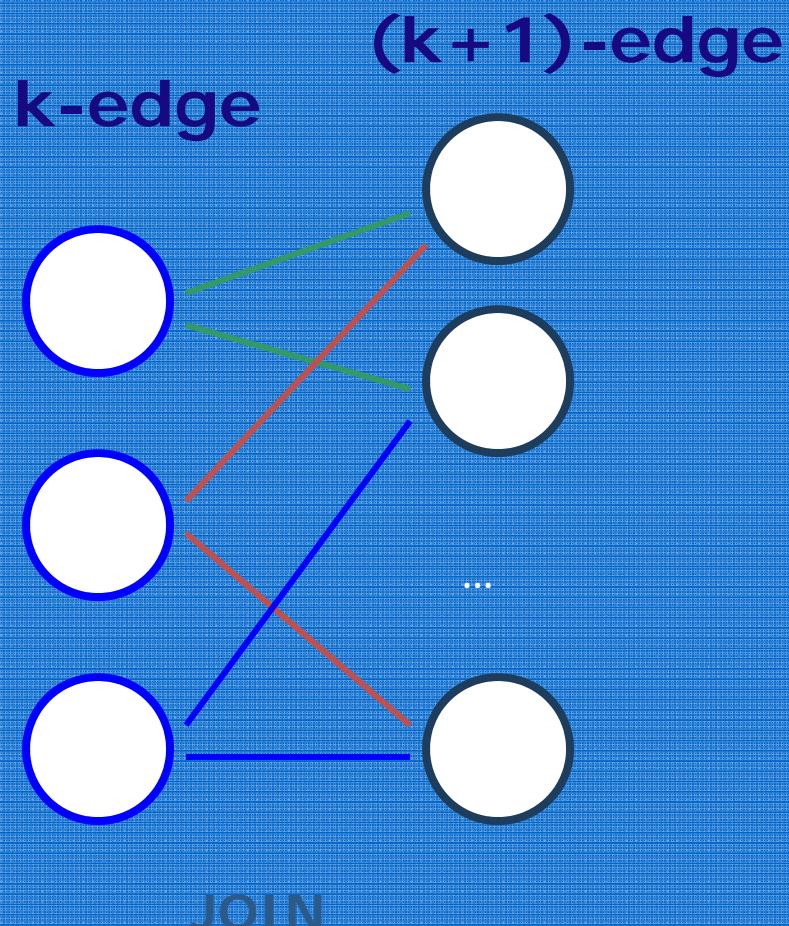
- *Frequent* subgraphs
 - A (sub)graph is *frequent* if its *support* (occurrence frequency) in a given dataset is no less than a *minimum support* threshold
- Applications of graph pattern mining
 - Mining biochemical structures
 - Program control flow analysis
 - Mining XML structures or Web communities
 - Building blocks for graph classification, clustering, compression, comparison, and correlation analysis

- Incomplete beam search – Greedy (Subdue)
- Inductive logic programming (WARMR)
- Graph theory-based approaches
 - Apriori-based approach
 - Pattern-growth approach

- Start with single vertices
- Expand best substructures with a new edge
- Limit the number of best substructures
 - Substructures are evaluated based on their ability to compress input graphs
 - Using minimum description length (DL)
 - Best substructure S in graph G minimizes: $DL(S) + DL(G \setminus S)$
- Terminate until no new substructure is discovered

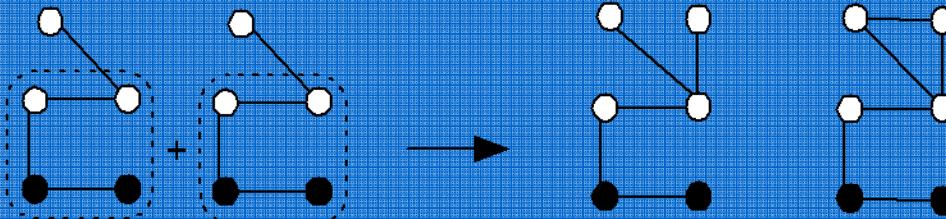
- Search order
 - breadth vs. depth
- Generation of candidate subgraphs
 - apriori vs. pattern growth
- Elimination of duplicate subgraphs
 - passive vs. active
- Support calculation
 - embedding store or not
- Discover order of patterns
 - path → tree → graph

Apriori-Based Approach

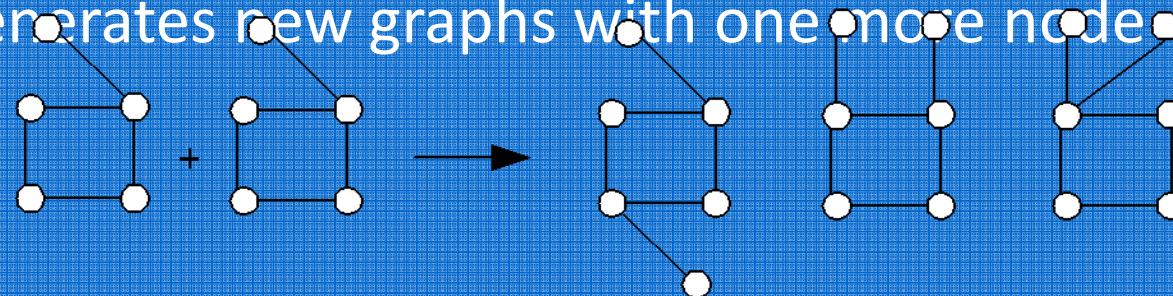


Apriori-Based, Breadth-First Search

- Methodology: breadth-search, joining two graphs



- AGM (Inokuchi, et al. PKDD'00)
 - generates new graphs with one more node
- FSG (Kuramochi and Karypis ICDM'01)
 - generates new graphs with one more edge

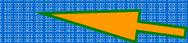


Graph Pattern Explosion Problem

- If a graph is frequent, all of its subgraphs are frequent – **the Apriori property**
- An n -edge frequent graph may have 2^n subgraphs
- Among 422 chemical compounds which are confirmed to be active in an AIDS antiviral screen dataset, there are **1,000,000** frequent graph patterns if the minimum support is 5%

Graph Mining

- Methods for Mining Frequent Subgraphs
- Mining Variant and Constrained Substructure Patterns
- Applications:
 - Graph Indexing
 - Similarity Search
 - Classification and Clustering
- Summary

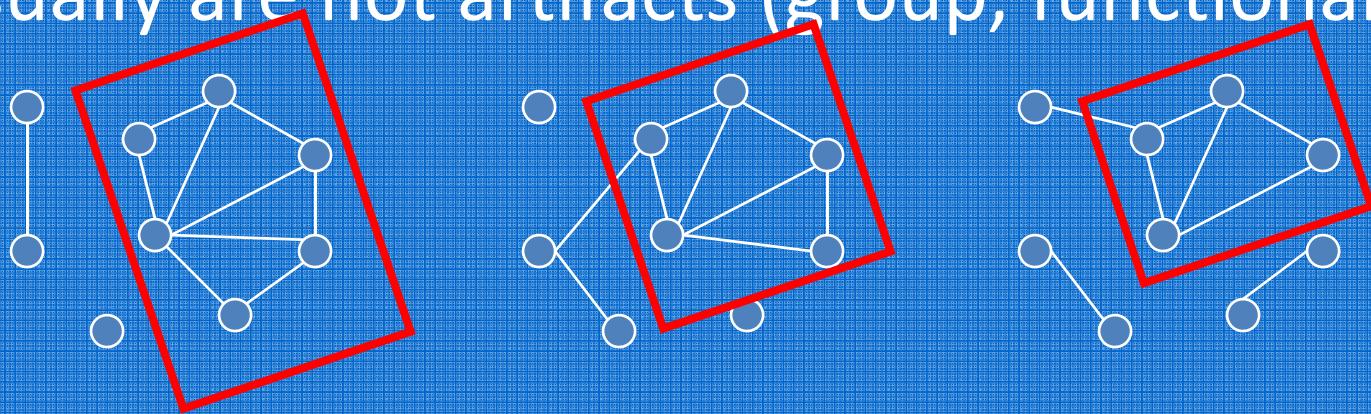


Constrained Patterns

- Density
- Diameter
- Connectivity
- Degree
- Min, Max, Avg

Constraint-Based Graph Pattern Mining

- Highly connected subgraphs in a large graph usually are not artifacts (group, functionality)



- Recurrent patterns discovered in multiple graphs are more robust than the patterns mined from a single graph

- Methods for Mining Frequent Subgraphs
- Mining Variant and Constrained Substructure Patterns
- Applications:
 - Classification and Clustering
 - Graph Indexing
 - Similarity Search
- Summary



Graph Clustering

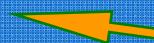
- Graph similarity measure
 - Feature-based similarity measure
 - Each graph is represented as a feature vector
 - The similarity is defined by the distance of their corresponding vectors
 - Frequent subgraphs can be used as features
 - Structure-based similarity measure
 - Maximal common subgraph
 - Graph edit distance: insertion, deletion, and relabel
 - Graph alignment distance

Graph Classification

- Local structure based approach
 - Local structures in a graph, e.g., neighbors surrounding a vertex, paths with fixed length
- Graph pattern-based approach
 - Subgraph patterns from domain knowledge
 - Subgraph patterns from data mining
- Kernel-based approach
 - Random walk (Gärtner '02, Kashima et al. '02, ICML'03, Mahé et al. ICML'04)
 - Optimal local assignment (Fröhlich et al. ICML'05)
- Boosting (Kudo et al. NIPS'04)

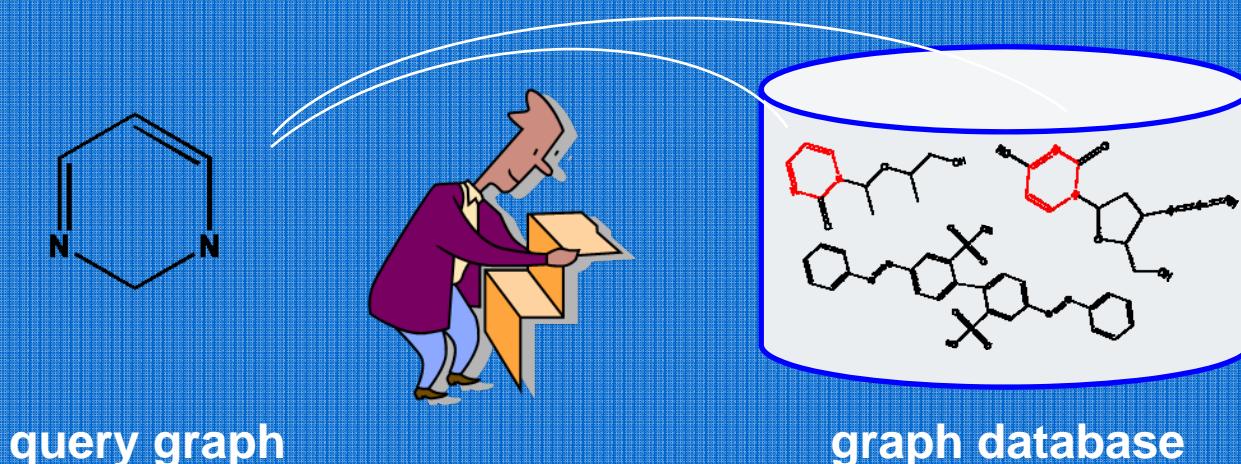
- Subgraph patterns from domain knowledge
 - Molecular descriptors
- Subgraph patterns from data mining
- General idea
 - Each graph is represented as a feature vector $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$, where x_i is the frequency of the i -th pattern in that graph
 - Each vector is associated with a class label
 - Classify these vectors in a vector space

- Methods for Mining Frequent Subgraphs
- Mining Variant and Constrained Substructure Patterns
- Applications:
 - Classification and Clustering
 - Graph Indexing
 - Similarity Search
- Summary



Graph Search

- Querying graph databases:
 - Given a graph database and a query graph, find all the graphs containing this query graph



Scalability Issue

- Sequential scan
 - Disk I/Os
 - Subgraph isomorphism testing
- An indexing mechanism is needed
 - DayLight: Daylight.com (commercial)
 - GraphGrep: Dennis Shasha, et al. PODS'02
 - Grace: Srinath Srinivasa, et al. ICDE'03

Summary: Graph Mining

- Graph mining has wide applications
- Frequent and closed subgraph mining methods
 - gSpan and CloseGraph: pattern-growth depth-first search approach
- Graph indexing techniques
 - Frequent and discriminative subgraphs are high-quality indexing features
- Similarity search in graph databases
 - Indexing and feature-based matching
- Further development and application exploration

Social Network Analysis

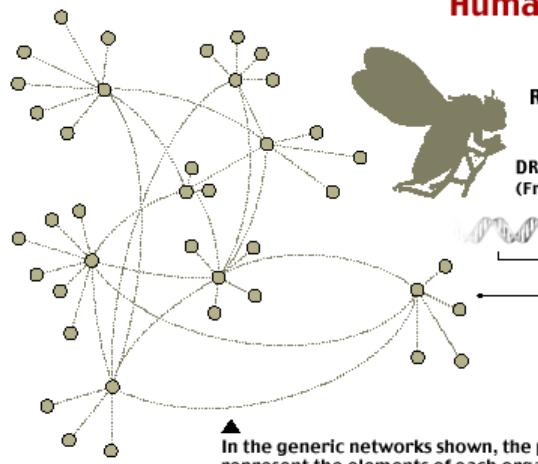
Social Network Analysis

- Social Network Introduction
- Statistics and Probability Theory
- Models of Social Network Generation
- Networks in Biological System
- Mining on Social Network
- Summary



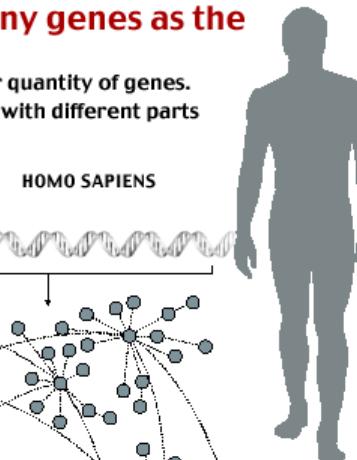
Humans have only about three times as many genes as the fly,

so human complexity seems unlikely to come from a sheer quantity of genes.
Rather, some scientists suggest, each human has a network with different parts
like genes, proteins and groups



In the generic networks shown, the points represent the elements of each organism's genetic network, and the dotted lines show the interactions between them. Humans have many more ele-

DROSOPHILA MELANOGASTER
(Fruit fly)



HOMO SAPIENS

Steve Duenes/The New York Times

In this example the fly has 40 genes, and the human

elements
interactions

NETWORK

“Natural” Networks and Universality

- Consider many kinds of networks:
 - social, technological, business, economic, content,...
- These networks tend to share certain *informal* properties:
 - large scale; continual growth
 - distributed, organic growth: vertices “decide” who to link to
 - interaction restricted to links
 - mixture of local and long-distance connections
 - abstract notions of distance: geographical, content, social,...
- Do natural networks share more *quantitative* universals?
- What would these “universals” be?
- How can we make them precise and measure them?
- How can we explain their universality?
- This is the domain of *social network theory*
- Sometimes also referred to as *link analysis*

Some Interesting Quantities

- *Connected components:*
 - how many, and how large?
- *Network diameter:*
 - maximum (worst-case) or average?
 - exclude infinite distances? (disconnected components)
 - the small-world phenomenon
- *Clustering:*
 - to what extent that links tend to cluster “locally”?
 - what is the balance between local and long-distance connections?
 - what roles do the two types of links play?
- *Degree distribution:*
 - what is the typical degree in the network?
 - what is the overall distribution?

A “Canonical” Natural Network has...

- *Few* connected components:
 - often only 1 or a small number, indep. of network size
- *Small* diameter:
 - often a constant independent of network size (like 6)
 - or perhaps growing only logarithmically with network size or even shrink?
 - typically exclude infinite distances
- A *high* degree of clustering:
 - considerably more so than for a random network
 - in tension with small diameter
- A *heavy-tailed* degree distribution:
 - a small but reliable number of high-degree vertices
 - often of *power law* form

Probabilistic Models of Networks

- All of the network generation models we will study are *probabilistic* or *statistical* in nature
- They can generate networks of any size
- They often have various *parameters* that can be set:
 - size of network generated
 - average degree of a vertex
 - fraction of long-distance connections
- The models generate a *distribution* over networks
- Statements are always *statistical* in nature:
 - *with high probability*, diameter is small
 - *on average*, degree distribution has heavy tail
- Thus, we're going to need some basic statistics and probability theory

Social Network Analysis

- Social Network Introduction
- Statistics and Probability Theory
- Models of Social Network Generation
- Networks in Biological System
- Mining on Social Network
- Summary

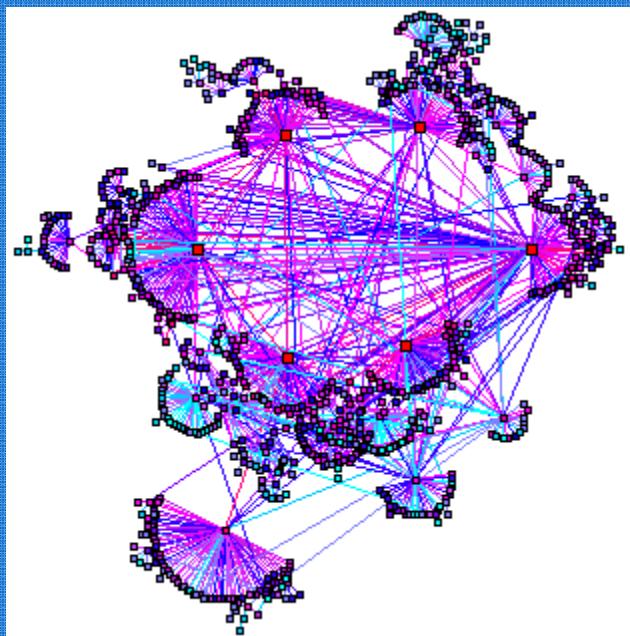


World Wide Web

Nodes: WWW documents

Links: URL links

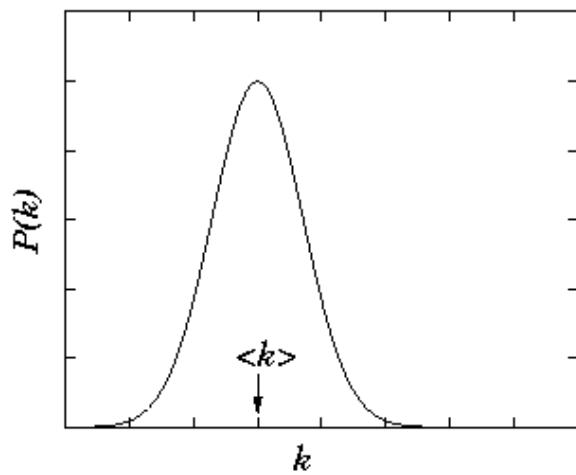
800 million documents
(S. Lawrence, 1999)



ROBOT: collects all
URL's found in a
document and follows
them recursively

World Wide Web

Expected Result



$$\langle K \rangle \sim 6$$

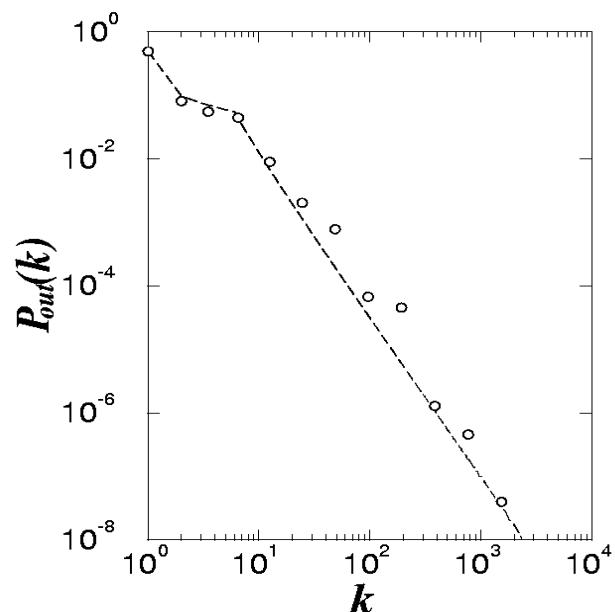
$$P(k=500) \sim 10^{-99}$$

$$N_{\text{WWW}} \sim 10^9$$

$$\Rightarrow N(k=500) \sim 10^{-90}$$

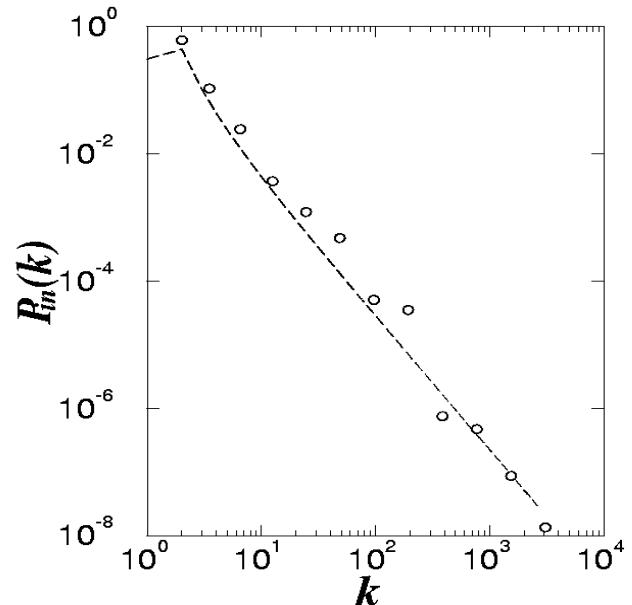
December 26,
2012

Real Result



$$P_{\text{out}}(k) \sim k^{-\gamma_{\text{out}}}$$

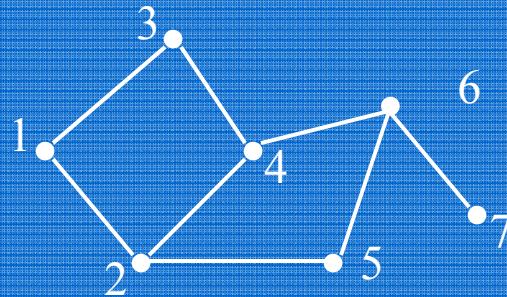
$$P(k=500) \sim 10^{-6}$$



$$P_{\text{in}}(k) \sim k^{-\gamma_{\text{in}}}$$

$$\begin{aligned} N_{\text{WWW}} &\sim 10^9 \\ \Rightarrow N(k=500) &\sim 10^3 \end{aligned}$$

World Wide Web



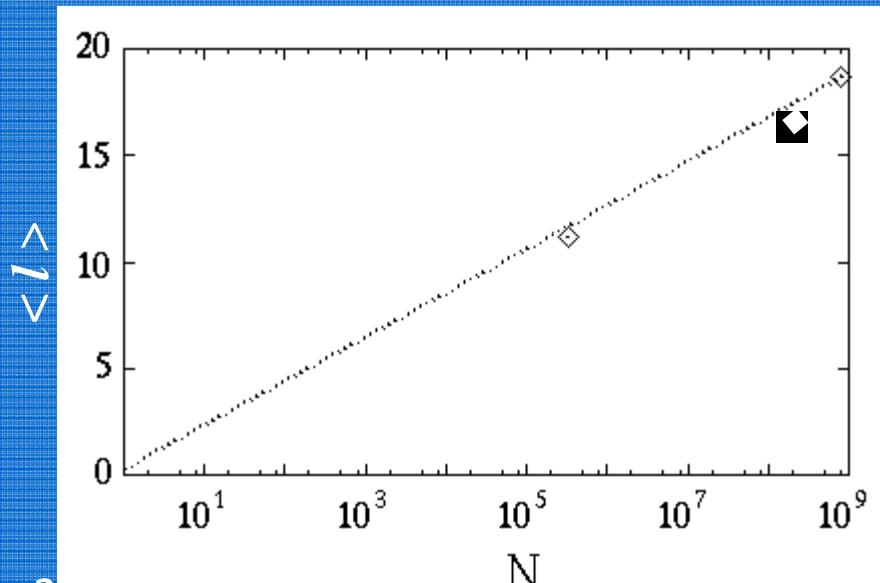
$$l_{15}=2 [1 \rightarrow 2 \rightarrow 5]$$

$$l_{17}=4 [1 \rightarrow 3 \rightarrow 4 \rightarrow 6 \rightarrow 7]$$

$$\dots \langle l \rangle = ??$$

- **Finite size scaling:** create a network with N nodes with $P_{in}(k)$ and $P_{out}(k)$

$$\langle l \rangle = 0.35 + 2.06 \log(N)$$



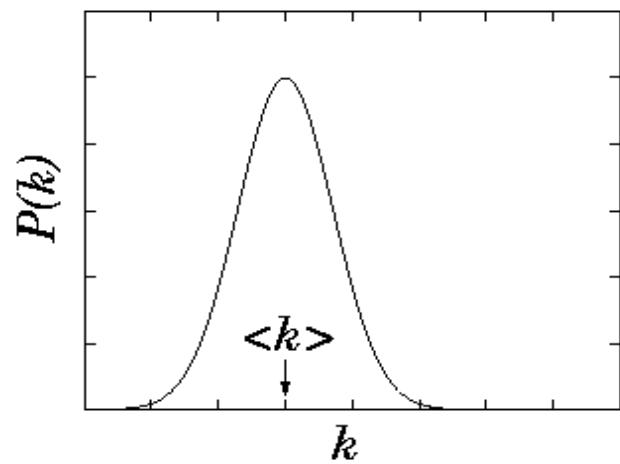
19 degrees of separation

R. Albert et al Nature (99)

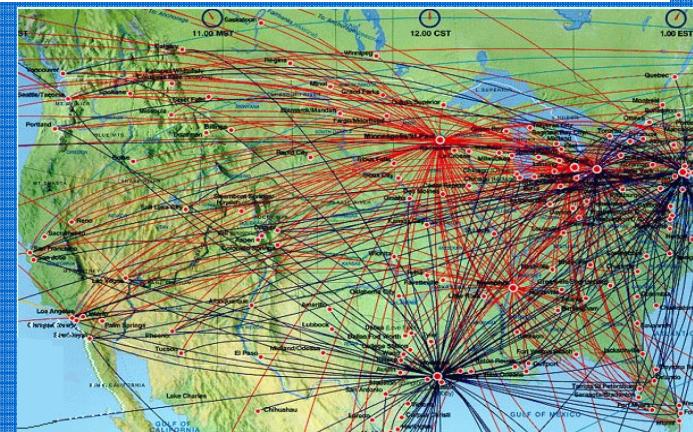
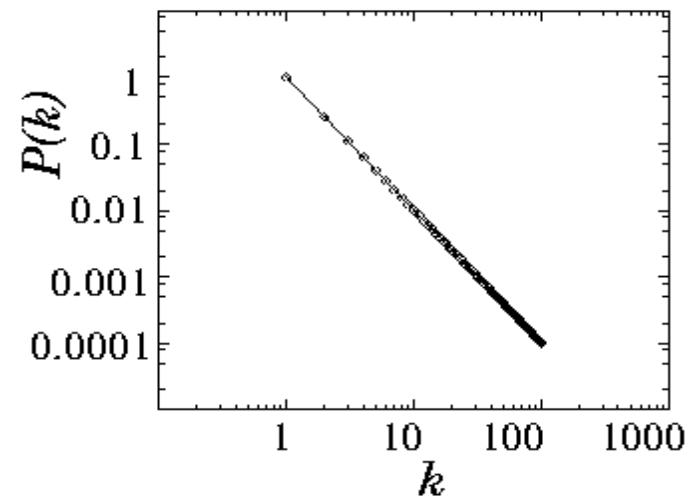
based on 800 million webpages
[S. Lawrence et al Nature (99)]

What does that mean?

Poisson distribution



Power law distribution



Exponential Network

Scale-free Network

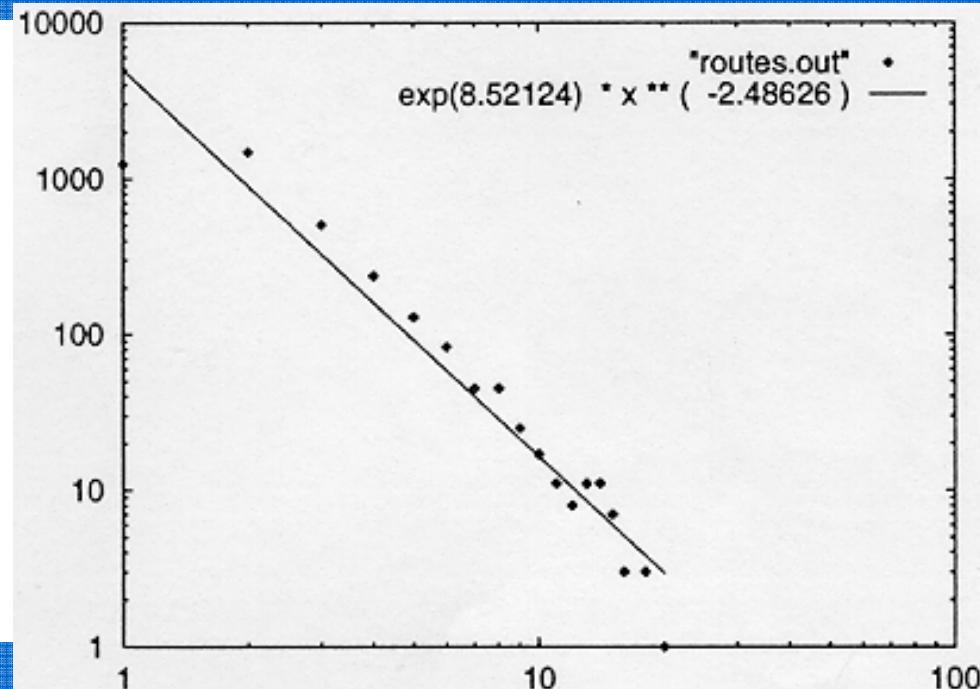
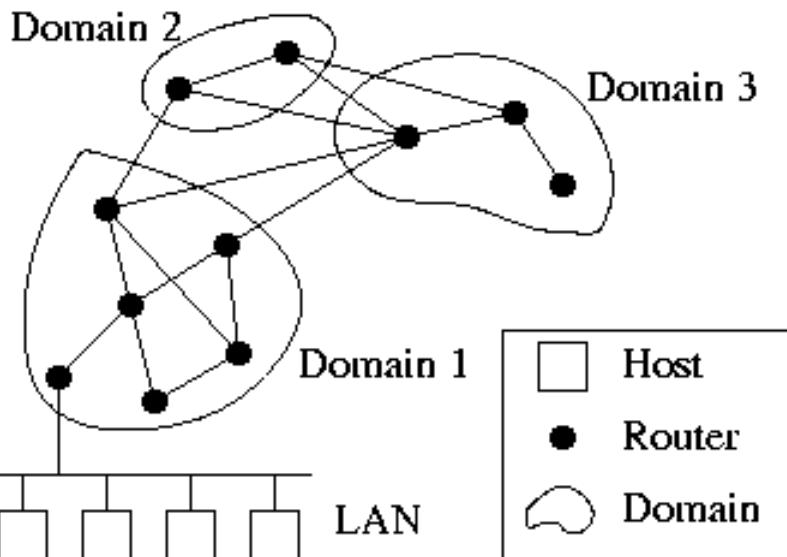
Scale-free Networks

- The number of nodes (N) is not fixed
 - Networks continuously expand by additional new nodes
 - WWW: addition of new nodes
 - Citation: publication of new papers
- The attachment is not uniform
 - A node is linked with higher probability to a node that already has a large number of links
 - WWW: new documents link to well known sites (CNN, Yahoo, Google)
 - Citation: Well cited papers are more likely to be cited again

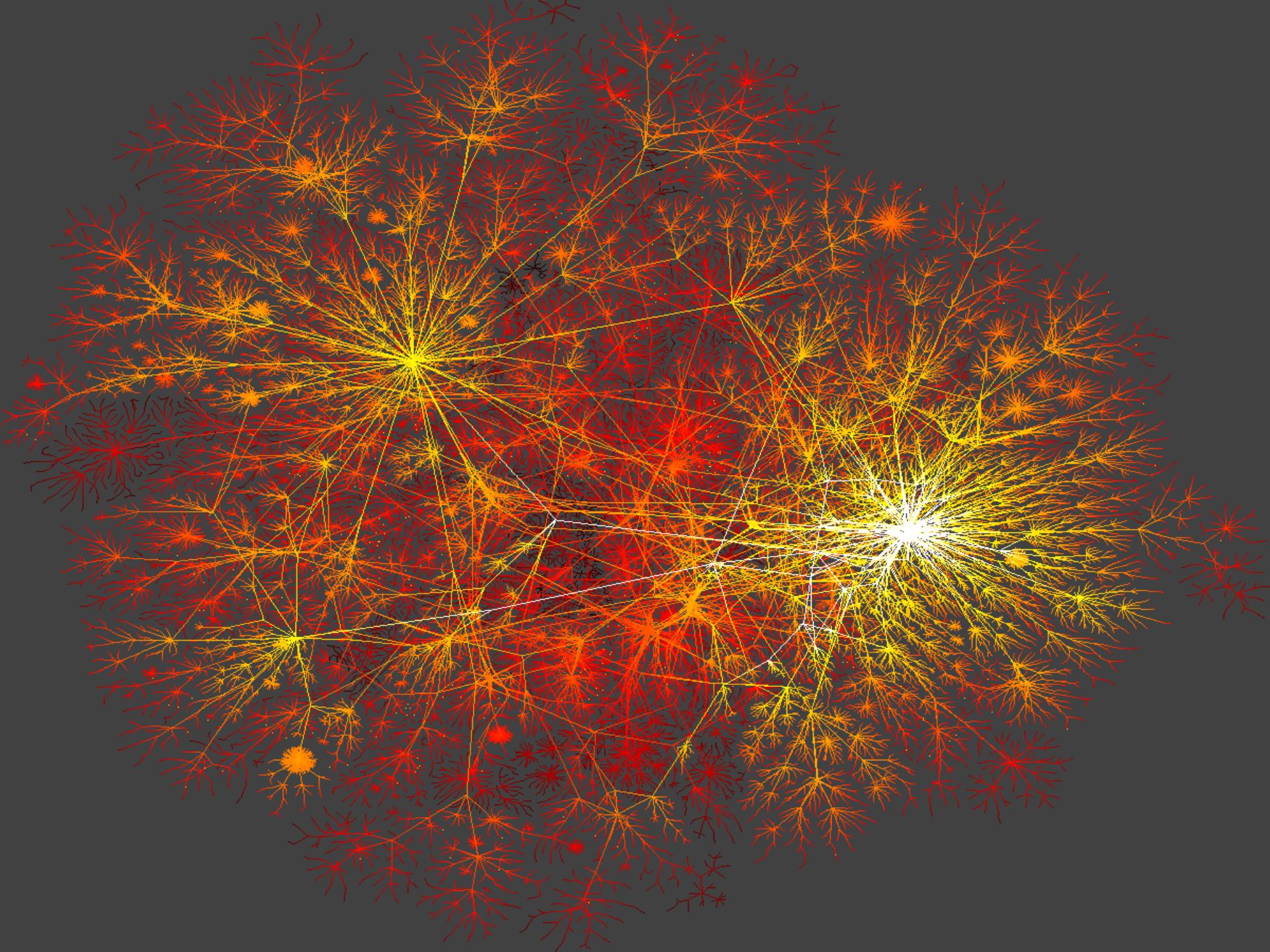
Case1: Internet Backbone

Nodes: computers, routers

Links: physical lines



(Faloutsos, Faloutsos and Faloutsos, 1999)



Social Network Analysis

- Social Network Introduction
- Statistics and Probability Theory
- Models of Social Network Generation
- Networks in Biological System
- Mining on Social Network
- Summary



Information on the Social Network

- Heterogeneous, multi-relational data represented as a graph or network
 - Nodes are objects
 - May have different kinds of objects
 - Objects have attributes
 - Objects may have labels or classes
 - Edges are links
 - May have different kinds of links
 - Links may have attributes
 - Links may be directed, are not required to be binary
- Links represent relationships and interactions between objects - rich content for mining

What is New for Link Mining Here

- Traditional machine learning and data mining approaches assume:
 - A random sample of homogeneous objects from single relation
- Real world data sets:
 - Multi-relational, heterogeneous and semi-structured
- Link Mining
 - Newly emerging research area at the intersection of research in social network and link analysis, hypertext and web mining, graph mining, relational learning and inductive logic programming

A Taxonomy of Common Link Mining Tasks

- Object-Related Tasks
 - Link-based object ranking
 - Link-based object classification
 - Object clustering (group detection)
 - Object identification (entity resolution)
- Link-Related Tasks
 - Link prediction
- Graph-Related Tasks
 - Subgraph discovery
 - Graph classification
 - Generative model for graphs

What Is a Link in Link Mining?

- Link: relationship among data
- Two kinds of linked networks
 - homogeneous vs. heterogeneous
- Homogeneous networks
 - Single object type and single link type
 - Single model social networks (e.g., friends)
 - WWW: a collection of linked Web pages
- Heterogeneous networks
 - Multiple object and link types
 - Medical network: patients, doctors, disease, contacts, treatments
 - Bibliographic network: publications, authors, venues

PageRank: Capturing Page Popularity (Brin & Page'98)

- Intuitions
 - Links are like citations in literature
 - A page that is cited often can be expected to be more useful in general
- PageRank is essentially “citation counting”, but improves over simple counting
 - Consider “indirect citations” (being cited by a highly cited paper counts a lot...)
 - Smoothing of citations (every page is assumed to have a non-zero citation count)
- PageRank can also be interpreted as random surfing (thus capturing popularity)

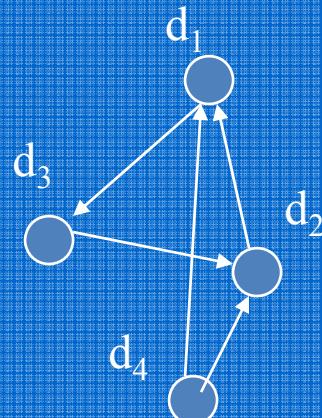
The PageRank Algorithm (Brin & Page'98)

Random surfing model:

At any page,

With prob. α , randomly jumping to a page

With prob. $(1 - \alpha)$, randomly picking a link to follow



$$M = \begin{bmatrix} 0 & 0 & 1/2 & 1/2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 \end{bmatrix}$$

“Transition matrix”

Same as
 α/N (why?)

$$p_{t+1}(d_i) = (1 - \alpha) \sum_{d_j \in IN(d_i)} m_{ji} p_t(d_j) + \alpha \sum_k \frac{1}{N} p_t(d_k)$$

$$p(d_i) = \sum_k \left[\frac{1}{N} \alpha + (1 - \alpha) m_{ki} \right] p(d_k)$$

Stationary (“stable”) distribution, so we ignore time

Initial value $p(d) = 1/N$

Iterate until converge

Essentially an eigenvector problem....

$$\bar{p} = (\alpha I + (1 - \alpha)M)^T \bar{p}$$

$$I_{ij} = 1/N$$

Link Prediction

- Predict whether a link exists between two entities, based on attributes and other observed links
- Applications
 - **Web**: predict if there will be a link between two pages
 - **Citation**: predicting if a paper will cite another paper
 - **Epidemics**: predicting who a patient's contacts are
- Methods
 - Often viewed as a binary classification problem
 - Local conditional probability model, based on structural and attribute features
 - Difficulty: sparseness of existing links
 - Collective prediction, e.g., Markov random field model

Multirelational Data Mining

Multirelational Data Mining

- Classification over multiple-relations in databases
 - Clustering over multi-relations by user-guidance
 - LinkClus: Efficient clustering by exploring the power law distribution
 - Distinct: Distinguishing objects with identical names by link analysis
 - Mining across multiple heterogeneous data and information repositories
 - Summary
- 

Outline

Theme: “Knowledge is power, but knowledge is hidden in massive links”

- Starting with PageRank and HITS
- CrossMine: Classification of multi-relations by link analysis
- CrossClus: Clustering over multi-relations by user-guidance
- More recent work and conclusions

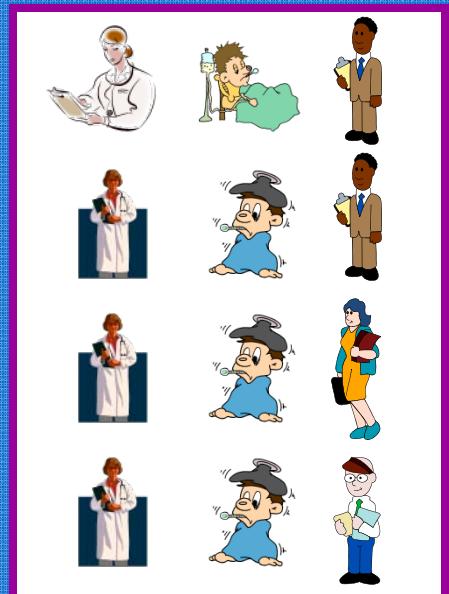


Traditional Data Mining

- Work on single “flat” relations



flatten

- Lose information of linkages and relationships
- Cannot utilize information of database structures or schemas

Multi-Relational Data Mining (MRDM)

- Motivation
 - Most structured data are stored in relational databases
 - MRDM can utilize linkage and structural information
- Knowledge discovery in multi-relational environments
 - Multi-relational rules
 - Multi-relational clustering
 - Multi-relational classification
 - Multi-relational linkage analysis
 - ...

Applications of MRDM

- e-Commerce: discovering patterns involving customers, products, manufacturers, ...
- Bioinformatics/Medical databases: discovering patterns involving genes, patients, diseases, ...
- Networking security: discovering patterns involving hosts, connections, services, ...
- Many other relational data sources
 - Example: Evidence Extraction and Link Discovery (EELD): A DARPA-funding project that emphasizes multi-relational and multi-database linkage analysis

Importance of Multi-relational Classification (from EELD Program

Description)

- The objective of the EELD Program is to research, develop, demonstrate, and transition critical technology that will enable significant improvement in our ability to detect asymmetric threats ..., e.g., a loosely organized terrorist group.
- ... Patterns of activity that, in isolation, are of limited significance but, when combined, are indicative of potential threats, will need to be learned.
- Addressing these threats can only be accomplished by developing a new level of autonomic information surveillance and analysis to extract, discover, and link together sparse evidence from vast amounts of data sources, in different formats and with differing types and degrees of structure, to represent and evaluate the significance of the related evidence, and to learn patterns to guide the extraction, discovery, linkage and evaluation processes.

MRDM Approaches

- Inductive Logic Programming (ILP)
 - Find models that are coherent with background knowledge
- Multi-relational Clustering Analysis
 - Clustering objects with multi-relational information
- Probabilistic Relational Models
 - Model cross-relational probabilistic distributions
- Efficient Multi-Relational Classification
 - The CrossMine Approach [Yin et al, 2004]

Inductive Logic Programming (ILP)

- Find a hypothesis that is consistent with background knowledge (training data)
 - FOIL, Golem, Progol, TILDE, ...
- Background knowledge
 - Relations (predicates), Tuples (ground facts)

Training examples

Daughter(mary, ann)	+
Daughter(eve, tom)	+
Daughter(tom, ann)	-
Daughter(eve, ann)	-

Background knowledge

Parent(ann, mary)
Parent(ann, tom)
Parent(tom, eve)
Parent(tom, ian)

Female(ann)
Female(mary)
Female(eve)

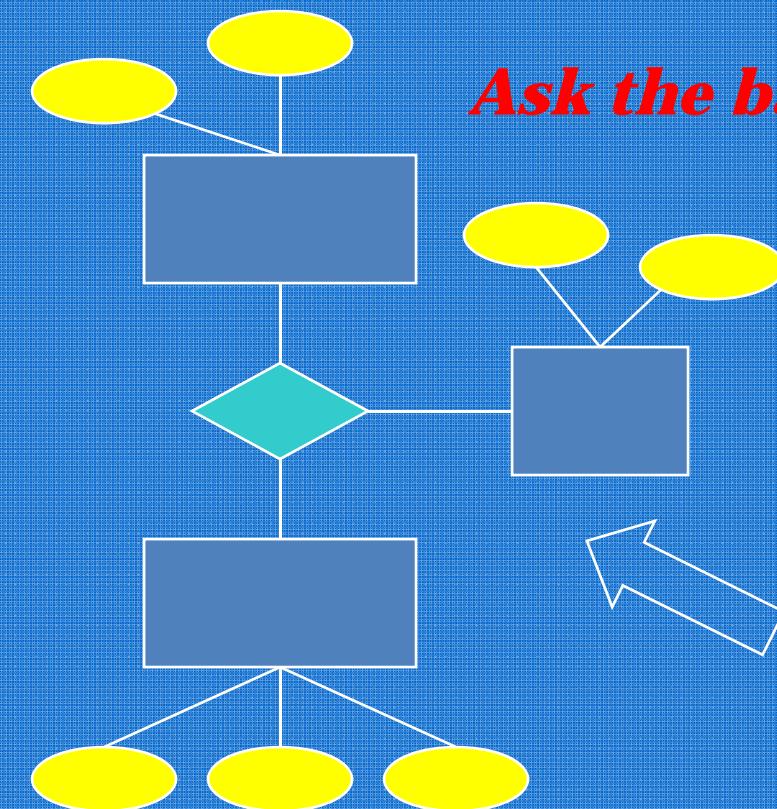
Inductive Logic Programming (ILP)

- Hypothesis
 - The hypothesis is usually a set of rules, which can predict certain attributes in certain relations
 - $\text{Daughter}(X,Y) \leftarrow \text{female}(X), \text{parent}(Y,X)$

Automatically Classifying Objects Using Multiple Relations

- Why not convert multiple relational data into a single table by joins?
 - Relational databases are designed by domain experts via semantic modeling (e.g., E-R modeling)
 - Indiscriminative joins may loose some essential information
 - One universal relation may not be appealing to efficiency, scalability and semantics preservation
- Our approach to multi-relational classification:
 - Automatically classifying objects using multiple relations

An Example: Loan Applications



Ask the backend database

Approve or not?

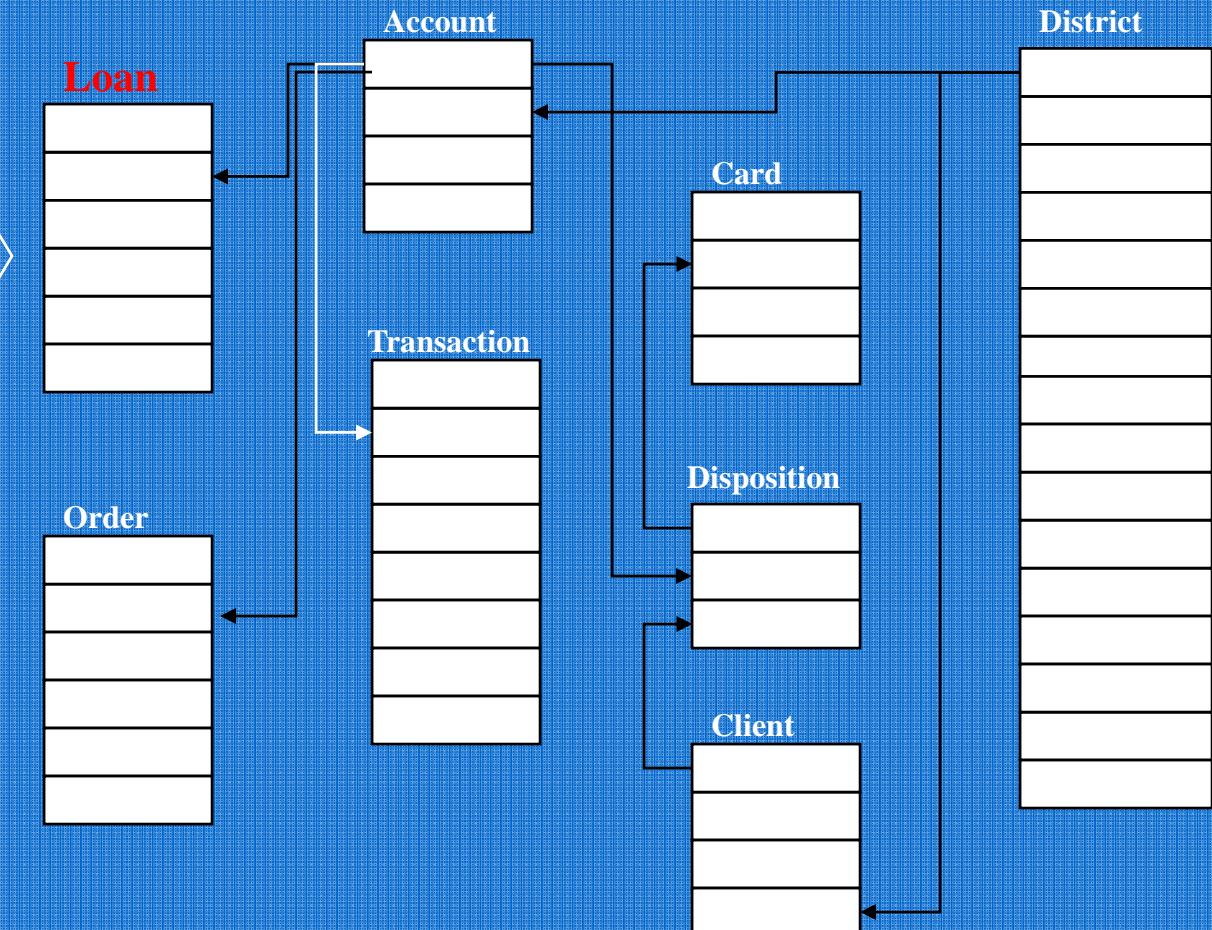
Apply for loan



The Backend Database

Target relation:

Each tuple has a class label, indicating whether a loan is paid on time.

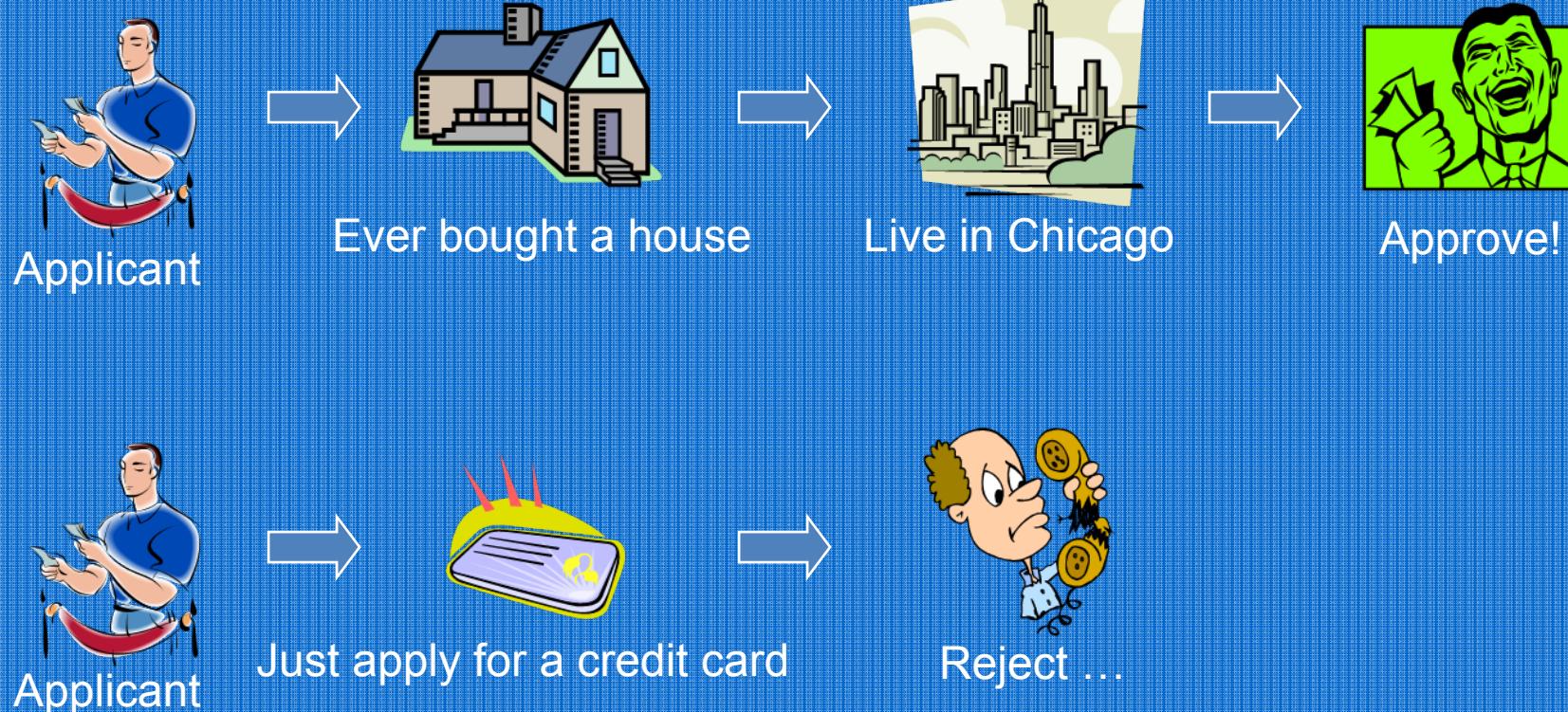


How to make decisions to loan applications?

Roadmap

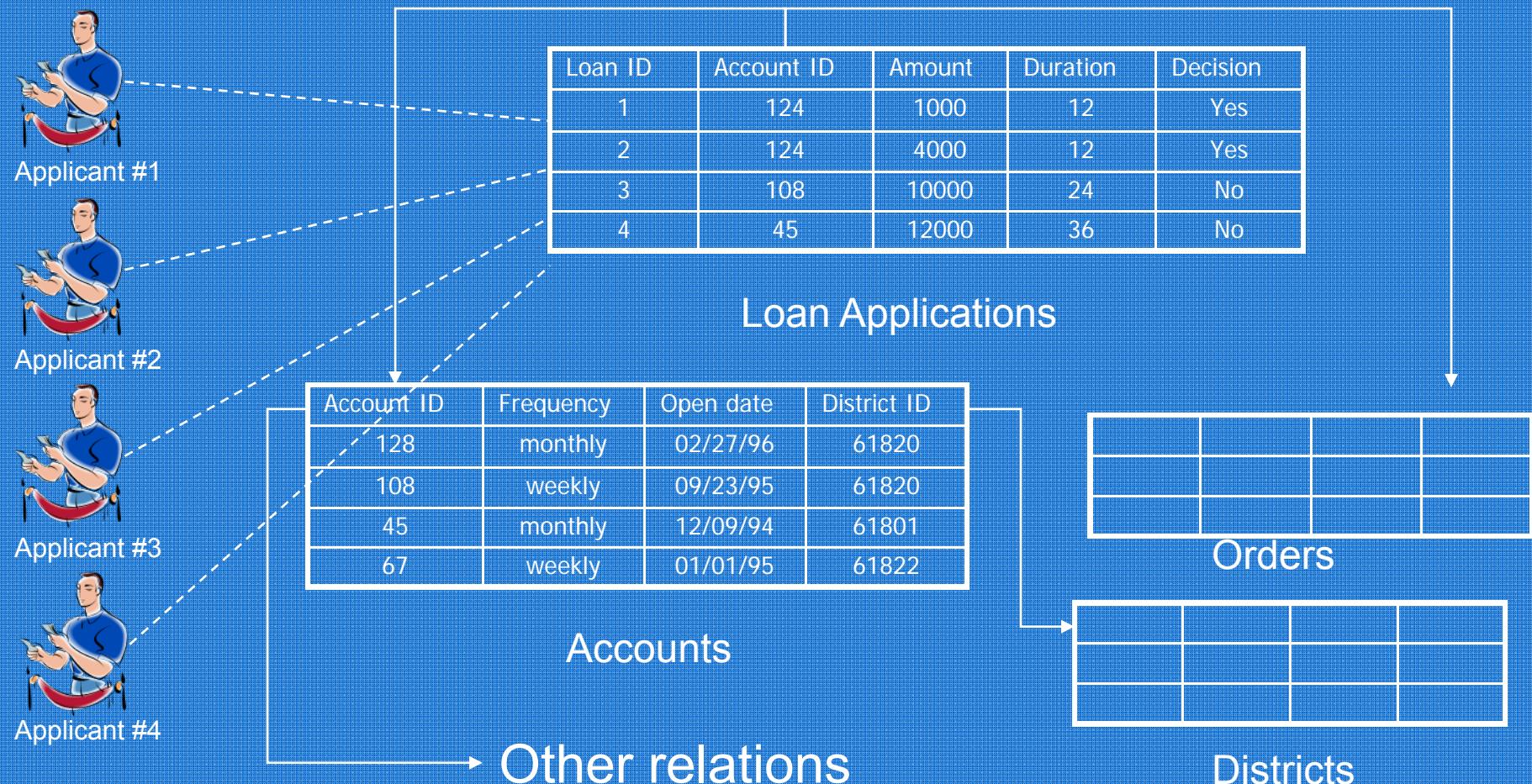
- Motivation
- Rule-based Classification
- Tuple ID Propagation
- Rule Generation
- Negative Tuple Sampling
- Performance Study

Rule-based Classification



Rule Generation

- Search for good predicates across multiple relations



Previous Approaches

- Inductive Logic Programming (ILP)
 - To build a rule
 - Repeatedly find the best predicate
 - To evaluate a predicate on relation R , first join target relation with R
 - Not scalable because
 - Huge search space (numerous candidate predicates)
 - Not efficient to evaluate each predicate
 - To evaluate a predicate
$$\text{Loan}(L, +) :- \text{Loan}(L, A, ?, ?, ?, ?), \text{Account}(A, ?, \text{'monthly'}, ?)$$
first join loan relation with account relation
- CrossMine is more scalable and more than one hundred times faster on datasets with reasonable sizes

Rule Generation

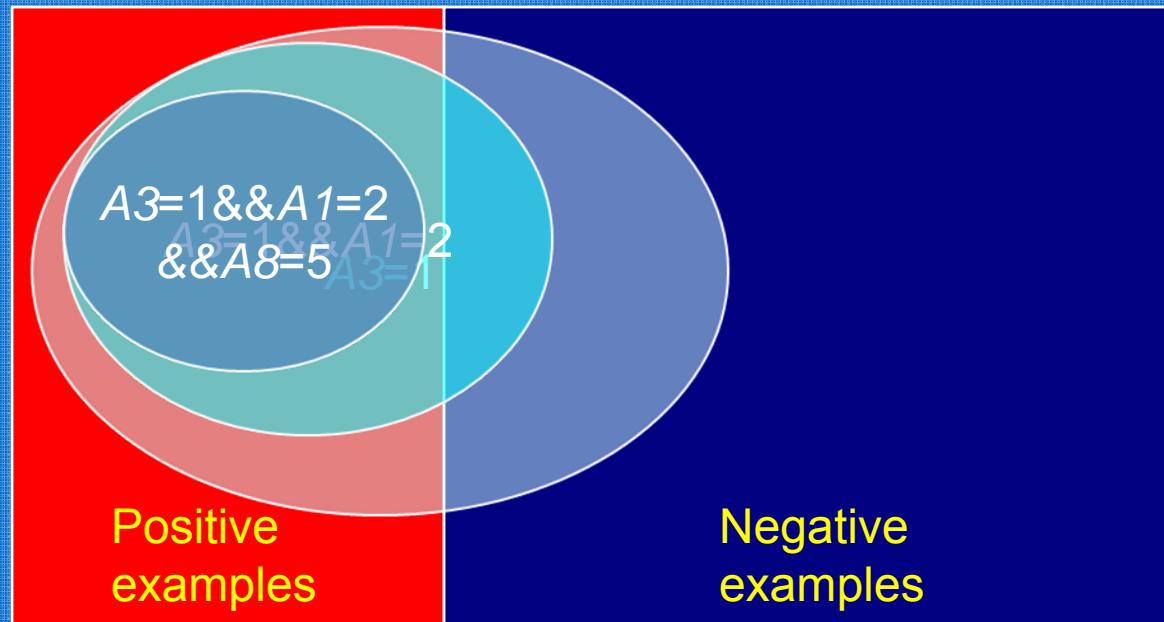
- To generate a rule

while(true)

 find the best predicate p

if foil-gain(p)>threshold **then** add p to current rule

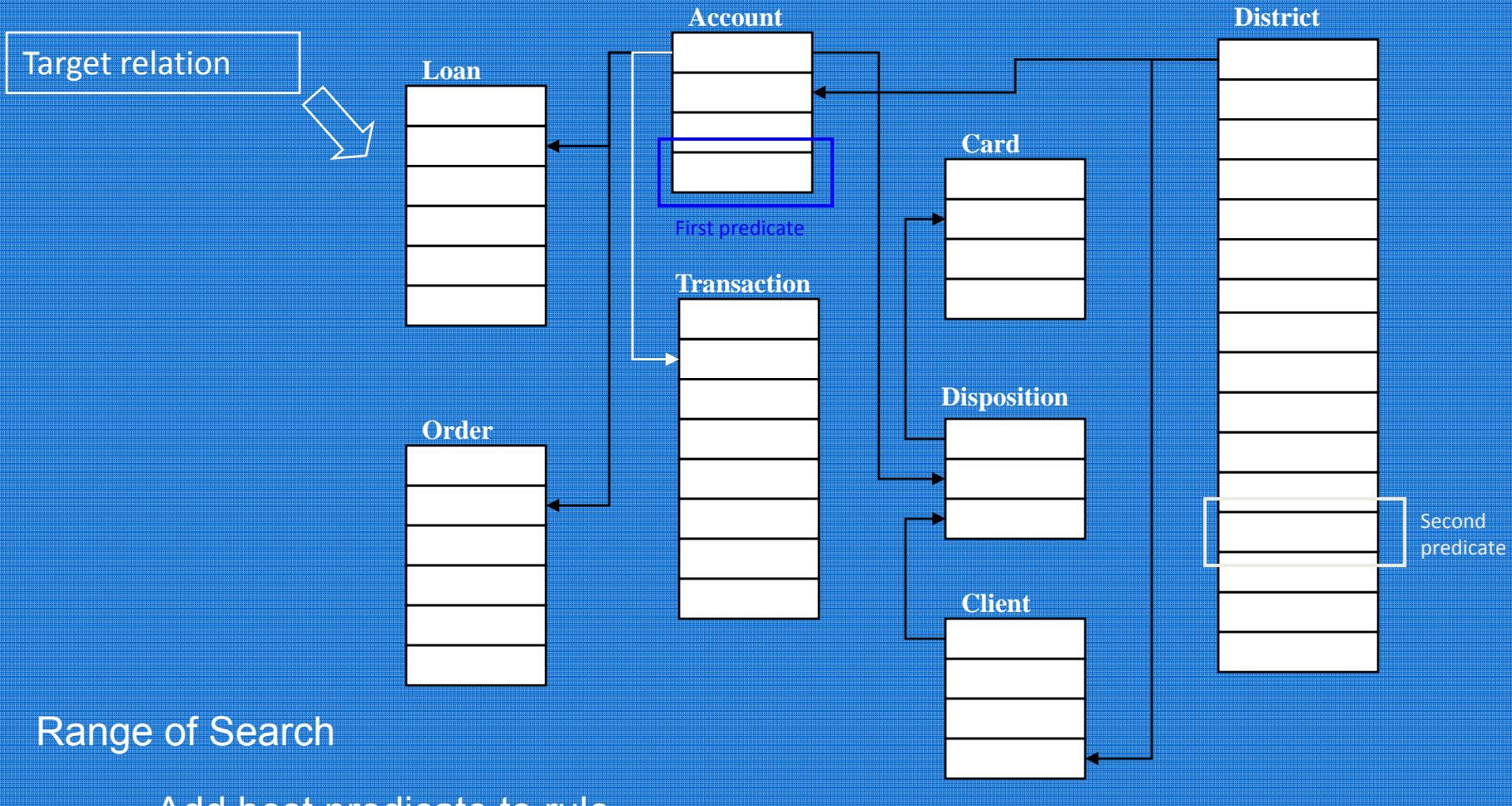
else break



Rule Generation

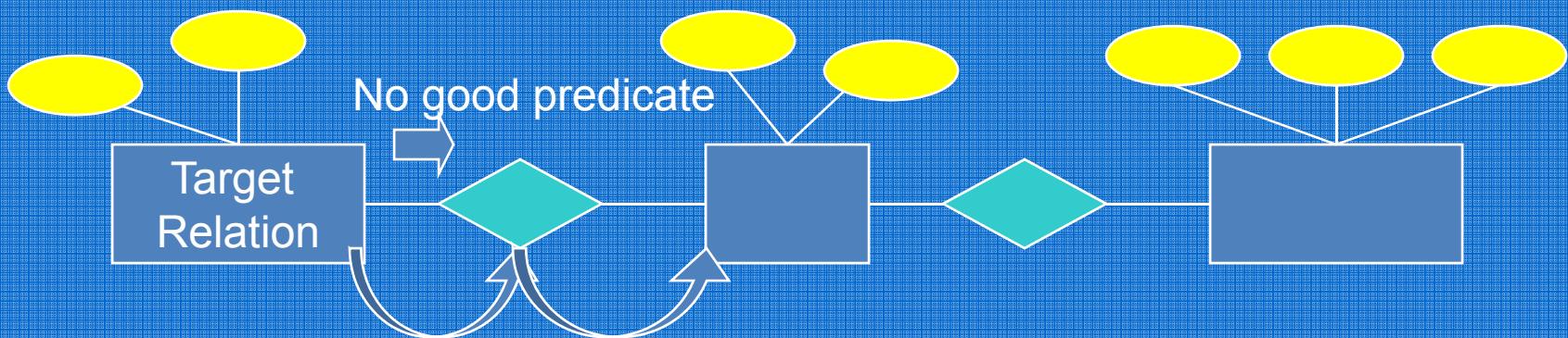
- Start from the target relation
 - Only the target relation is active
- Repeat
 - Search in all active relations
 - Search in all relations joinable to active relations
 - Add the best predicate to the current rule
 - Set the involved relation to active
- Until
 - The best predicate does not have enough gain
 - Current rule is too long

Rule Generation: Example



Look-one-ahead in Rule Generation

- Two types of relations: Entity and Relationship
- Often cannot find useful predicates on relations of relationship



- Solution of CrossMine:
 - When propagating IDs to a relation of relationship, propagate one more step to next relation of entity.

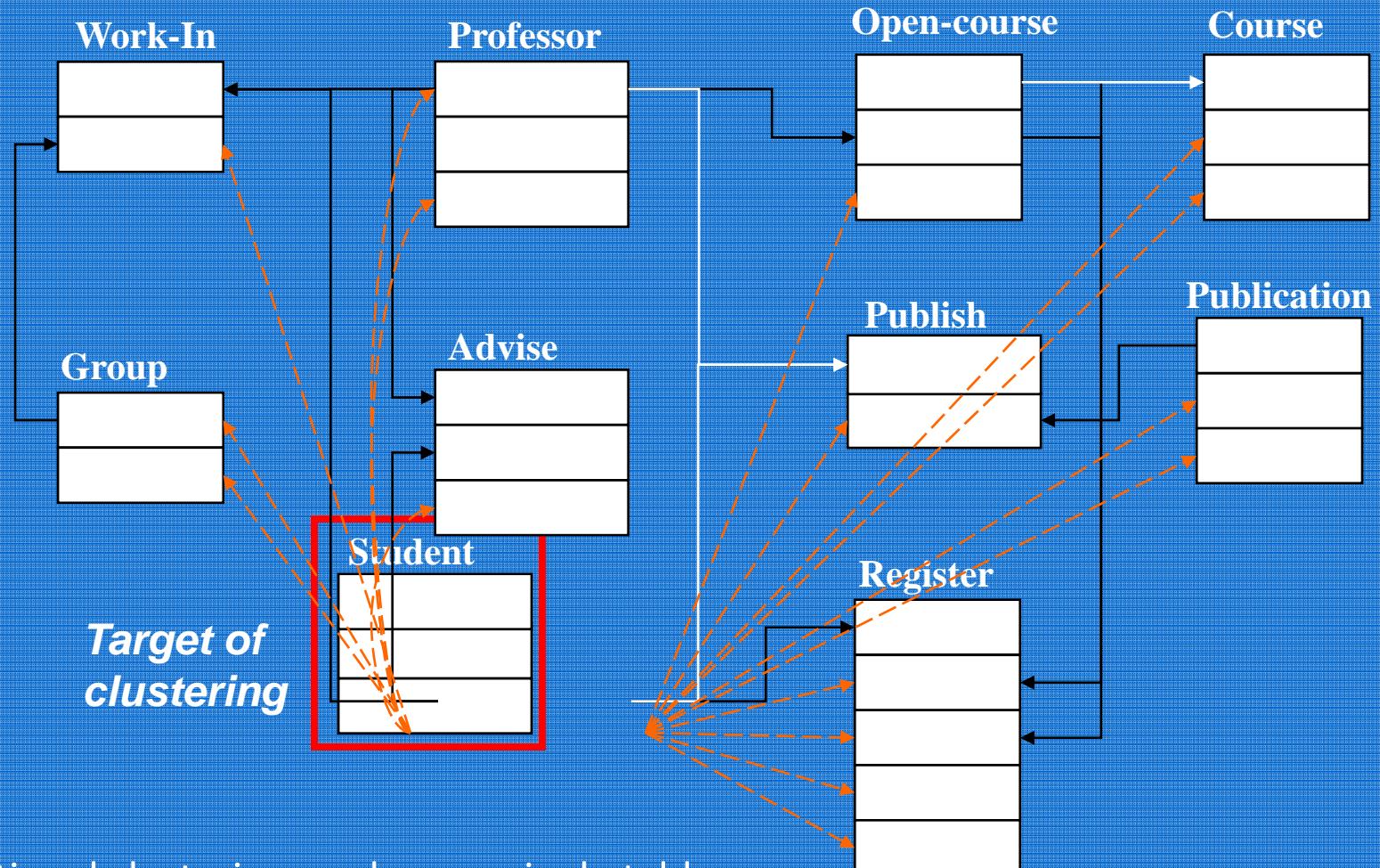
Multirelational Data Mining

- Classification over multiple-relations in databases
- Clustering over multi-relations by user-guidance 
- LinkClus: Efficient clustering by exploring the power law distribution
- Distinct: Distinguishing objects with identical names by link analysis
- Mining across multiple heterogeneous data and information repositories
- Summary

Multi-Relational and Multi-DB Mining

- Classification over multiple-relations in databases
- Clustering over multi-relations by User-Guidance 
- Mining across multi-relational databases
- Mining across multiple heterogeneous data and information repositories
- Summary

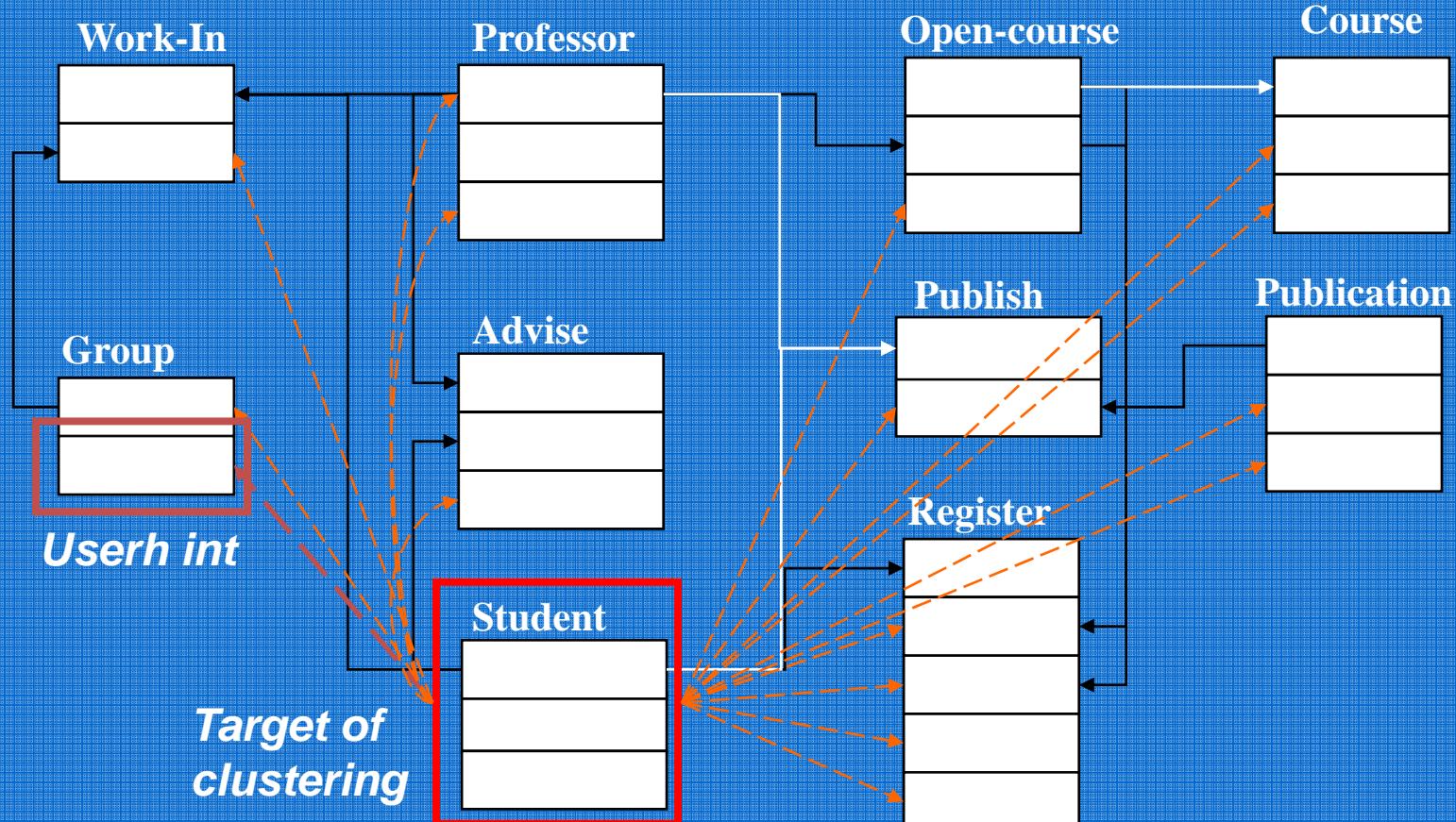
Motivation 1: Multi-Relational Clustering



- Traditional clustering works on a single table
- Most data is semantically linked with multiple relations

Thus we need information in multiple relations

Motivation 2: User-Guided Clustering



- User usually has a goal of clustering, e.g., clustering students by research area
- User specifies his clustering goal to CrossClus

Comparing with Classification



- User-specified *feature* (in the form of *attribute*) is used as a hint, not class labels
 - The attribute may contain too many or too few distinct values
 - E.g., a user may want to cluster students into 20 clusters instead of 3
 - Additional features need to be included in cluster analysis

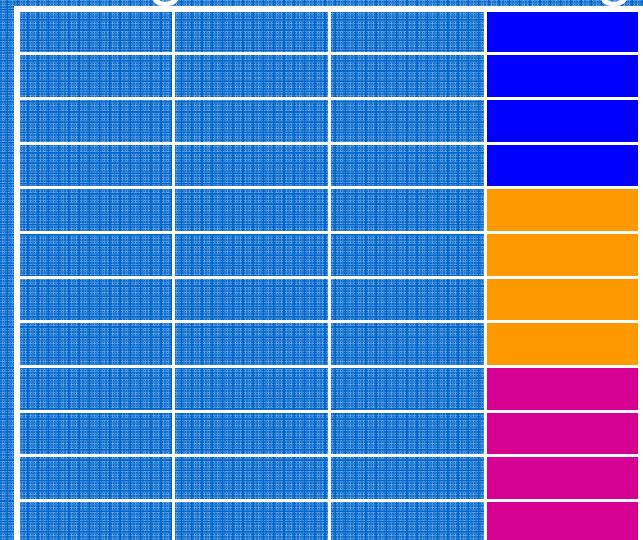
Comparing with Semi-supervised Clustering

- Semi-supervised clustering [Wagstaff, et al' 01, Xing, et al.'02]
 - User provides a training set consisting of “similar” and “dissimilar” pairs of objects
- User-guided clustering
 - User specifies an attribute as a hint, and more relevant features are found for clustering

Semi-supervised clustering

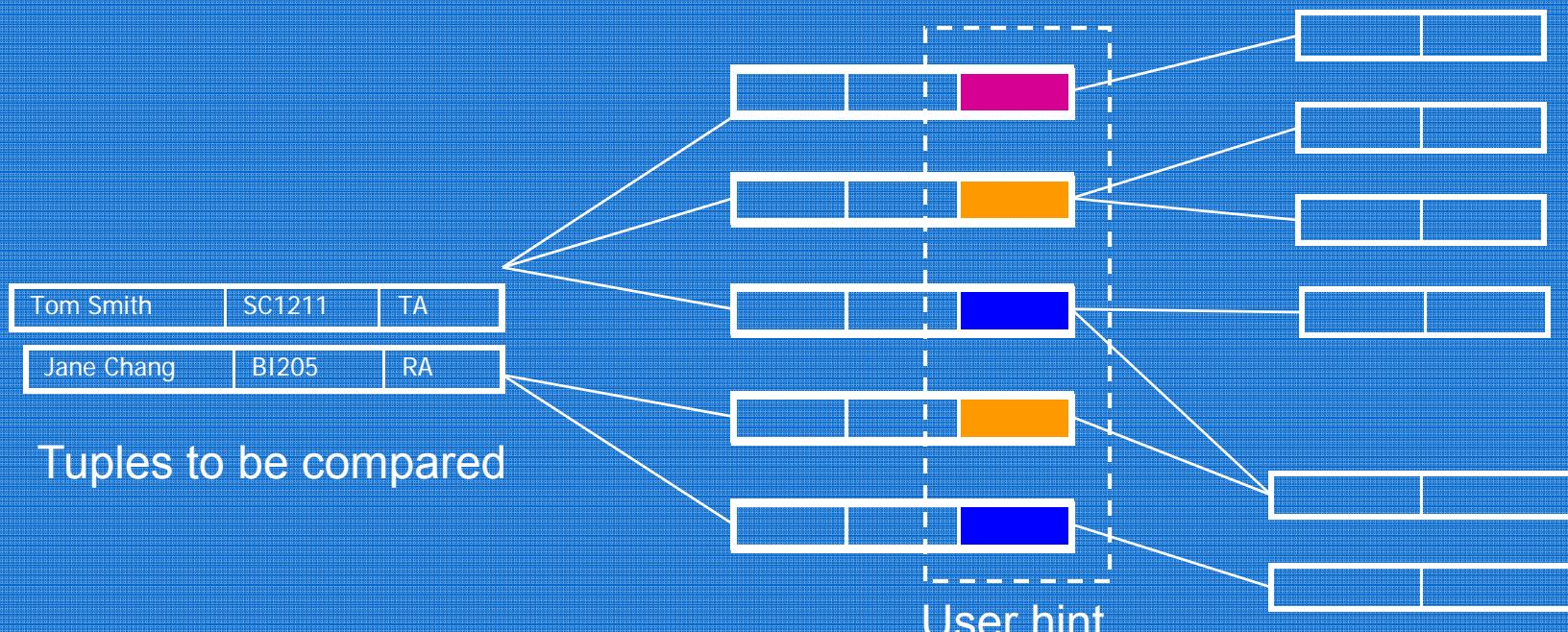


User-guided clustering



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*



Searching for Pertinent Features

- Different features convey different aspects of information

Research area

Research group area

Conferences of papers

Advisor

Demographic info

Permanent address

Nationality

Academic Performances

GPA

GRE score

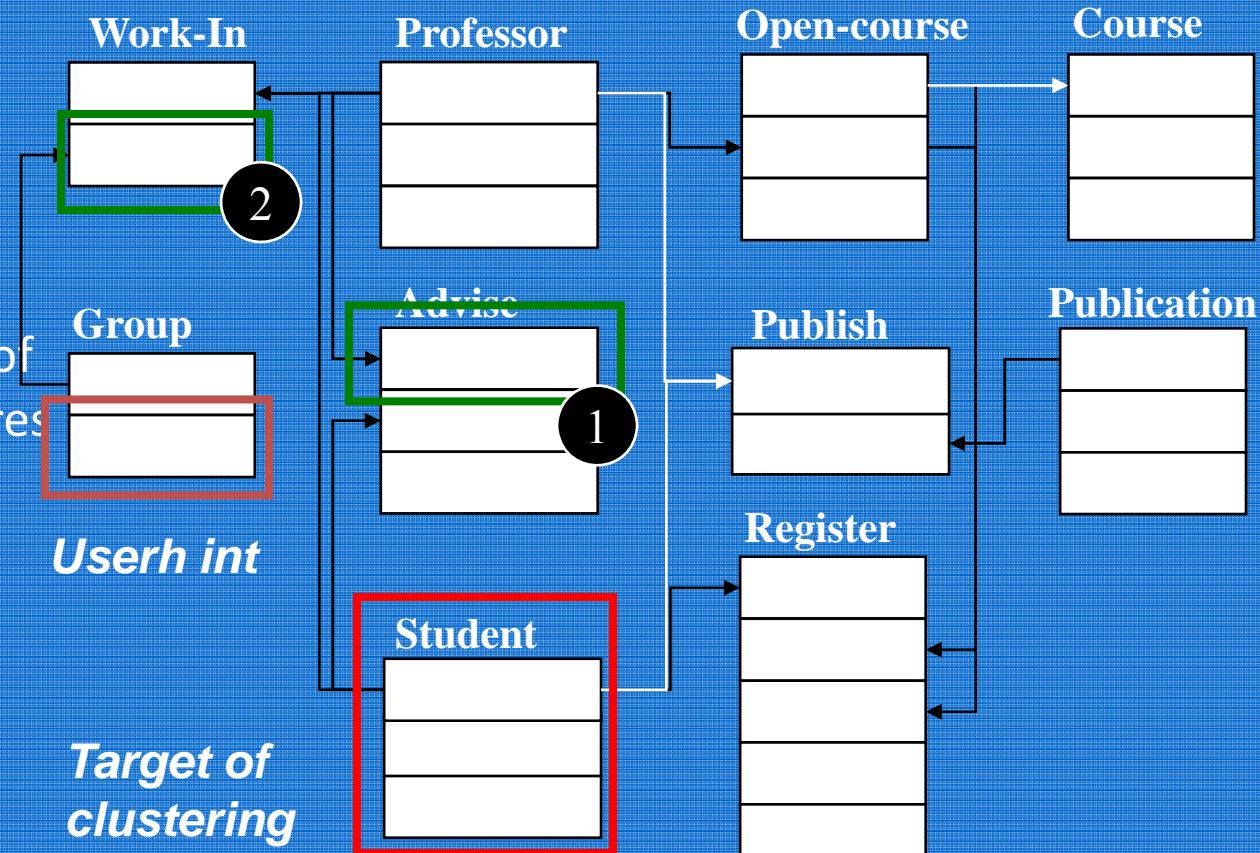
Number of papers

- Features conveying same aspect of information usually cluster objects in more similar ways
 - research group areas vs. conferences of publications
- Given user specified feature
 - Find pertinent features by computing feature similarity

Heuristic Search for Pertinent Features

Overall procedure

1. Start from the user-specified feature
2. Search in neighborhood of existing pertinent features
3. Expand search range gradually



- Tuple ID propagation [Yin, et al.'04] is used to create multi-relational features
 - IDs of target tuples can be propagated along any join path, from which we can find tuples joinable with each target tuple

Roadmap

1. Overview
2. Feature Pertinence
3. Searching for Features
4. Clustering
5. Experimental Results

- Given a set of L pertinent features f_1, \dots, f_L , similarity between two objects

$$\text{sim}(t_1, t_2) = \sum_{i=1}^L \text{sim}_{f_i}(t_1, t_2) \cdot f_i.\text{weight}$$

- Weight of a feature is determined in feature search by its similarity with other pertinent features
- For clustering, we use CLARANS, a scalable k -medoids [Ng & Han'94] algorithm

Roadmap

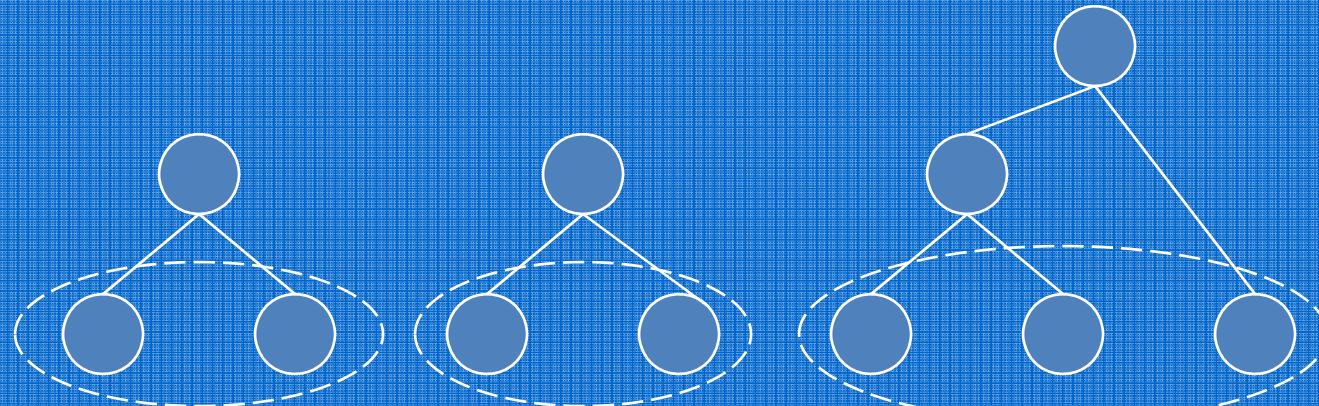
1. Overview
2. Feature Pertinence
3. Searching for Features
4. Clustering
5. Experimental Results

How to Measure Similarity between Clusters?

- Single-link (highest similarity between points in two clusters)?
 - No, because references to different objects can be connected.
- Complete-link (minimum similarity between them)?
 - No, because references to the same object may be weakly connected.
- Average-link (average similarity between points in two clusters)?
 - A better measure

Clustering Procedure

- Procedure
 - Initialization: Use each reference as a cluster
 - Keep finding and merging the most similar pair of clusters
 - Until no pair of clusters is similar enough



Efficient Computation

- In agglomerative hierarchical clustering, one needs to repeatedly compute similarity between clusters
 - When merging clusters C_1 and C_2 into C_3 , we need to compute the similarity between C_3 and any other cluster
 - Very expensive when clusters are large
- We invent methods to compute similarity incrementally
 - Neighborhood similarity

$$Resem(C_3, C_i) = \frac{|C_1| \cdot Resem(C_1, C_i) + |C_2| \cdot Resem(C_2, C_i)}{|C_1| + |C_2|}.$$

$$WalkProb_Q(C_3 \rightarrow C_i) = \frac{|C_1| \cdot WalkProb_Q(C_1 \rightarrow C_i) + |C_2| \cdot WalkProb_Q(C_2 \rightarrow C_i)}{|C_1| + |C_2|}.$$

Multirelational Data Mining

- Classification over multiple-relations in databases
- Clustering over multi-relations by user-guidance
- LinkClus: Efficient clustering by exploring the power law distribution
- Distinct: Distinguishing objects with identical names by link analysis
- Mining across multiple heterogeneous data and information repositories
- Summary



Summary

- Knowledge is power, but knowledge is hidden in massive links
- More stories than Web page rank and search
- CrossMine: Classification of multi-relations by link analysis
- CrossClus: Clustering over multi-relations by user-guidance
- LinkClus: Efficient clustering by exploring the power law distribution
- Distinct: Distinguishing objects with identical names by link analysis
- Much more to be explored!

Review Questions

- State the importance of sliding window model to analyze stream data?
- Write a note an data stream management systems(DSMS)
- State the difference between one-time query and continuous query.
- How does the lossy country algorithm find frequent items?
- Give a note on stream query processing?
- What is a time –series database?
- Define sequential pattern mining?
- What is periodicity analysis?
- Distinguish between full periodic pattern and partial periodic pattern
- State Markov chain model
- State the importance of synopses in context with screen data?
- State the need for biological sequence analysis?
- Discuss about constraint based mining?
- What is a social network?
- Brief out multi relation data mining?

Bibliography

- Data mining concepts and Techniques by Jiawei Han and Micheline Kamber

Mining Object, Spatial, and Multimedia Data—

Mining Object, Spatial and Multi-Media Data

- Mining object data sets
- Mining spatial databases and data warehouses
 - Spatial DBMS
 - Spatial Data Warehousing
 - Spatial Data Mining
 - Spatiotemporal Data Mining
- Mining multimedia data
- Summary



Mining Complex Data Objects: Generalization of Structured Data

- Set-valued attribute
 - Generalization of each value in the set into its corresponding higher-level concepts
 - Derivation of the general behavior of the set, such as the number of elements in the set, the types or value ranges in the set, or the weighted average for numerical data
 - E.g., $hobby = \{tennis, hockey, chess, violin, PC_games\}$ generalizes to $\{sports, music, e_games\}$
- List-valued or a sequence-valued attribute
 - Same as set-valued attributes except that the order of the elements in the sequence should be observed in the generalization

Generalizing Spatial and Multimedia Data

- **Spatial data:**
 - Generalize detailed geographic points into clustered regions, such as business, residential, industrial, or agricultural areas, according to land usage
 - Require the merge of a set of geographic areas by spatial operations
- **Image data:**
 - Extracted by aggregation and/or approximation
 - Size, color, shape, texture, orientation, and relative positions and structures of the contained objects or regions in the image
- **Music data:**
 - Summarize its melody: based on the approximate patterns that repeatedly occur in the segment
 - Summarized its style: based on its tone, tempo, or the major musical instruments played

Generalizing Object Data

- Object identifier
 - generalize to the lowest level of class in the class/subclass hierarchies
- Class composition hierarchies
 - generalize only those **closely related in semantics** to the current one
- Construction and mining of object cubes
 - Extend the attribute-oriented induction method
 - Apply a sequence of class-based generalization operators on different attributes
 - Continue until getting a small number of generalized objects that can be summarized as a concise in high-level terms
 - Implementation
 - Examine each attribute, generalize it to simple-valued data
 - Construct a multidimensional data cube (**object cube**)
 - Problem: it is not always desirable to generalize a set of values to single-valued data

Ex.: Plan Mining by Divide and Conquer

- Plan: a sequence of actions
 - E.g., Travel (flight): <traveler, departure, arrival, d-time, a-time, airline, price, seat>
- Plan mining: extraction of important or significant generalized (sequential) patterns from a planbase (a large collection of plans)
 - E.g., Discover travel patterns in an air flight database, or
 - find significant patterns from the sequences of actions in the repair of automobiles
- Method
 - Attribute-oriented induction on sequence data
 - A generalized travel plan: <small-big*-small>
 - Divide & conquer: Mine characteristics for each subsequence
 - E.g., big*: same airline, small-big: nearby region

A Travel Database for Plan Mining

- Example: Mining a travel planbase

Travel plan table

plan#	action#	departure	depart_time	arrival	arrival_time	airline	...
1	1	ALB	800	JFK	900	TWA	...
1	2	JFK	1000	ORD	1230	UA	...
1	3	ORD	1300	LAX	1600	UA	...
1	4	LAX	1710	SAN	1800	DAL	...
2	1	SPI	900	ORD	950	AA	...
.
.
.

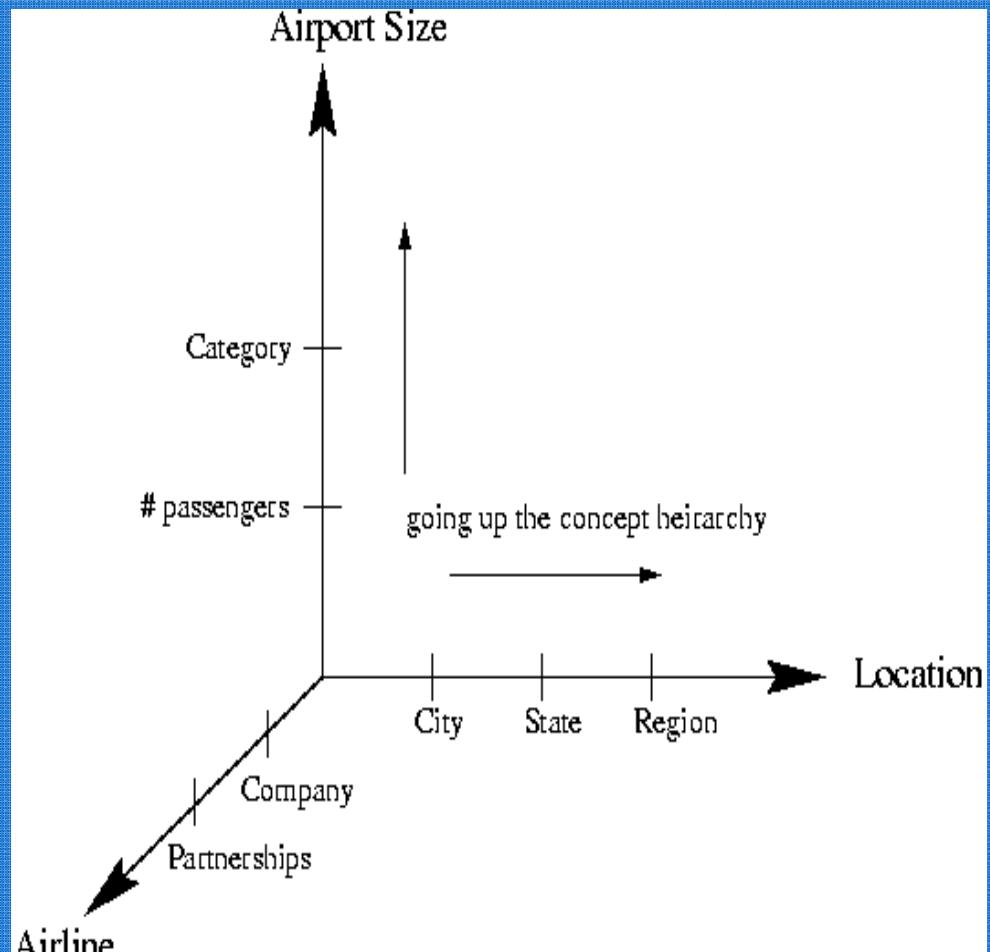
Airport info table

airport_code	city	state	region	airport_size	...
1	1	ALB		800	...
1	2	JFK		1000	...
1	3	ORD		1300	...
1	4	LAX		1710	...
2	1	SPI		900	...
.
.
.

Multidimensional Analysis

- Strategy
 - Generalize the planbase in different directions
 - Look for sequential patterns in the generalized plans
 - Derive high-level plans

A multi-D model for the planbase



Mining Object, Spatial and Multi-Media Data

- Mining object data sets
- Mining spatial databases and data warehouses
 - Spatial DBMS
 - Spatial Data Warehousing
 - Spatial Data Mining
 - Spatiotemporal Data Mining
- Mining multimedia data
- Summary



What Is a Spatial Database System?

- Geometric, geographic or spatial data: space-related data
 - Example: Geographic space (2-D abstraction of earth surface), VLSI design, model of human brain, 3-D space representing the arrangement of chains of protein molecule.
- Spatial database system vs. image database systems.
 - Image database system: handling digital raster image (e.g., satellite sensing, computer tomography), may also contain techniques for object analysis and extraction from images and some spatial database functionality.
 - Spatial (geometric, geographic) database system: handling objects in space that have identity and well-defined extents, locations, and relationships.

GIS (Geographic Information System)

- **GIS (Geographic Information System)**
 - Analysis and visualization of geographic data
- Common analysis functions of GIS
 - Search (thematic search, search by region)
 - Location analysis (buffer, corridor, overlay)
 - Terrain analysis (slope/aspect, drainage network)
 - Flow analysis (connectivity, shortest path)
 - Distribution (nearest neighbor, proximity, change detection)
 - Spatial analysis/statistics (pattern, centrality, similarity, topology)
 - Measurements (distance, perimeter, shape, adjacency, direction)

Spatial DBMS (SDBMS)

- SDBMS is a software system that
 - supports spatial data models, spatial ADTs, and a query language supporting them
 - supports spatial indexing, spatial operations efficiently, and query optimization
 - can work with an underlying DBMS
- Examples
 - Oracle Spatial Data Cartridge
 - ESRI Spatial Data Engine

Modeling Spatial Objects

- What needs to be represented?
- Two important alternative views
 - Single objects: distinct entities arranged in space each of which has its own geometric description
 - modeling cities, forests, rivers
 - Spatially related collection of objects: describe space itself (about every point in space)
 - modeling land use, partition of a country into districts

Modeling Single Objects: Point, Line and Region

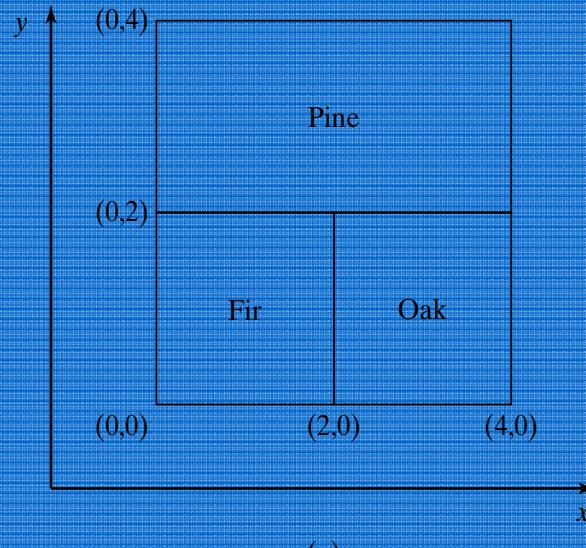
- Point: location only but not extent
- Line (or a curve usually represented by a polyline, a sequence of line segment):
 - moving through space, or connections in space (roads, rivers, cables, etc.)
- Region:
 - Something having extent in 2D-space (country, lake, park). It may have a hole or consist of several disjoint pieces.

Modeling Spatially Related Collection of Objects

- Modeling spatially related collection of objects: plane partitions and networks.
 - A partition: a set of region objects that are required to be disjoint (e.g., a thematic map). There exist often pairs of objects with a common boundary (adjacency relationship).
 - A network: a graph embedded into the plane, consisting of a set of point objects, forming its nodes, and a set of line objects describing the geometry of the edges, e.g., highways, rivers, power supply lines.
 - Other interested spatially related collection of objects: nested partitions, or a digital terrain (elevation) model.

Spatial Data Types and Models

- Field-based model: raster data
 - framework: partitioning of space
- Object-based model: vector model
 - point, line, polygon, Objects, Attributes



Object Viewpoint of Forest Stands		
Area-ID	Dominant Tree Species	Area/Boundary
FS1	Pine	$[(0,2),(4,2),(4,4),(0,4)]$
FS2	Fir	$[(0,0),(2,0),(2,2),(0,2)]$
FS3	Oak	$[(2,0),(4,0),(4,2),(2,2)]$

Field Viewpoint of Forest Stands

$$f(x,y) = \begin{cases} \text{"Pine," } 2 \leq x \leq 4; 2 < y \leq 4 \\ \text{"Fir," } 0 \leq x \leq 2; 0 \leq y \leq 2 \\ \text{"Oak," } 2 < x \leq 4; 0 \leq y \leq 2 \end{cases}$$

Spatial Query Language

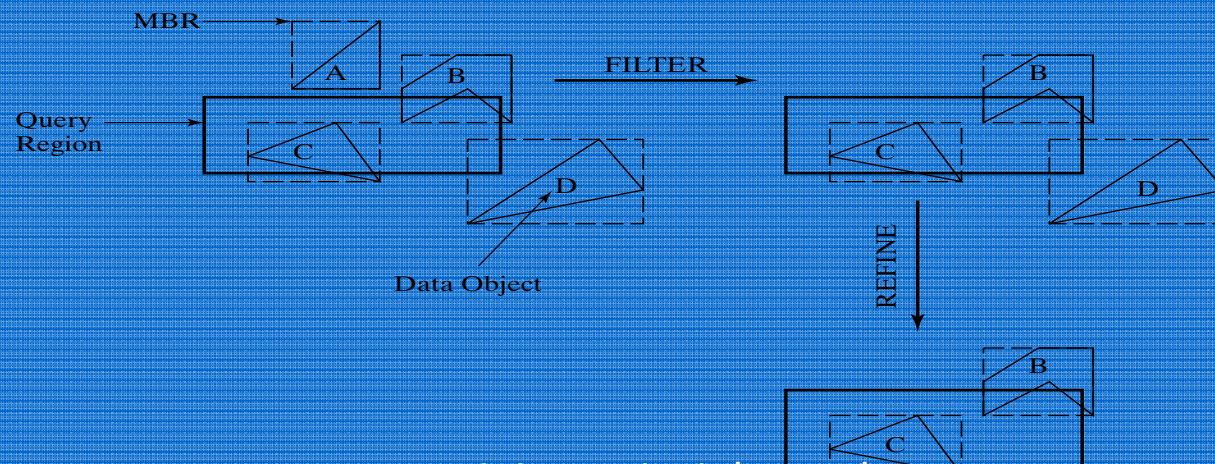
- Spatial query language
 - Spatial data types, e.g. point, line segment, polygon, ...
 - Spatial operations, e.g. overlap, distance, nearest neighbor, ...
 - Callable from a query language (e.g. SQL3) of underlying DBMS

```
SELECT S.name
FROM Senator S
WHERE S.district.Area() > 300
```

- Standards
 - SQL3 (a.k.a. SQL 1999) is a standard for query languages
 - OGIS is a standard for spatial data types and operators
 - Both standards enjoy wide support in industry

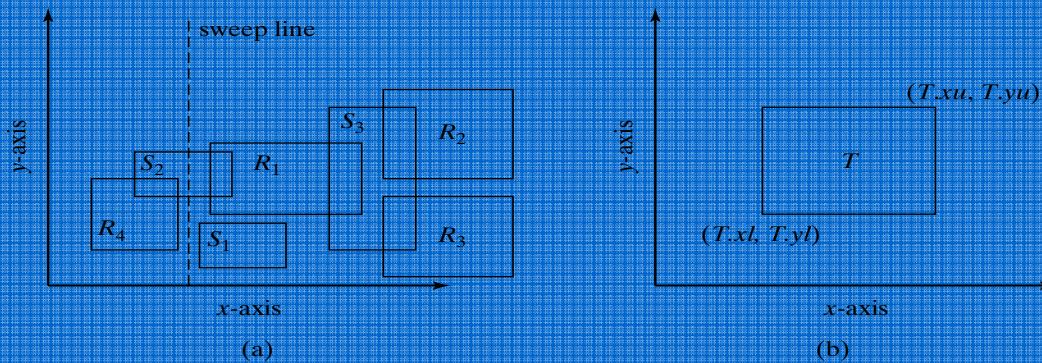
Query Processing

- Efficient algorithms to answer spatial queries
- Common Strategy: filter and refine
 - Filter: Query Region overlaps with MBRs (minimum bounding rectangles) of B, C, D
 - Refine: Query Region overlaps with B, C



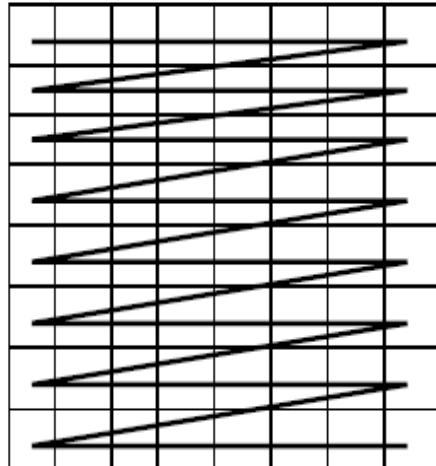
Join Query Processing

- Determining Intersection Rectangle
- Plane Sweep Algorithm
 - Place sweep filter identifies 5 intersections for refinement step

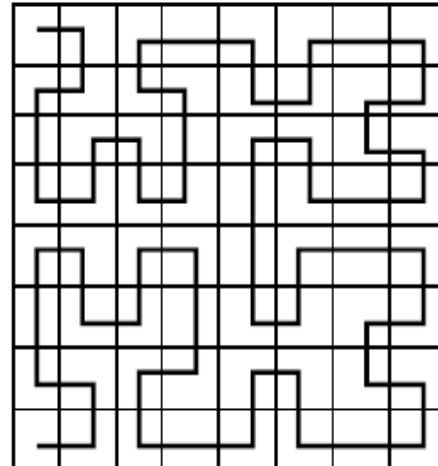


File Organization and Indices

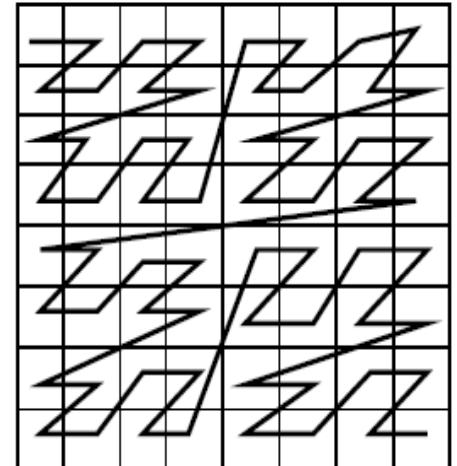
- SDBMS: Dataset is in the secondary storage, e.g. disk
- Space Filling Curves: An ordering on the locations in a multi-dimensional space
 - Linearize a multi-dimensional space
 - Helps search efficiently



Row



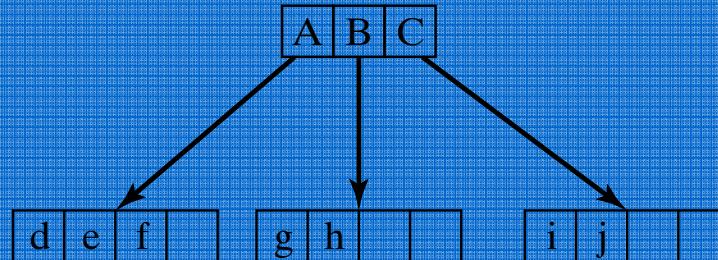
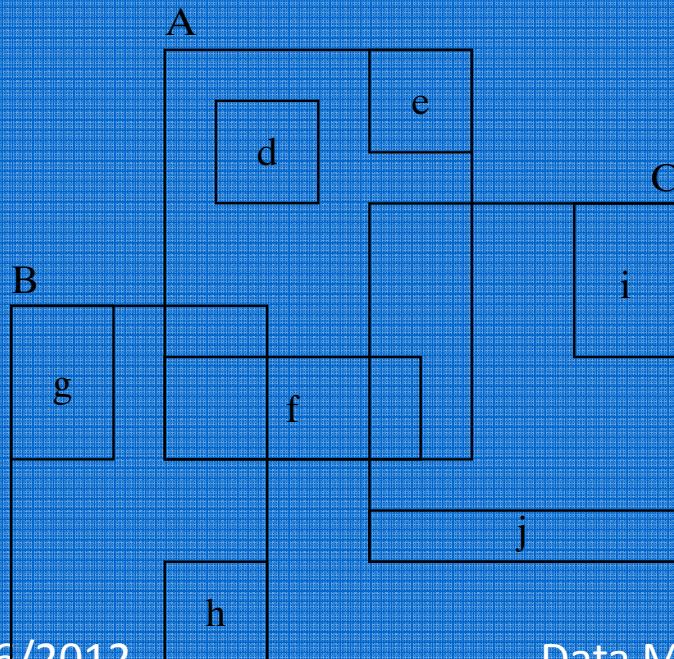
Peano-Hilbert
Data Mining: Principles and
Algorithms



Morton / Z-order

File Organization and Indices

- Spatial Indexing
 - B-tree works on spatial data with space filling curve
 - R-tree: Heighted balanced extention of B+ tree
 - Objects are represented as MBR
 - provides better performance



Spatial Query Optimization

- A spatial operation can be processed using different strategies
- Computation cost of each strategy depends on many parameters
- Query optimization is the process of
 - ordering operations in a query and
 - selecting efficient strategy for each operation
 - based on the details of a given dataset

Spatial Data Warehousing

- **Spatial data warehouse:** Integrated, subject-oriented, time-variant, and nonvolatile spatial data repository
- **Spatial data integration:** a big issue
 - Structure-specific formats (raster- vs. vector-based, OO vs. relational models, different storage and indexing, etc.)
 - Vendor-specific formats (ESRI, MapInfo, Integraph, IDRISI, etc.)
 - Geo-specific formats (geographic vs. equal area projection, etc.)
- **Spatial data cube:** multidimensional spatial database
 - Both dimensions and measures may contain spatial components

Dimensions and Measures in Spatial Data Warehouse



- Dimensions
 - non-spatial
 - e.g. “*25-30 degrees*” generalizes to “*hot*” (both are strings)
 - spatial-to-nonspatial
 - e.g. *Seattle* generalizes to description “*Pacific Northwest*” (as a string)
 - spatial-to-spatial
 - e.g. *Seattle* generalizes to *Pacific Northwest* (as a spatial region)
- Measures
 - numerical (e.g. monthly revenue of a region)
 - distributive (e.g. count, sum)
 - algebraic (e.g. average)
 - holistic (e.g. median, rank)
- spatial
 - collection of spatial pointers (e.g. pointers to all regions with temperature of 25-30 degrees in July)

Spatial Association Analysis

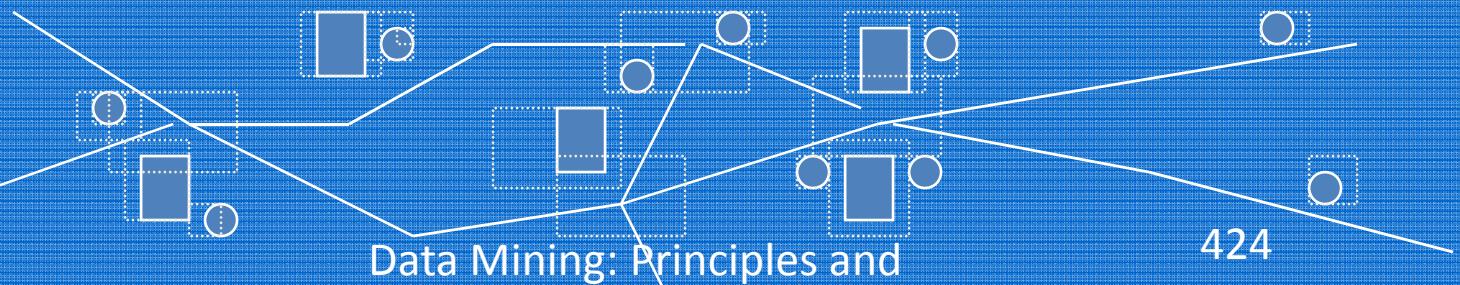
- Spatial association rule: $A \rightarrow B [s\%, c\%]$
 - A and B are sets of spatial or non-spatial predicates
 - Topological relations: *intersects*, *overlaps*, *disjoint*, etc.
 - Spatial orientations: *left_of*, *west_of*, *under*, etc.
 - Distance information: *close_to*, *within_distance*, etc.
 - $s\%$ is the support and $c\%$ is the confidence of the rule
- Examples

1) $is_a(x, large_town) \wedge intersect(x, highway) \rightarrow adjacent_to(x, water)$
[7%, 85%]

2) What kinds of objects are typically located close to golf courses?

Progressive Refinement Mining of Spatial Association Rules

- Hierarchy of spatial relationship:
 - g_close_to : *near_by*, *touch*, *intersect*, *contain*, etc.
 - First search for rough relationship and then refine it
- Two-step mining of spatial association:
 - Step 1: Rough spatial computation (as a filter)
 - Using MBR or R-tree for rough estimation
 - Step2: Detailed spatial algorithm (as refinement)
 - Apply only to those objects which have passed the rough spatial association test (no less than *min_support*)



Spatial Autocorrelation

- Spatial data tends to be highly self-correlated
 - Example: Neighborhood, Temperature
 - Items in a traditional data are independent of each other, whereas properties of locations in a map are often “**auto-correlated**”.
- First law of geography:

“Everything is related to everything, but nearby things are more related than distant things.”

Spatial Classification

- Methods in classification
 - Decision-tree classification, Naïve-Bayesian classifier + boosting, neural network, logistic regression, etc.
 - Association-based multi-dimensional classification -
Example: classifying house value based on proximity to lakes, highways, mountains, etc.
- Assuming learning samples are independent of each other
 - Spatial auto-correlation violates this assumption!
- Popular spatial classification methods
 - Spatial auto-regression (SAR)
 - Markov random field (MRF)

Spatial Auto-Regression

- Linear Regression

$$Y = X\beta + \varepsilon$$

- Spatial autoregressive regression (SAR)

$$Y = \rho W Y + X\beta + \varepsilon$$

- W : neighborhood matrix.
- ρ models strength of spatial dependencies
- ε error vector

The estimates of ρ and β can be derived using maximum likelihood theory or Bayesian statistics

Markov Random Field Based Bayesian Classifiers

- Bayesian classifiers

$$Pr(C_i|X) = \frac{Pr(X|C_i)Pr(C_i)}{Pr(X)}$$

- MRF

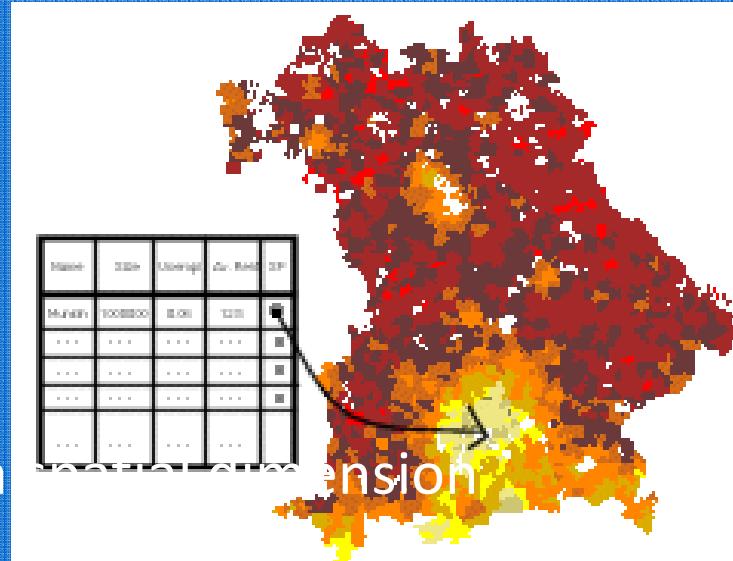
- A set of random variables whose interdependency relationship is represented by an undirected graph (i.e., a symmetric neighborhood matrix) is called a Markov Random Field.

$$Pr(C_i | X, L_i) = \frac{Pr(X | C_i, L_i) Pr(C_i | L_i)}{Pr(X)}$$

- L_i denotes set of labels in the neighborhood of s_i , excluding labels at s_i
 - $Pr(C_i | L_i)$ can be estimated from training data by examine the ratios of the frequencies of class labels to the total number of locations
 - $Pr(X | C_i, L_i)$ can be estimated using kernel functions from the observed values in the training dataset

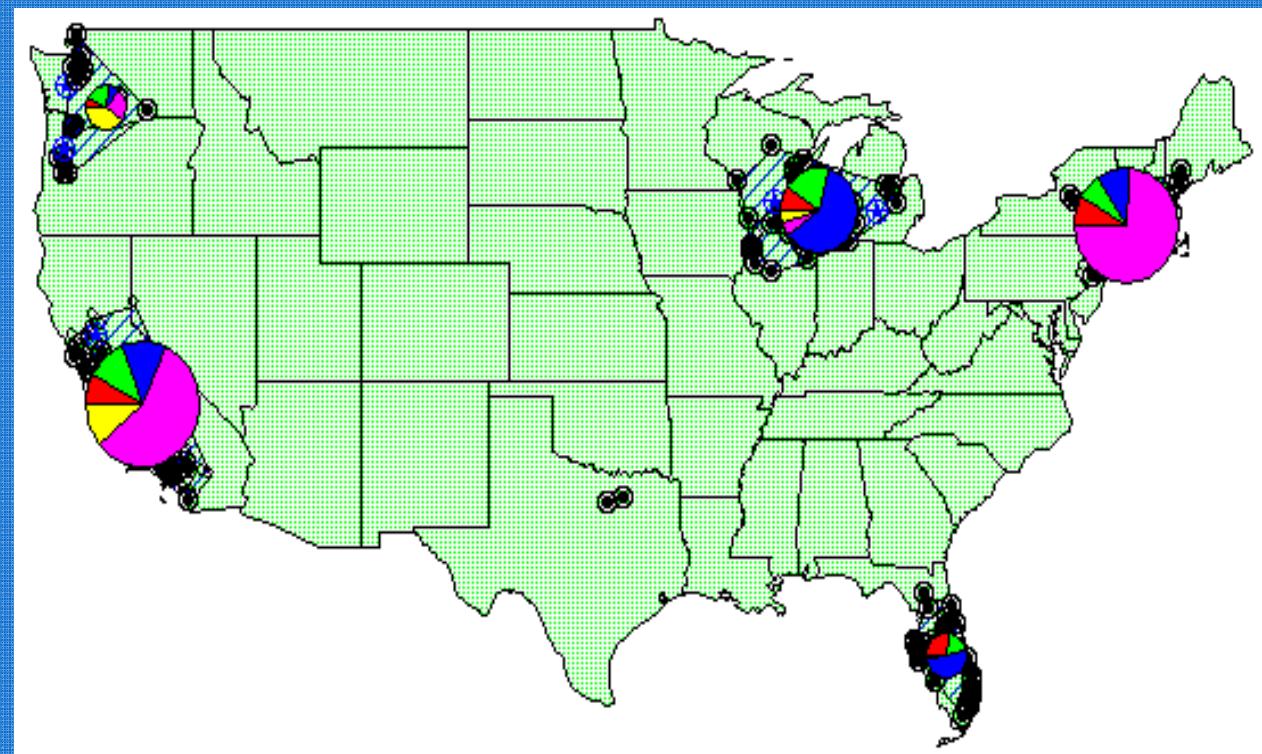
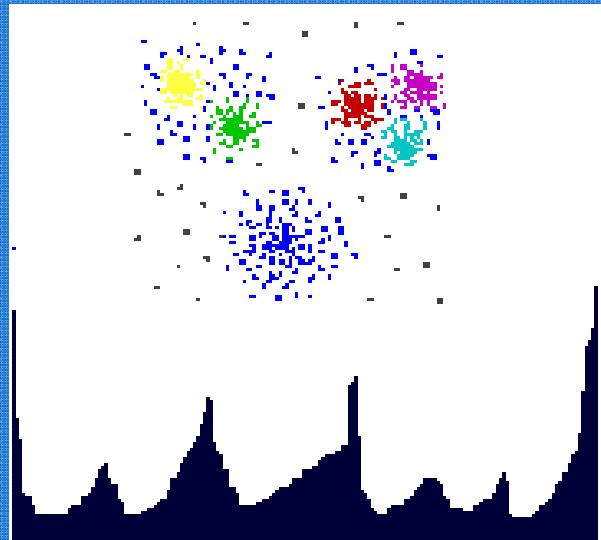
Spatial Trend Analysis

- Function
 - Detect changes and trends along a dimension
 - Study the trend of non-spatial or spatial data changing with space
- Application examples
 - Observe the trend of changes of the climate or vegetation with increasing distance from an ocean
 - Crime rate or unemployment rate change with regard to city geo-distribution



Spatial Cluster Analysis

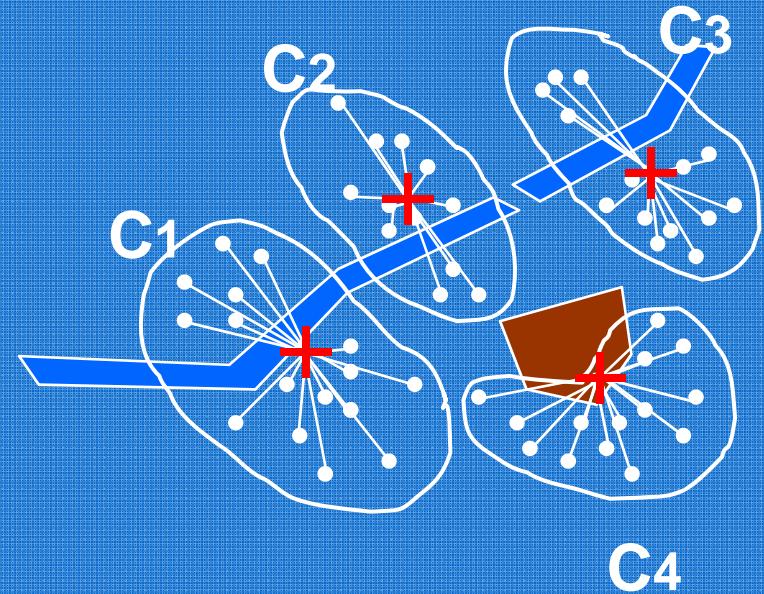
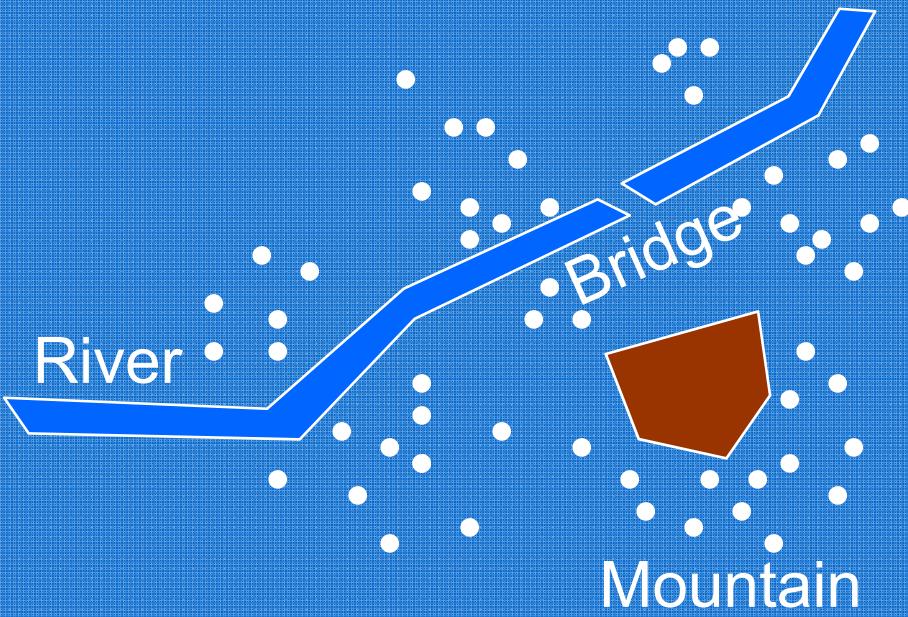
- Mining clusters—k-means, k-medoids, hierarchical, density-based, etc.
- Analysis of distinct features of the clusters



Constraints-Based Clustering

- Constraints on individual objects
 - Simple selection of relevant objects before clustering
- Clustering parameters as constraints
 - K-means, density-based: radius, min-# of points
- Constraints specified on clusters using SQL aggregates
 - Sum of the profits in each cluster $> \$1$ million
- Constraints imposed by physical obstacles
 - Clustering with obstructed distance

Constrained Clustering: Planning ATM Locations



Spatial data with obstacles

12/26/2012

Data Mining: Principles and
Algorithms

Clustering *without* taking
obstacles into consideration

432

Spatial Outlier Detection

- Outlier
 - Global outliers: Observations which is inconsistent with the rest of the data
 - Spatial outliers: A local instability of non-spatial attributes
- Spatial outlier detection
 - Graphical tests
 - Variogram clouds
 - Moran scatterplots
 - Quantitative tests
 - Scatterplots
 - Spatial Statistic $Z(S(x))$
 - Quantitative tests are more accurate than Graphical tests

Mining Object, Spatial and Multi-Media Data

- Mining object data sets
- Mining spatial databases and data warehouses
 - Spatial DBMS
 - Spatial Data Warehousing
 - Spatial Data Mining
 - Spatiotemporal Data Mining
- Mining multimedia data
- Summary



Similarity Search in Multimedia Data

- Description-based retrieval systems
 - Build indices and perform object retrieval based on image descriptions, such as keywords, captions, size, and time of creation
 - Labor-intensive if performed manually
 - Results are typically of poor quality if automated
- Content-based retrieval systems
 - Support retrieval based on the image content, such as color histogram, texture, shape, objects, and wavelet transforms

- Image sample-based queries
 - Find all of the images that are similar to the given image sample
 - Compare the feature vector (signature) extracted from the sample with the feature vectors of images that have already been extracted and indexed in the image database
- Image feature specification queries
 - Specify or sketch image features like color, texture, or shape, which are translated into a feature vector
 - Match the feature vector with the feature vectors of the images in the database

Approaches Based on Image Signature

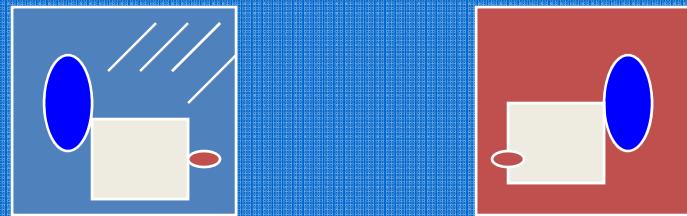
- Color histogram-based signature
 - The signature includes color histograms based on color composition of an image regardless of its scale or orientation
 - No information about shape, location, or texture
 - Two images with similar color composition may contain very different shapes or textures, and thus could be completely unrelated in semantics
- Multifeature composed signature
 - Define different distance functions for color, shape, location, and texture, and subsequently combine them to derive the overall result

Wavelet Analysis

- Wavelet-based signature
 - Use the dominant wavelet coefficients of an image as its signature
 - Wavelets capture shape, texture, and location information in a single unified framework
 - Improved efficiency and reduced the need for providing multiple search primitives
 - May fail to identify images containing similar objects that are in different locations.

One Signature for the Entire Image?

- Walrus: [NRS99] by Natsev, Rastogi, and Shim
- Similar images may contain similar regions, but a region in one image could be a translation or scaling of a matching region in the other

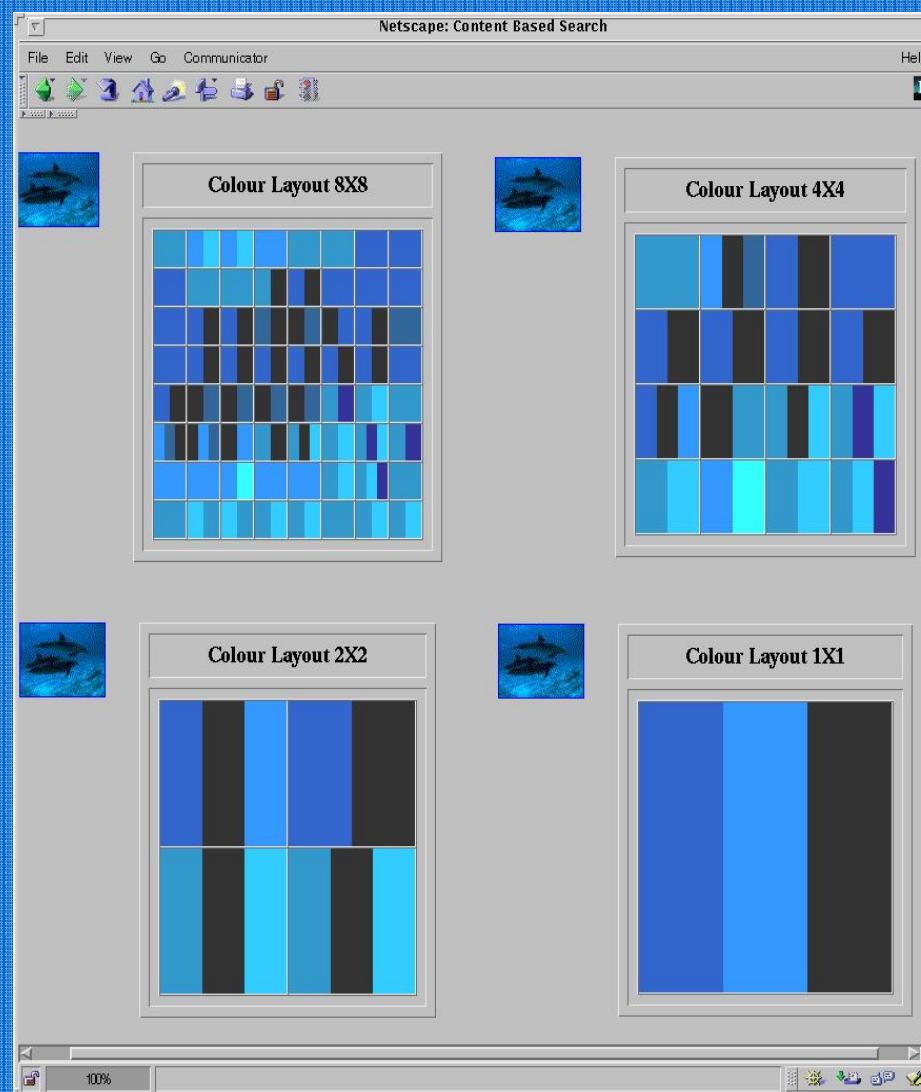
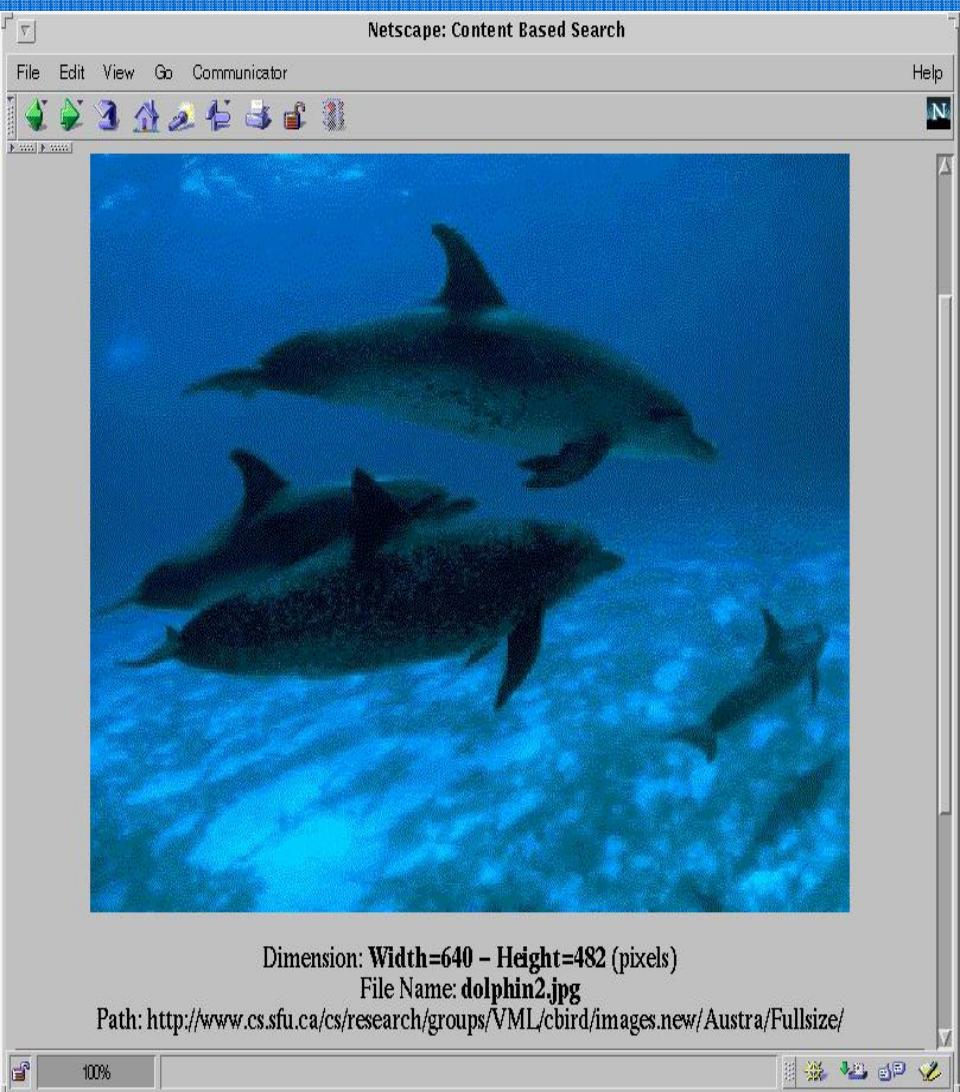


- Wavelet-based signature with region-based granularity
 - Define regions by clustering signatures of windows of varying sizes within the image
 - Signature of a region is the centroid of the cluster
 - Similarity is defined in terms of the fraction of the area of the two images covered by matching pairs of regions from two images

Multidimensional Analysis of Multimedia Data

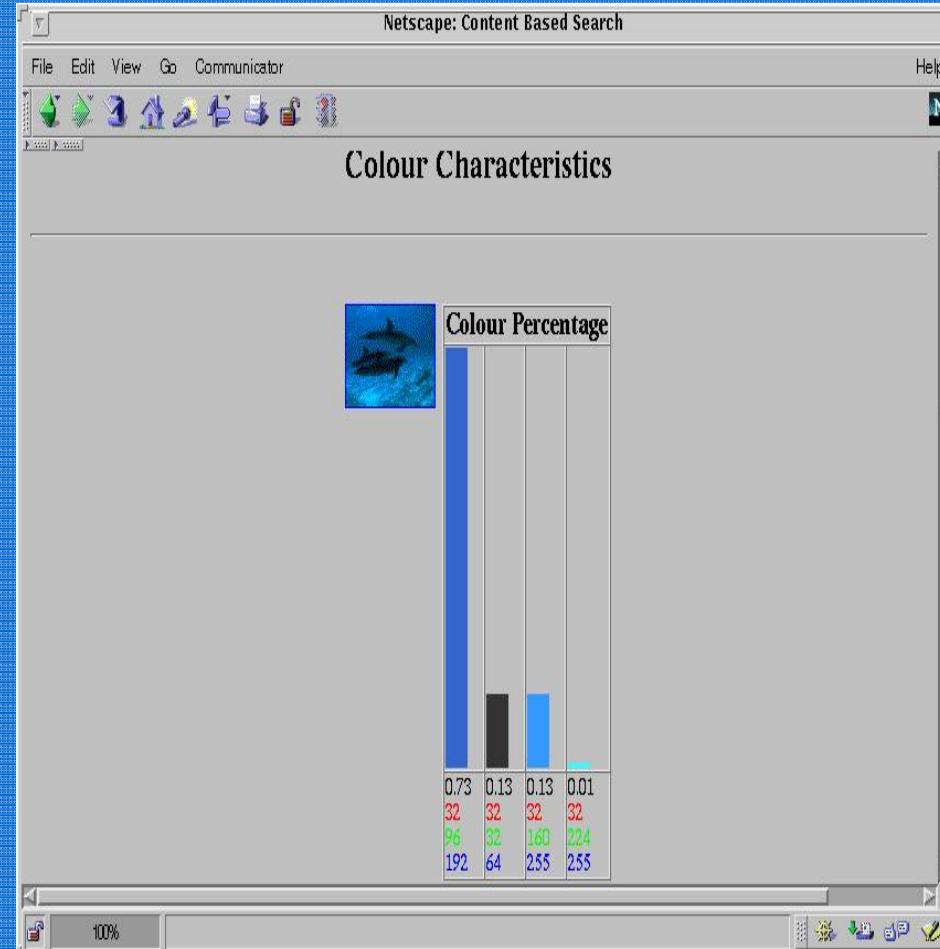
- Multimedia data cube
 - Design and construction similar to that of traditional data cubes from relational data
 - Contain additional dimensions and measures for multimedia information, such as color, texture, and shape
- The database does not store images but their descriptors
 - **Feature descriptor:** a set of vectors for each visual characteristic
 - Color vector: contains the color histogram
 - MFC (Most Frequent Color) vector: five color centroids
 - MFO (Most Frequent Orientation) vector: five edge orientation centroids
 - **Layout descriptor:** contains a color layout vector and an edge layout vector

Multi-Dimensional Search in Multimedia Databases

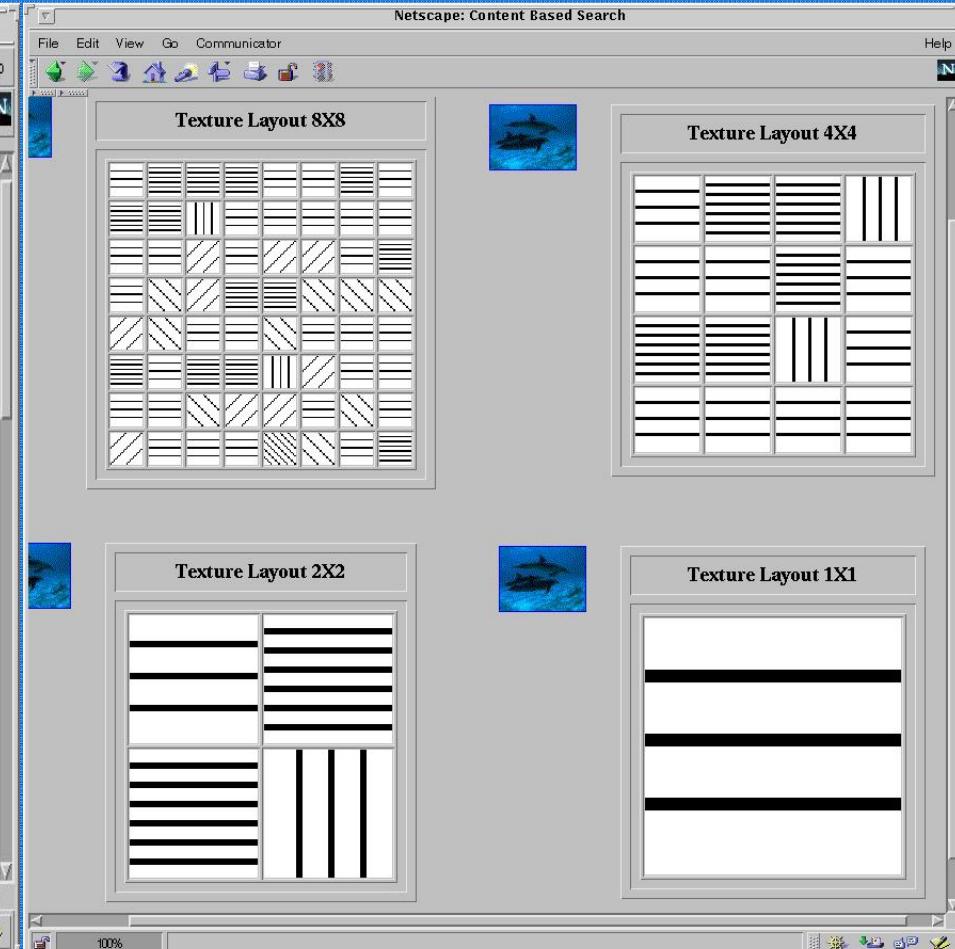


Multi-Dimensional Analysis in Multimedia Databases

Color histogram



Texture layout



Mining Multimedia Databases

Refining or combining searches



Search for “blue sky”
(top layout grid is blue)



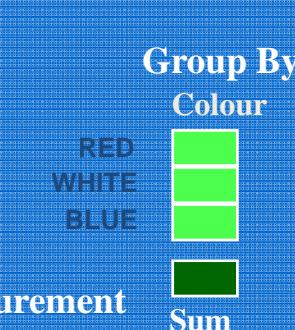
Search for “airplane in blue sky”
(top layout grid is blue and
keyword = “airplane”)



Search for “blue sky and
green meadows”
(top layout grid is blue
and bottom is green)

Mining Multimedia Databases

Two Dimensions



Cross Tab

	JPEG	GIF	By Colour
RED			
WHITE			
BLUE			
By Format			
Sum			

Three Dimensions

By Format & Size

By Colour & Size

By Size

JPEG

GIF

Small

Medium

Large

Very Large

By Format

RED
WHITE
BLUE

By Format & Colour
By Colour

- Format of image
- Duration
- Colors
- Textures
- Keywords
- Size
- Width
- Height
- Internet domain of image
- Internet domain of parent pages
- Image popularity

Dimensions

Mining Multimedia Databases in **MultiMediaMiner**



Classification in MultiMediaMiner

MultiMediaMiner

File Edit Query View Window Options Help

Dim: Keyword Level: Level0 Class% 85 Noise% 1.00

jupiter.cs.sfu.ca

Animal



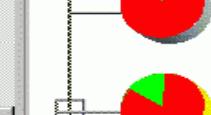
All



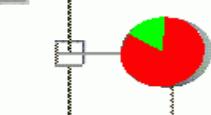
Book



Building



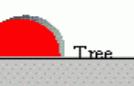
Airplane



Animal



Plant



Flower



Tree

Book



Media Format

- MOV
- AVI
- MPG
- GIF
- JPEG or JPG

Flower



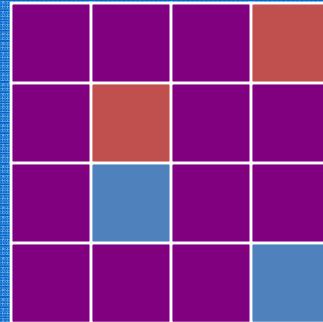
Mining Associations in Multimedia Data

- Special features:
 - Need # of occurrences besides Boolean existence, e.g.,
 - “Two red square and one blue circle” implies theme “air-show”
 - Need spatial relationships
 - Blue on top of white squared object is associated with brown bottom
 - Need multi-resolution and progressive refinement mining
 - It is expensive to explore detailed associations among objects at high resolution
 - It is crucial to ensure the completeness of search at multi-resolution space

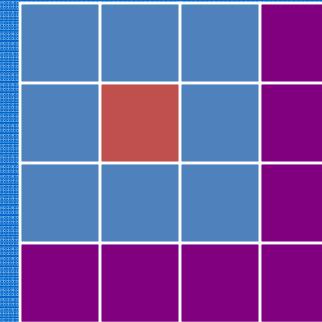
Mining Multimedia Databases

Spatial Relationships from Layout

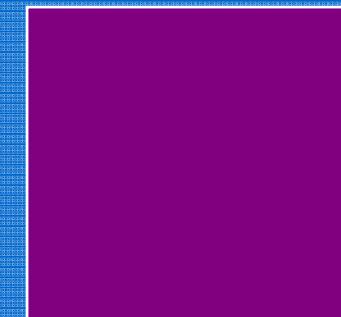
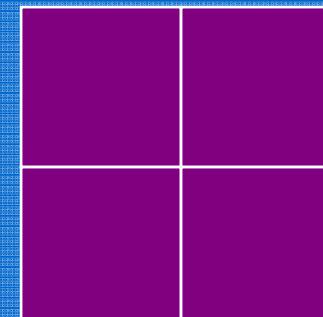
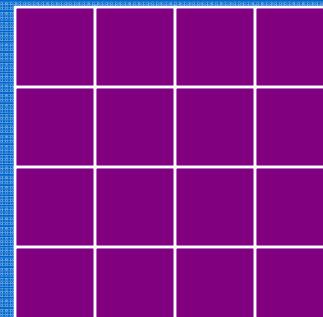
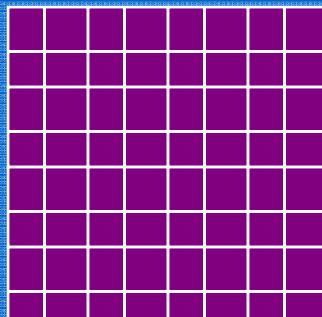
property P1 *on-top-of* property P2



property P1 *next-to* property P2

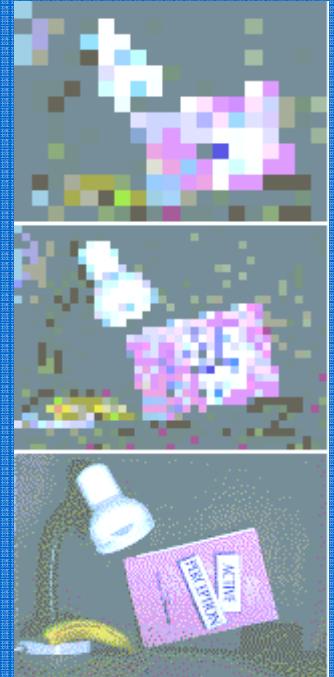


Different Resolution Hierarchy



Mining Multimedia Databases

From Coarse to Fine Resolution Mining



Challenge: Curse of Dimensionality

- Difficult to implement a data cube efficiently given a large number of dimensions, especially serious in the case of multimedia data cubes
- Many of these attributes are set-oriented instead of single-valued
- Restricting number of dimensions may lead to the modeling of an image at a rather rough, limited, and imprecise scale
- More research is needed to strike a balance between efficiency and power of representation

Summary

- Mining object data needs feature/attribute-based generalization methods
- Spatial, spatiotemporal and multimedia data mining is one of important research frontiers in data mining with broad applications
- **Spatial data warehousing, OLAP and mining** facilitates multidimensional spatial analysis and finding spatial associations, classifications and trends
- **Multimedia data mining** needs **content-based retrieval** and **similarity search** integrated with mining methods

Mining Text and Web Data

Mining Text and Web Data

- Text mining, natural language processing and information extraction: An Introduction
- Text categorization methods
- Mining Web linkage structures
- Summary



Mining Text Data: An Introduction

Data Mining / Knowledge Discovery



Structured Data

```
HomeLoan (
  Loanee: Frank Rizzo
  Lender: MWF
  Agency: Lake View
  Amount: $200,000
  Term: 15 years
)
```

Multimedia



Free Text

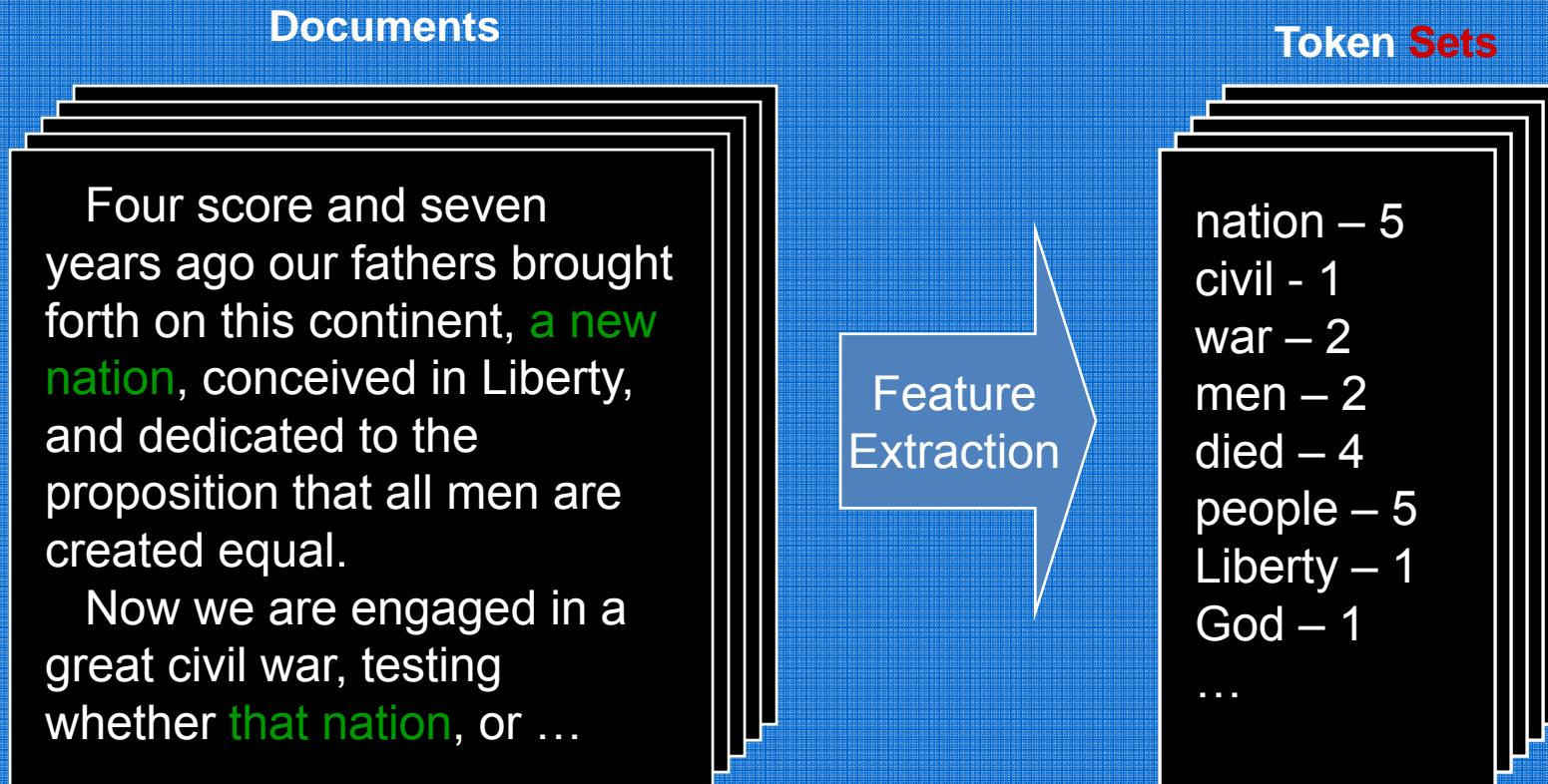
Frank Rizzo bought his home from Lake View Real Estate in 1992. He paid \$200,000 under a 15-year loan from MW Financial.

Hypertext

[Frank Rizzo](#) Bought
[this home](#) from [Lake View Real Estate](#) In **1992**.

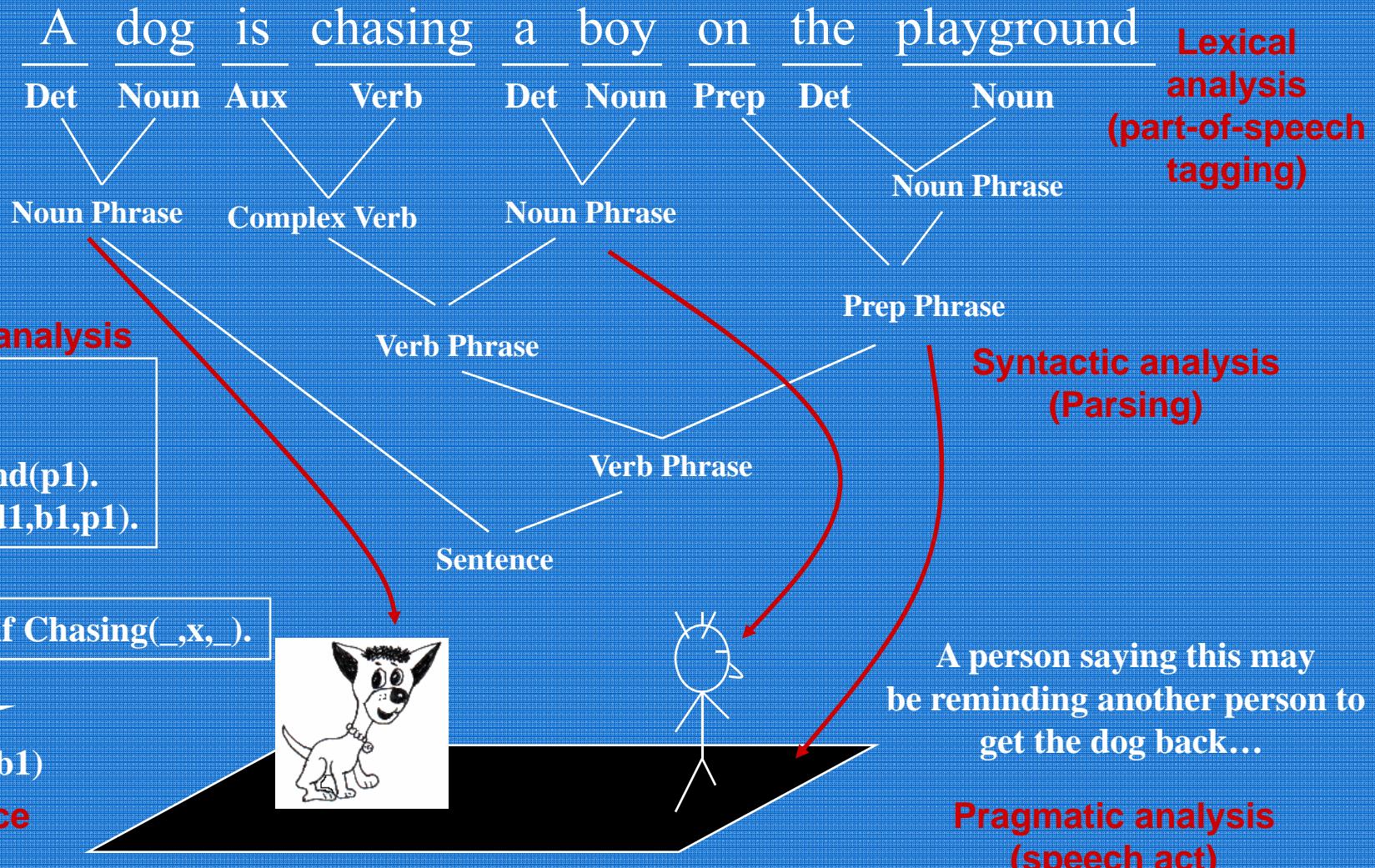
...

Bag-of-Tokens Approaches



**Loses all order-specific information!
Severely limits context!**

Natural Language Processing



General NLP—Too Difficult!

- Word-level ambiguity
 - “design” can be a noun or a verb (Ambiguous POS)
 - “root” has multiple meanings (Ambiguous sense)
- Syntactic ambiguity
 - “natural language processing” (Modification)
 - “A man saw a boy with a telescope.” (PP Attachment)
- Anaphora resolution
 - “John persuaded Bill to buy a TV for himself.”
(himself = John or Bill?)
- Presupposition
 - “He has quit smoking.” implies that he smoked before.

**Humans rely on context to interpret (when possible).
This context may extend beyond a given document!**

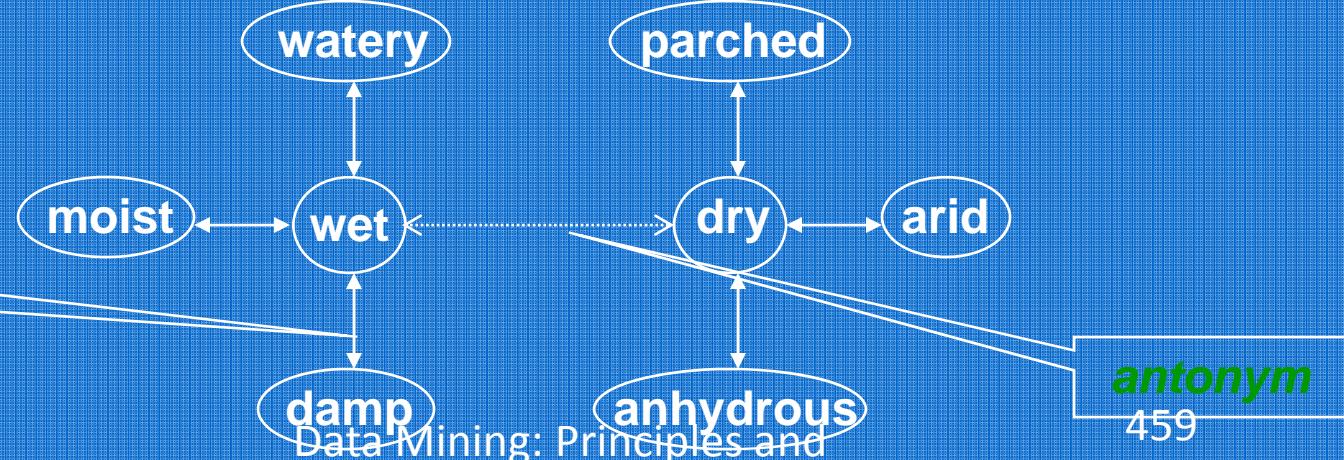
Progress on **Useful Sub-Goals**:

- English Lexicon
- Part-of-Speech Tagging
- Word Sense Disambiguation
- Phrase Detection / Parsing

WordNet

An extensive **lexical network** for the English language

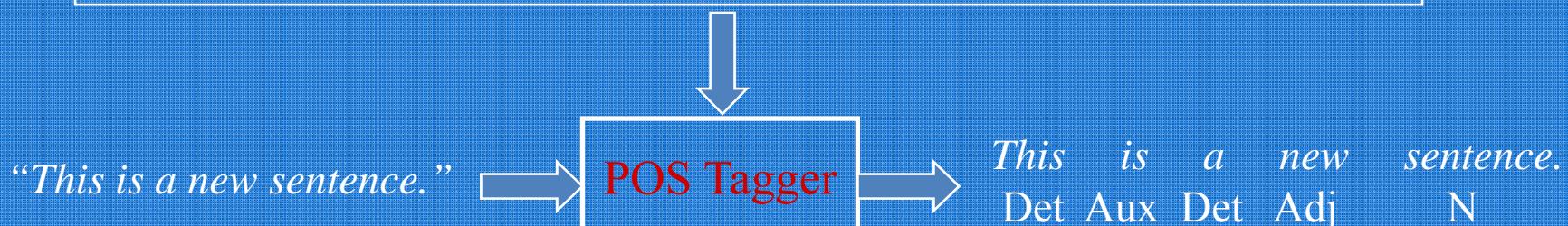
- Contains over **138,838 words**.
- Several graphs, one for each **part-of-speech**.
- **Synsets** (synonym sets), each defining a semantic sense.
- **Relationship** information (antonym, hyponym, meronym ...)
- Downloadable for **free** (UNIX, Windows)
- Expanding to **other languages** (Global WordNet Association)
- Funded **>\$3 million**, mainly government (translation interest)
- Founder **George Miller, National Medal of Science, 1991**.



Part-of-Speech Tagging

Training data (Annotated text)

<i>This</i>	<i>sentence</i>	<i>serves</i>	<i>as</i>	<i>an</i>	<i>example</i>	<i>of</i>	<i>annotated</i>	<i>text...</i>
Det	N	V1	P	Det	N	P	V2	N



Pick the most likely tag sequence.

$$\left(p(t_1 | w_1) \dots p(t_k | w_k) p(w_1) \dots p(w_k) \right)$$

$$p(t_i | t_{i-1})$$

Independent assignment
Most common tag

Partial dependency
(HMM)

Word Sense Disambiguation

?

"The difficulties of computational linguistics are rooted in ambiguity."

N Aux V P N

Supervised Learning

Features:

- Neighboring POS tags (N Aux V P N)
- Neighboring words (*linguistics are rooted in ambiguity*)
- Stemmed form (*root*)
- Dictionary/Thesaurus entries of neighboring words
- High co-occurrence words (*plant, tree, origin, ...*)
- Other *senses* of word within discourse

Algorithms:

- Rule-based Learning (e.g. IG guided)
- Statistical Learning (*i.e.* Naïve Bayes)
- Unsupervised Learning (*i.e.* Nearest Neighbor)

Parsing

Choose **most likely** parse tree...

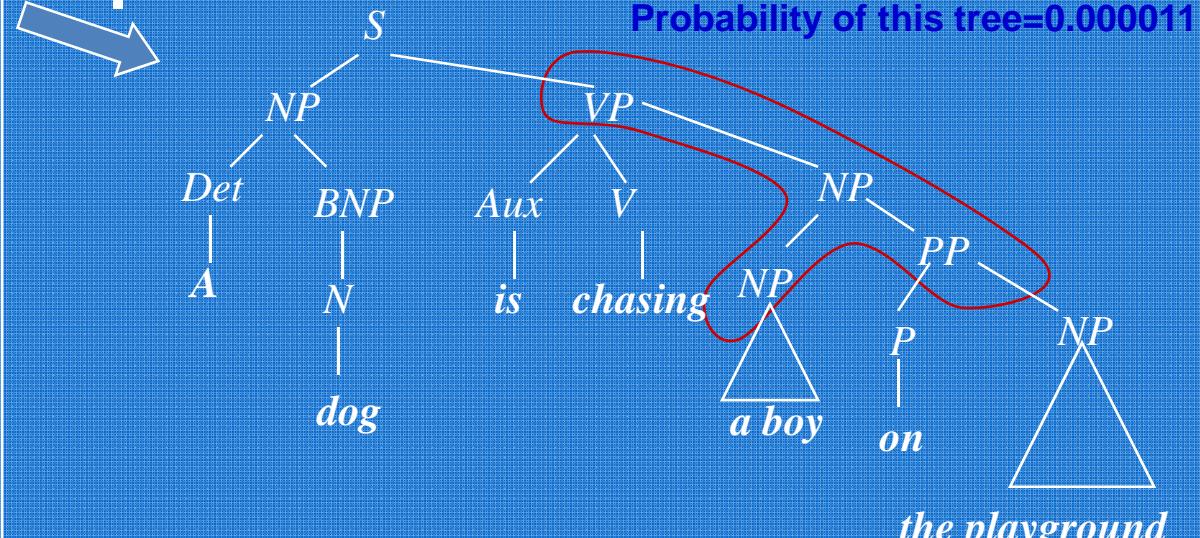
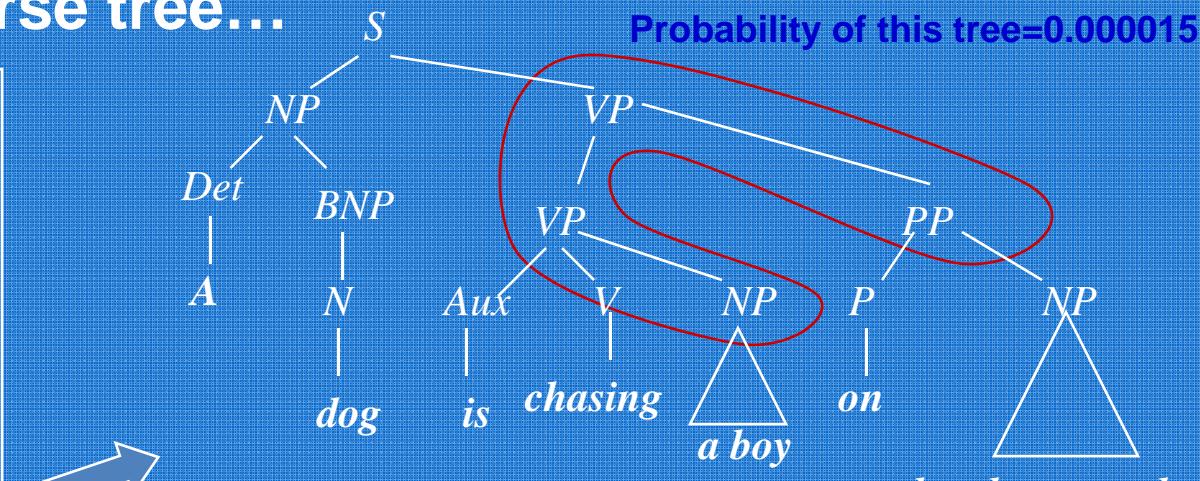
Probabilistic CFG

Grammar

$S \rightarrow NP\ VP$	1.0
$NP \rightarrow Det\ BNP$	0.3
$NP \rightarrow BNP$	0.4
$NP \rightarrow NP\ PP$	0.3
$BNP \rightarrow N$	
$VP \rightarrow V$	
$VP \rightarrow Aux\ V\ NP$	
$VP \rightarrow VP\ PP$	
$PP \rightarrow P\ NP$	1.0

Lexicon

$V \rightarrow chasing$	0.01
$Aux \rightarrow is$	
$N \rightarrow dog$	0.003
$N \rightarrow boy$	
$N \rightarrow playground$	
$Det \rightarrow the$	
$Det \rightarrow a$	
$P \rightarrow on$	



Obstacles

- **Ambiguity**
“A man saw a boy with a telescope.”
- **Computational Intensity**
Imposes a context horizon.

Text Mining NLP Approach:

1. Locate promising fragments using **fast IR methods** (bag-of-tokens).
2. Only apply **slow NLP techniques** to promising fragments.

Summary: Shallow NLP

However, shallow NLP techniques are feasible and useful:

- Lexicon – machine understandable linguistic knowledge
 - possible senses, definitions, synonyms, antonyms, typeof, etc.
- POS Tagging – limit ambiguity (word/POS), entity extraction
 - “...research interests include **text mining** as well as **bioinformatics**.”

NP

N

- WSD – stem/synonym/hyponym matches (doc and query)
 - Query: “*Foreign cars*” Document: “*I’m selling a 1976 Jaguar...*”
- Parsing – logical view of information (inference?, translation?)
 - “*A man saw a boy with a telescope.*”

Even without complete NLP, any additional knowledge extracted from text data can only be beneficial.

Ingenuity will determine the applications.

Mining Text and Web Data

- Text mining, natural language processing and information extraction: An Introduction
- Text information system and information retrieval
- Text categorization methods
- Mining Web linkage structures
- Summary



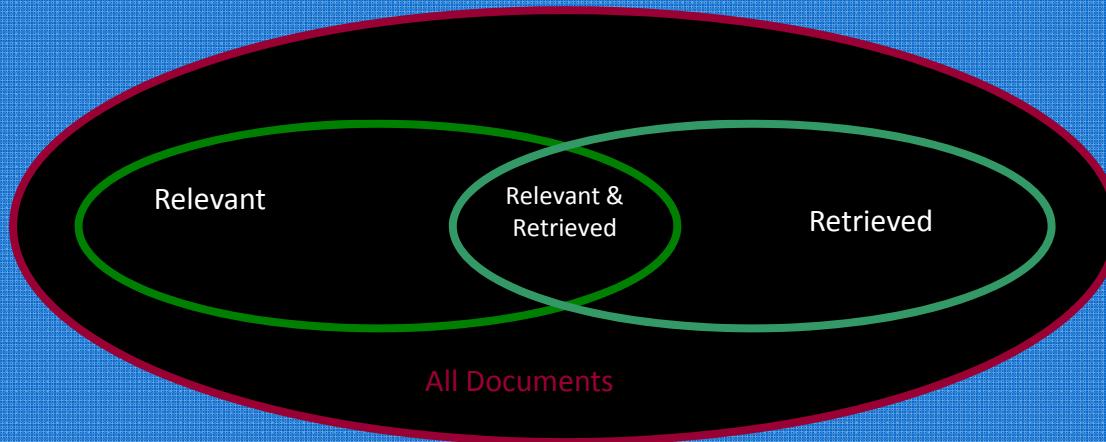
Text Databases and IR

- Text databases (document databases)
 - Large collections of documents from various sources: news articles, research papers, books, digital libraries, e-mail messages, and Web pages, library database, etc.
 - Data stored is usually *semi-structured*
 - Traditional information retrieval techniques become inadequate for the increasingly vast amounts of text data
- Information retrieval
 - A field developed in parallel with database systems
 - Information is organized into (a large number of) documents
 - Information retrieval problem: locating relevant documents based on user input, such as keywords or example documents

Information Retrieval

- Typical IR systems
 - Online library catalogs
 - Online document management systems
- Information retrieval vs. database systems
 - Some DB problems are not present in IR, e.g., update, transaction management, complex objects
 - Some IR problems are not addressed well in DBMS, e.g., unstructured documents, approximate search using keywords and relevance

Basic Measures for Text Retrieval



- **Precision:** the percentage of retrieved documents that are in fact relevant to the query (i.e., “correct” responses)

$$precision = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Retrieved\}|}$$

- **Recall:** the percentage of documents that are relevant to the query and were, in fact, retrieved

$$precision = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

Information Retrieval Techniques

- Basic Concepts
 - A document can be described by a set of representative keywords called **index terms**.
 - Different index terms have varying relevance when used to describe document contents.
 - This effect is captured through the **assignment of numerical weights to each index term** of a document. (e.g.: frequency, tf-idf)
- DBMS Analogy
 - Index Terms → **Attributes**
 - Weights → **Attribute Values**

Information Retrieval Techniques

- Index Terms (Attribute) Selection:
 - Stop list
 - Word stem
 - Index terms weighting methods
- Terms \times Documents Frequency Matrices
- Information Retrieval Models:
 - Boolean Model
 - Vector Model
 - Probabilistic Model

Boolean Model

- Consider that index terms are either present or absent in a document
- As a result, the index term weights are assumed to be all binaries
- A query is composed of index terms linked by three connectives: **not**, **and**, and **or**
 - e.g.: car **and** repair, plane **or** airplane
- The Boolean model predicts that each document is either relevant or non-relevant based on the match of a document to the query

Keyword-Based Retrieval

- A document is represented by a string, which can be identified by a set of keywords
- Queries may use **expressions** of keywords
 - E.g., car *and* repair shop, tea *or* coffee, DBMS *but not* Oracle
 - Queries and retrieval should consider **synonyms**, e.g., repair and maintenance
- Major difficulties of the model
 - **Synonymy:** A keyword T does not appear anywhere in the document, even though the document is closely related to T , e.g., data mining
 - **Polysemy:** The same keyword may mean different things in different contexts, e.g., mining

Similarity-Based Retrieval in Text Data

- Finds similar documents based on a set of common keywords
- Answer should be based on the degree of relevance based on the nearness of the keywords, relative frequency of the keywords, etc.
- Basic techniques
- Stop list
 - Set of words that are deemed “irrelevant”, even though they may appear frequently
 - E.g., *a, the, of, for, to, with*, etc.
 - Stop lists may vary when document set varies

Similarity-Based Retrieval in Text Data

- Word stem
 - Several words are small syntactic variants of each other since they share a common word stem
 - E.g., *drug, drugs, drugged*
- A term frequency table
 - Each entry $frequent_table(i, j) = \# \text{ of occurrences of the word } t_i \text{ in document } d_j$
 - Usually, the *ratio* instead of the absolute number of occurrences is used
- Similarity metrics: measure the closeness of a document to a query (a set of keywords)
 - Relative term occurrences
 - Cosine distance:

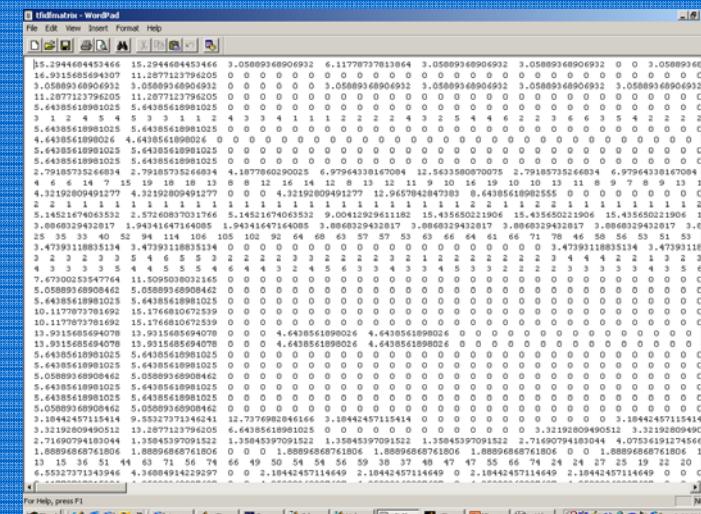
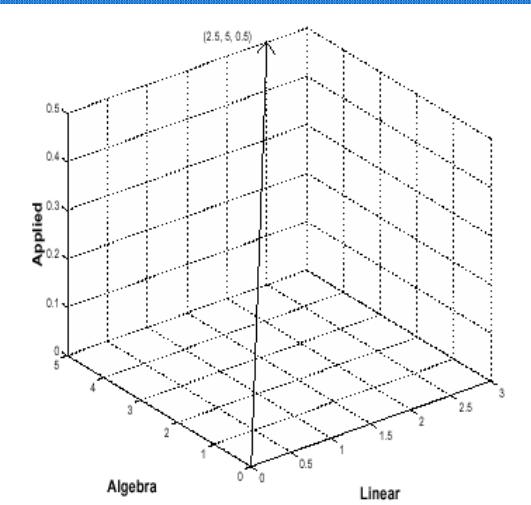
$$sim(v_1, v_2) = \frac{v_1 \cdot v_2}{\|v_1\| \|v_2\|}$$

Indexing Techniques

- Inverted index
 - Maintains two hash- or B+-tree indexed tables:
 - `document_table`: a set of document records $\langle \text{doc_id}, \text{postings_list} \rangle$
 - `term_table`: a set of term records, $\langle \text{term}, \text{postings_list} \rangle$
 - Answer query: Find all docs associated with one or a set of terms
 - + easy to implement
 - – do not handle well synonymy and polysemy, and posting lists could be too long (storage could be very large)
- Signature file
 - Associate a signature with each document
 - A signature is a representation of an ordered list of terms that describe the document
 - Order is obtained by frequency analysis, stemming and stop lists

Vector Space Model

- Documents and user queries are represented as m-dimensional vectors, where m is the total number of index terms in the document collection.
- The degree of similarity of the document d with regard to the query q is calculated as the correlation between the vectors that represent them, using measures such as the Euclidian distance or the cosine of the angle between these two vectors.



Probabilistic Model

- Basic assumption: Given a user query, there is a set of documents which contains exactly the relevant documents and no other (ideal answer set)
- Querying process as a process of specifying the properties of an ideal answer set. Since these properties are not known at query time, an initial guess is made
- This initial guess allows the generation of a preliminary probabilistic description of the ideal answer set which is used to retrieve the first set of documents
- An interaction with the user is then initiated with the purpose of improving the probabilistic description of the answer set

Types of Text Data Mining

- Keyword-based association analysis
- Automatic document classification
- Similarity detection
 - Cluster documents by a common author
 - Cluster documents containing information from a common source
- Link analysis: unusual correlation between entities
- Sequence analysis: predicting a recurring event
- Anomaly detection: find information that violates usual patterns
- Hypertext analysis
 - Patterns in anchors/links
 - Anchor text correlations with linked objects

Keyword-Based Association Analysis

- Motivation
 - Collect sets of keywords or terms that occur frequently together and then find the **association** or **correlation** relationships among them
- Association Analysis Process
 - Preprocess the text data by parsing, stemming, removing stop words, etc.
 - Evoke association mining algorithms
 - Consider each document as a transaction
 - View a set of keywords in the document as a set of items in the transaction
 - Term level association mining
 - No need for human effort in tagging documents
 - The number of meaningless results and the execution time is greatly reduced

Text Classification

- Motivation
 - Automatic classification for the large number of on-line text documents (Web pages, e-mails, corporate intranets, etc.)
- Classification Process
 - Data preprocessing
 - Definition of training set and test sets
 - Creation of the classification model using the selected classification algorithm
 - Classification model validation
 - Classification of new/unknown text documents
- Text document classification differs from the classification of relational data
 - Document databases are not structured according to attribute-value pairs

Text Classification(2)

- Classification Algorithms:
 - Support Vector Machines
 - K-Nearest Neighbors
 - Naïve Bayes
 - Neural Networks
 - Decision Trees
 - Association rule-based
 - Boosting

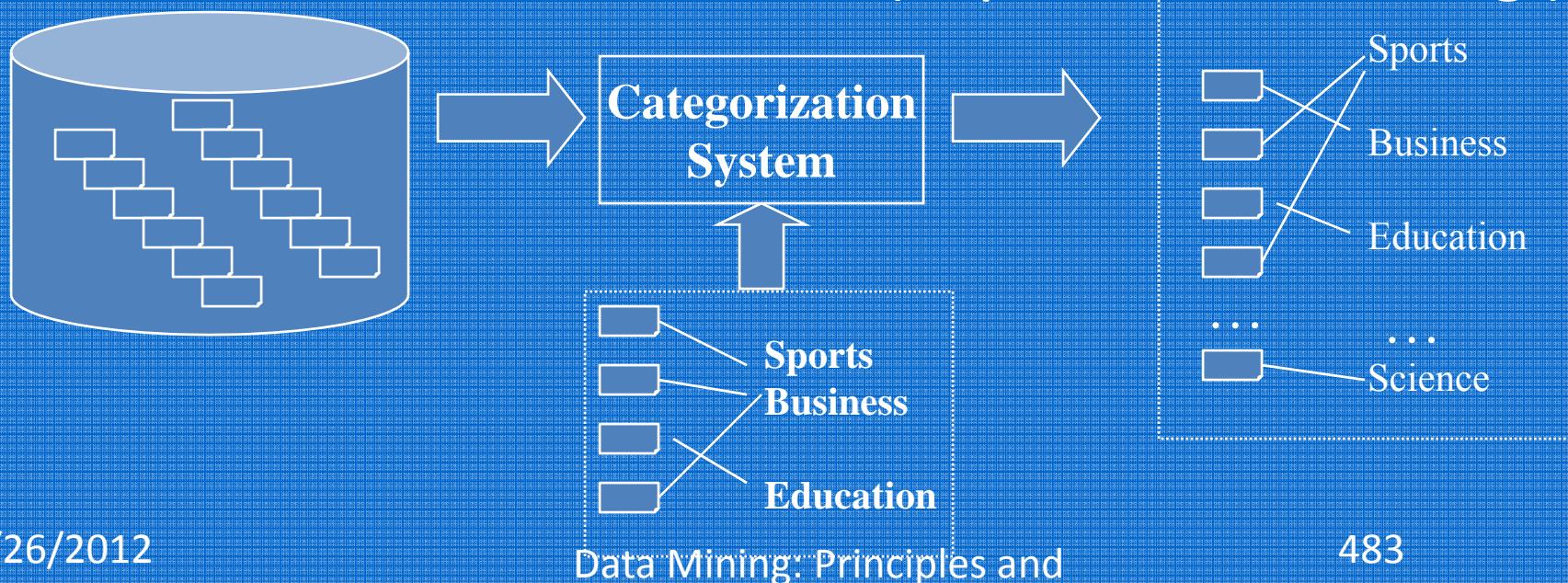
			#1	#2	#3	#4	#5
		# of documents # of training documents # of test documents # of categories	21,450 14,704 6,746 135	14,347 10,667 3,680 93	13,272 9,610 3,662 92	12,902 9,603 3,299 90	12,902 9,603 3,299 10
System	Type	Results reported by					
WORD	(non-learning)	[Yang 1999]	.150	.310	.290	.752	.815 .720
PROPBAYES	probabilistic	[Dumais et al. 1998]					
BIM	probabilistic	[Joachims 1998]					
Nb	probabilistic	[Lam et al. 1997]					
Lewis	probabilistic	[Lewis 1992a]					
C4.5	decision tree	[Li and Yamanishi 1999]					
IND	decision tree	[Li and Yamanishi 1999]					
SWAP-1	decision rules	[Yang and Liu 1999]					
RIPPER	decision rules	[Lewis and Ringouette 1994]	.670				
SLEEPING EXPERTS	decision rules	[Apté et al. 1994]					
DL-ESO	decision rules	[Cohen and Singer 1999]					
CHARADE	decision rules	[Cohen and Singer 1999]					
CHARADE	decision rules	[Li and Yamanishi 1999]					
LLSF	regression	[Moulinier and Gascoin 1996]					
LLSF	regression	[Moulinier et al. 1996]					
BALANCEDWINNOW	on-line linear	[Yang 1999]					
WIDROW-HOFF	on-line linear	[Yang and Liu 1999]					
ROCCIO	batch linear	[Dagan et al. 1997]					
FINDSIM	batch linear	[Lam and Ho 1998]					
ROCCIO	batch linear	[Cohen and Singer 1999]					
ROCCIO	batch linear	[Dumais et al. 1998]					
ROCCIO	batch linear	[Joachims 1998]					
ROCCIO	batch linear	[Lam and Ho 1998]					
CLASSI	neural network	[Li and Yamanishi 1999]					
NNET	neural network	[Ng et al. 1997]					
NNET	neural network	[Yang and Liu 1999]					
NNET	neural network	[Wiesner et al. 1995]					
Gis-W	example-based	[Lam and Ho 1998]					
k-NN	example-based	[Joachims 1998]					
k-NN	example-based	[Lam and Ho 1998]					
k-NN	example-based	[Yang 1999]					
k-NN	example-based	[Yang and Liu 1999]					
SVMLIGHT	SVM	[Dumais et al. 1998]					
SVMLIGHT	SVM	[Joachims 1998]					
SVMLIGHT	SVM	[Li and Yamanishi 1999]					
SVMLIGHT	SVM	[Yang and Liu 1999]					
ADABOOST.MH	committee committee	[Schapire and Singer 2000]					
		[Weiss et al. 1999]					
	Bayesian not	[Dumais et al. 1998]					
	Bayesian not	[Lam et al. 1997]					
		542 (MF ₁)					

Document Clustering

- Motivation
 - Automatically group related documents based on their contents
 - No predetermined training sets or taxonomies
 - Generate a taxonomy at runtime
- Clustering Process
 - Data preprocessing: remove stop words, stem, feature extraction, lexical analysis, etc.
 - Hierarchical clustering: compute similarities applying clustering algorithms.
 - Model-Based clustering (Neural Network Approach): clusters are represented by “exemplars”. (e.g.: SOM)

Text Categorization

- Pre-given categories and labeled document examples (Categories may form hierarchy)
- Classify new documents
- A standard classification (supervised learning)



Applications

- News article classification
- Automatic email filtering
- Webpage classification
- Word sense disambiguation
-

Categorization Methods

- Manual: Typically rule-based
 - Does not scale up (labor-intensive, rule inconsistency)
 - May be appropriate for special data on a particular domain
- Automatic: Typically exploiting machine learning techniques
 - Vector space model based
 - Prototype-based (Rocchio)
 - K-nearest neighbor (KNN)
 - Decision-tree (learn rules)
 - Neural Networks (learn non-linear classifier)
 - Support Vector Machines (SVM)
 - Probabilistic or generative model based
 - Naïve Bayes classifier

How to Measure Similarity?

- Given two documents

$$D_i = (w_{i1}, w_{i2}, \dots, w_{iN})$$

$$D_j = (w_{j1}, w_{j2}, \dots, w_{jN})$$

- Similarity definition

- cosine

$$Sim(D_i, D_j) = \sum_{t=i}^N w_{it} * w_{jt}$$

- no

$$Sim(D_i, D_j) = \frac{\sum_{t=i}^N w_{it} * w_{jt}}{\sqrt{\sum_{t=1}^N (w_{it})^2 * \sum_{t=1}^N (w_{jt})^2}}$$

Illustrative Example

doc1

text
mining
search
engine
text

$$\text{Sim}(\text{newdoc}, \text{doc1}) = 4.8 * 2.4 + 4.5 * 4.5$$

doc2

travel
text
map
travel

$$\text{Sim}(\text{newdoc}, \text{doc2}) = 2.4 * 2.4$$

To whom is newdoc more similar?

doc3

government
president
congress

	text	mining	travel	map	search	engine	govern	president	congress
IDF(faked)	2.4	4.5	2.8	3.3	2.1	5.4	2.2	3.2	4.3
doc1	2(4.8)	1(4.5)			1(2.1)	1(5.4)			
doc2	1(2.4)		2 (5.6)	1(3.3)					
doc3							1 (2.2)	1(3.2)	1(4.3)
newdoc	1(2.4)	1(4.5)							
.....									

Categorization Methods

- Vector space model
 - K-NN
 - Decision tree
 - Neural network
 - Support vector machine
- Probabilistic model
 - Naïve Bayes classifier
- Many, many others and variants exist [F.S. 02]
 - e.g. Bim, Nb, Ind, Swap-1, LLSF, Widrow-Hoff, Rocchio, Gis-W,

Evaluation (con't)

- Benchmarks
 - Classic: Reuters collection
 - A set of newswire stories classified under categories related to economics.
- Effectiveness
 - Difficulties of strict comparison
 - different parameter setting
 - different “split” (or selection) between training and testing
 - various optimizations
 - However widely recognizable
 - Best: Boosting-based committee classifier & SVM
 - Worst: Naïve Bayes classifier
 - Need to consider other factors, especially efficiency

Summary: Text Categorization

- Wide application domain
- Comparable effectiveness to professionals
 - Manual TC is not 100% and unlikely to improve substantially.
 - A.T.C. is growing at a steady pace
- Prospects and extensions
 - Very noisy text, such as text from O.C.R.
 - Speech transcripts

Research Problems in Text Mining

- Google: what is the next step?
- How to find the pages that match approximately the sophisticated documents, with incorporation of user-profiles or preferences?
- Look back of Google: inverted indices
- Construction of indices for the sophisticated documents, with incorporation of user-profiles or preferences
- Similarity search of such pages using such indices

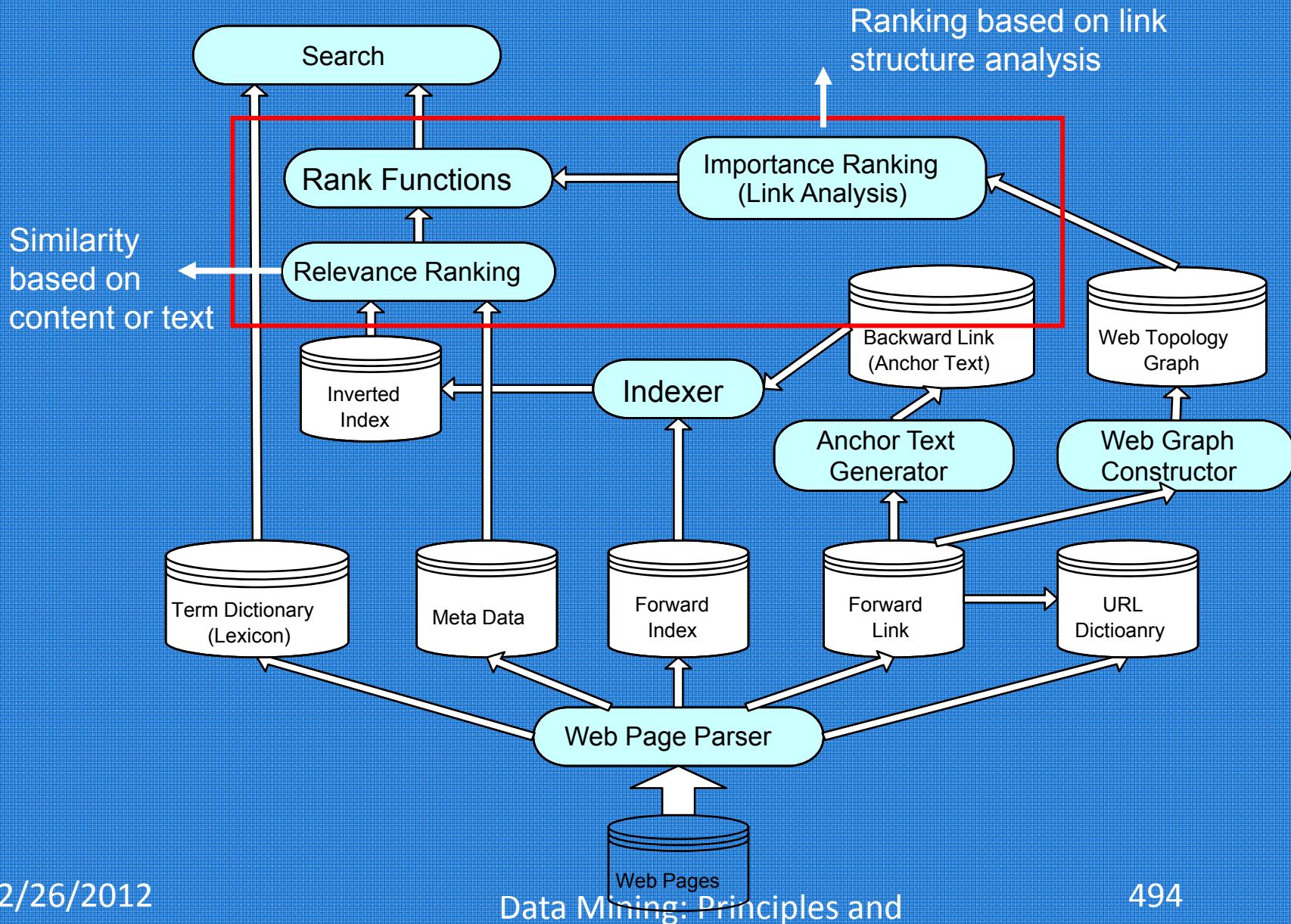
- Text mining, natural language processing and information extraction: An Introduction
- Text categorization methods
- Mining Web linkage structures
 - Based on the slides by Deng Cai
- Summary



Outline

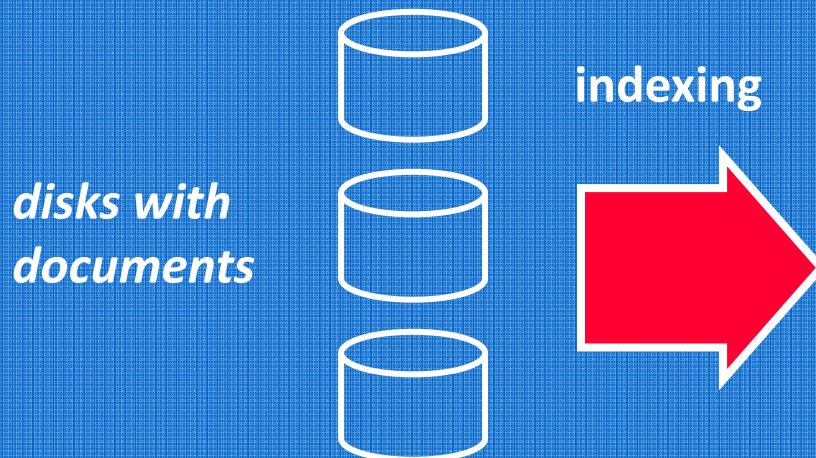
- Background on Web Search
- VIPS (VIision-based Page Segmentation)
- Block-based Web Search
- Block-based Link Analysis
- Web Image Search & Clustering

Search Engine – Two Rank Functions



Relevance Ranking

- Inverted index
 - A data structure for supporting text queries
 - like index in a book



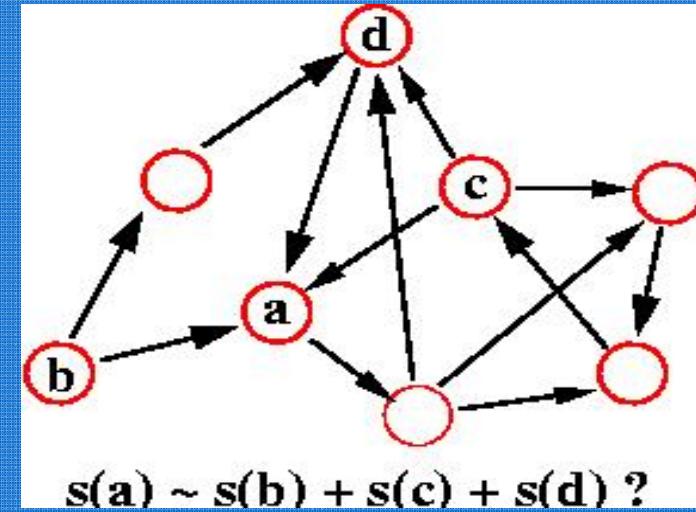
aalborg	3452, 11437,
:	
:	
:	
arm	4, 19, 29, 98, 143, ...
armada	145, 457, 789, ...
armadillo	678, 2134, 3970, ...
armani	90, 256, 372, 511, ...
:	
:	
:	
:	
zz	602, 1189, 3209, ...

inverted index

The PageRank Algorithm

- Basic idea

- *significance of a page is determined by the significance of the pages linking to it*



- More precisely:

- Link graph: adjacency matrix A ,
 - Constructs a probability transition matrix M by renormalizing each row of A to sum to 1
 - Treat the web graph as a markov chain (random surfer)

$$\varepsilon U + (1 - \varepsilon)M \quad U_{ij} = 1/n \text{ for all } i, j$$

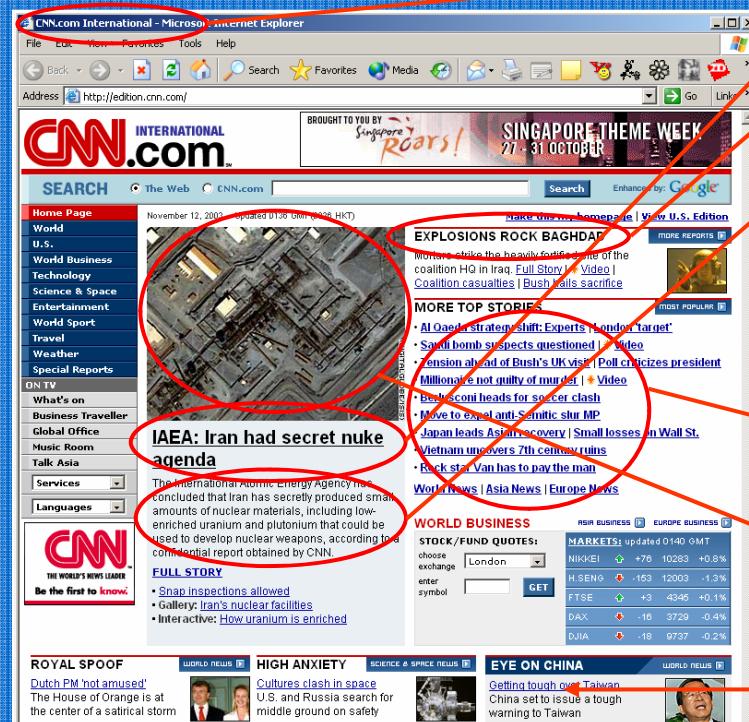
- The vector of PageRank scores p is then defined to be the stationary distribution of this Markov chain. Equivalently, p is the principal right eigenvector of the transition matrix

$$(\varepsilon U + (1 - \varepsilon)M)^T$$

$$(\varepsilon U + (1 - \varepsilon)M)^T p = p$$

Layout Structure

- Compared to plain text, a web page is a 2D presentation
 - Rich visual effects created by different term types, formats, separators, blank areas, colors, pictures, etc
 - Different parts of a page are not equally important



Title: CNN.com International

H1: IAEA: Iran had secret nuke agenda
H3: EXPLOSIONS ROCK BAGHDAD

TEXT BODY (with position and font type): The International Atomic Energy Agency has concluded that Iran has secretly produced small amounts of nuclear materials including low enriched uranium and plutonium that could be used to develop nuclear weapons according to a confidential report obtained by CNN...

Hyperlink:

- URL: [http://www.cnn.com/...](http://www.cnn.com/)

- Anchor Text: Al Qaeda...

Image:

- URL: [http://www.cnn.com/image/...](http://www.cnn.com/image/)

- Alt & Caption: Iran nuclear ...

Anchor Text: CNN Homepage News ...

Web Page Block—Better Information Unit

CNN.com International - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://edition.cnn.com/

CNN.com INTERNATIONAL

BROUGHT TO YOU BY Singapore Roars! SINGAPORE THEME WEEK 27 - 31 OCTOBER

SEARCH The Web CNN.com

November 12, 2003 -- Updated 0136 GMT (0936 HKT)

EXPLOSIONS ROCK BAGHDAD
MORE REPORTS
Mortars strike the heavily fortified site of the coalition HQ in Iraq: [Full Story](#) | [*Video](#) | [Coalition casualties](#) | [Bush hails sacrifice](#)

MORE TOP STORIES
[Al Qaeda strategy shift: Experts](#) | [London 'target'](#)
[Saudi bomb suspects questioned](#) | [*Video](#)
[Tension ahead of Bush's UK visit](#) | [Poll criticizes president](#)
[Millionaire not guilty of murder](#) | [*Video](#)
[Berlusconi heads for soccer clash](#)
[Move to expel anti-Semitic slur MP](#)
[Japan leads Asian recovery](#) | [Small losses on Wall St.](#)
[Vietnam uncovers 2nd century ruins](#)
[Rock star Van has to pay the piper](#)
[World News](#) | [Asia News](#) | [Europe News](#)

WORLD BUSINESS
STOCK/FUND QUOTES:
 choose exchange enter symbol
 MARKETS: updated 0140 GMT

	NIKKEI	H.SENG	FTSE	DAX
▲	+76 10283 +0.8%	-153 12003 -1.3%	+3 4345 +0.1%	-16 3729 -0.4%
▲	0 9737 -0.2%			

ROYAL SPOOF WORLD NEWS HIGH ANXIETY SCIENCE & SPACE NEWS EYE ON CHINA WORLD NEWS
 Dutch PM 'not amused'
 The House of Orange is at the center of a satirical storm

Cultures clash in space U.S. and Russia search for middle ground on safety

Getting tough over Taiwan China set to issue a tough warning to Taiwan

Web Page Blocks

Importance = Low

Importance = Med

Importance = High

Motivation for VIPS (VIision-based Page Segmentation)

- Problems of treating a web page as an atomic unit
 - Web page usually contains not only pure content
 - Noise: navigation, decoration, interaction, ...
 - Multiple topics
 - Different parts of a page are not equally important
- Web page has internal structure
 - Two-dimension logical structure & Visual layout presentation
 - > Free text document
 - < Structured document
- Layout – the 3rd dimension of Web page
 - 1st dimension: content
 - 2nd dimension: hyperlink

Is DOM a Good Representation of Page Structure?

- Page segmentation
 - Extract structural elements: TITLE, H1~H6, ...
 - *DOM is more like a tree than a grid; it necessarily reflects the page structure.*
- How about XML?
 - A long way to go!

Page Analysis - IEEE Standards Association Home Page.htm
 http://standards.ieee.org

Page Analysis - Yahooligans! E-Cards
 http://ecards.yahooligans.com/content/ecards/category?c=133&g=16

YAHOO!LIGANS! E-Cards

Home > Yahooligans! E-Cards > Send an E-Card

Animals

1 Choose a Card 2 Address the Card 3 Choose a Message 4 Preview/Send Card

Just a Hello From My Doghouse? Woo-hoo! A Barking

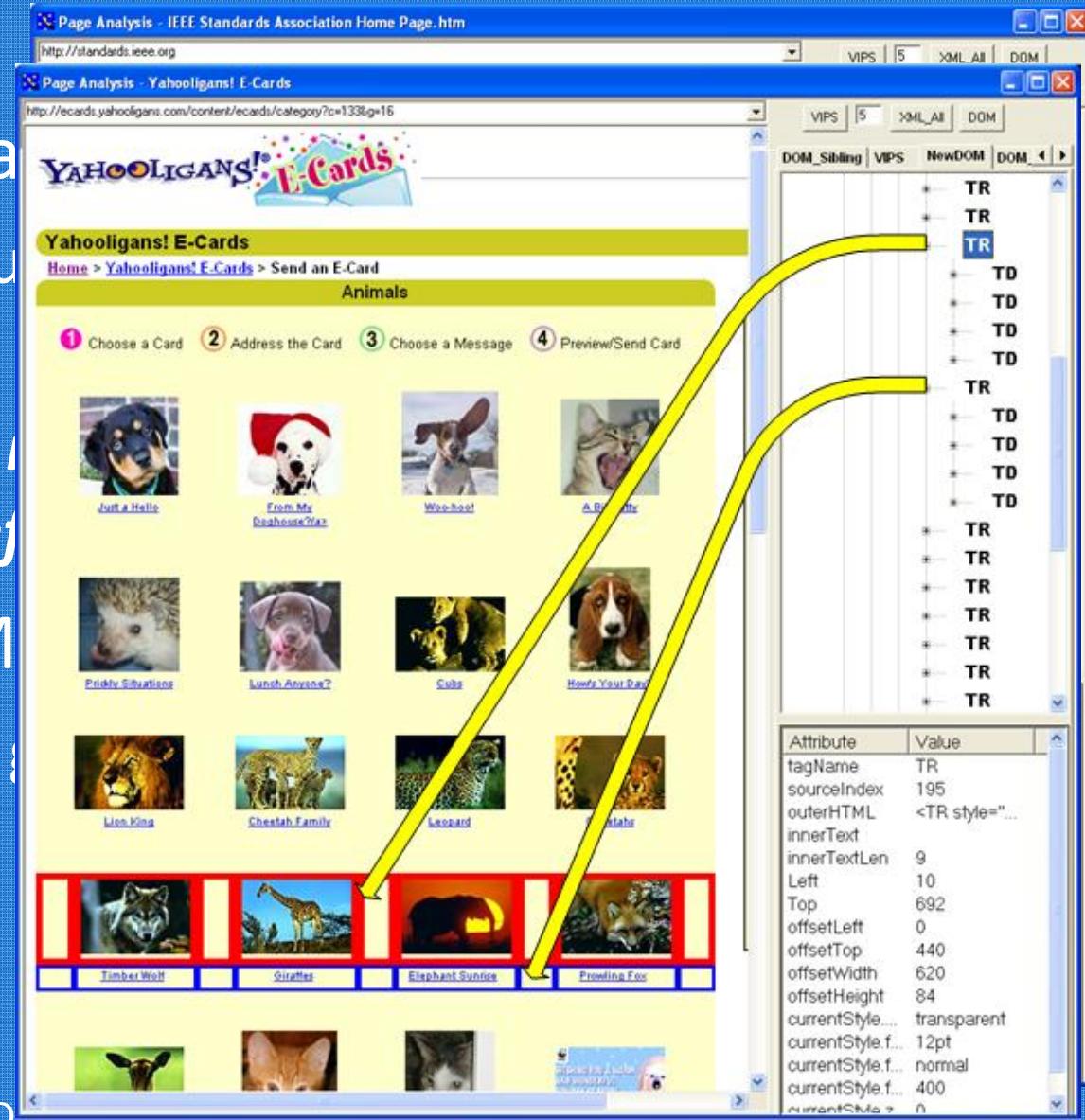
Prickly Situation Lunch Anyone? Cuba How's Your Day

Lion King Cheetah Family Leopard Saber

Timber Wolf Giraffes Elephant Surprise Prowling Fox

Attribute Value

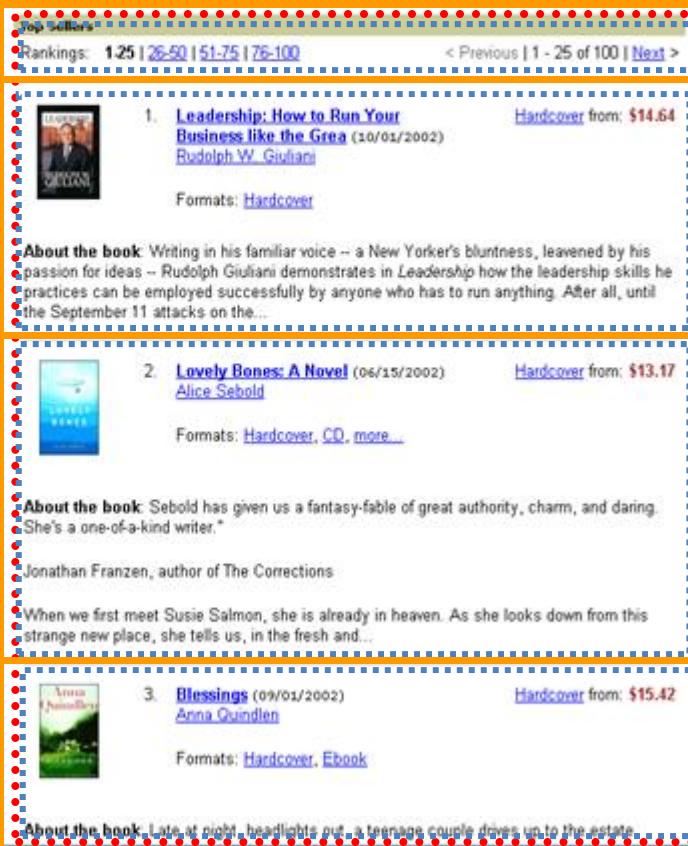
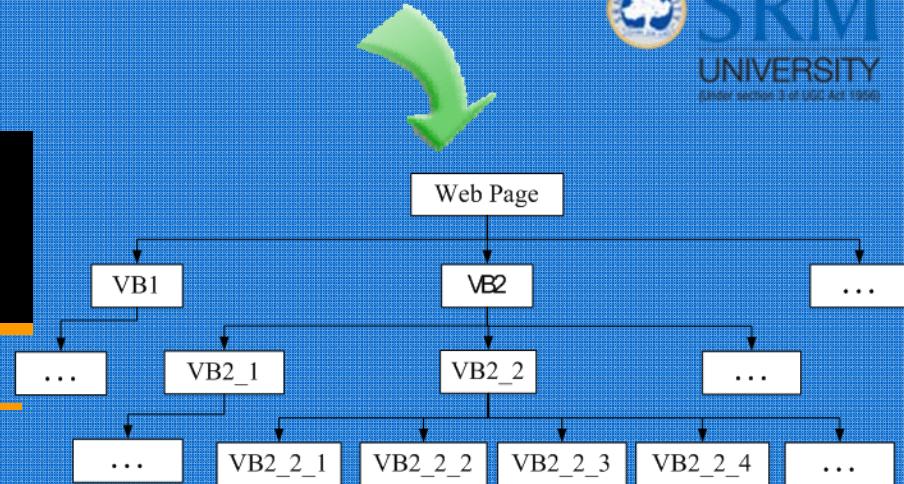
tagName	TR
sourceIndex	195
outerHTML	<TR style="background-color: red; color: white; text-align: center; padding: 5px; margin-bottom: 10px;>
innerText	
innerTextLen	9
Left	10
Top	692
offsetLeft	0
offsetTop	440
offsetWidth	620
offsetHeight	84
currentStyle.fontSize	12pt
currentStyle.fontWeight	normal
currentStyle.fontColor	400
currentStyle.color	0



VIPS Algorithm

- Motivation:
 - In many cases, topics can be distinguished with visual clues. Such as position, distance, font, color, etc.
- Goal:
 - Extract the semantic structure of a web page based on its visual presentation.
- Procedure:
 - Top-down partition the web page based on the separators
- Result
 - A tree structure, each node in the tree corresponds to a block in the page.
 - Each node will be assigned a value (Degree of Coherence) to indicate how coherent of the content in the block based on visual perception.
 - Each block will be assigned an importance value
 - Hierarchy or flat

VIPS: An Example

- A hierarchical structure of layout block
- A *Degree of Coherence (DOC)* is defined for each block
 - Show the intra coherence of the block
 - *DoC* of child block must be no less than its parent's
- The *Permitted Degree of Coherence (PDOC)* can be pre-defined to achieve different granularities for the content structure
 - The segmentation will stop only when all the blocks' *DoC* is no less than *PDoC*
 - The smaller the *PDoC*, the coarser the content structure would be

Block-based Web Search

- Index block instead of whole page
- Block retrieval
 - Combing DocRank and BlockRank
- Block query expansion
 - Select expansion term from relevant blocks

A Sample of User Browsing Behavior

Welcome, cai_deng

Personalize News Home Page - Sign Out

Yahoo! News Fri, Feb 20, 2004

Search News Stories for Search Advanced

[News Home](#)

► [Top Stories](#)

- [Crimes and Trials](#)
- [Most Popular](#)
- [U.S. National](#)
- [Business](#)
- [World](#)
- [Entertainment](#)
- [Sports](#)
- [Technology](#)
- [Politics](#)
- [Health](#)
- [Science](#)
- [Oddly Enough](#)
- [Health](#)
- [Oddly](#)
- [Op/Ed](#)
- [Local](#)
- [Comics](#)
- [News Photos](#)
- [Most Popular](#)
- [Weather](#)
- [Audio/Video](#)
- [Full Coverage](#)

[Full Coverage](#)

More about Iraq

Supreme Court - AP

High Court to Mull 'Enemy Combatant' Rule

 Associated Press

1 hour, 12 minutes ago

By GINA HOLLAND, Associated Press Writer

WASHINGTON - The Supreme Court agreed Friday to decide whether U.S. citizens arrested in America as "enemy combatants" may be held indefinitely without access to lawyers or courts, setting the stage for a major ruling on presidential powers versus civil liberties.



[AP Photo](#)

The justices had already agreed to consider the government's detentions of terror suspects — American and foreign — caught overseas and held incommunicado.

But the case of former Chicago gang member Jose Padilla is seen as the one that will set a key standard as the government pursues the open-ended war on terror: Does the threat of attack justify giving federal authorities unprecedented legal latitude to hold their own citizens?

"The Padilla case is the most significant case for the government," said Scott Silliman, a Duke University law professor. "The court will have the opportunity to define what it is we call the 'war on terrorism.'"

CLICK HERE FOR INSTANT QUOTES

ReliaQuote
A Better Way to Buy Life Insurance

10-Year Level Term Life Insurance
Male/Female
Monthly Premiums
No Nicotine

Save Up to 70%

\$250,000		
AGE	35	45
M	\$10.22	\$18.49
F	\$9.14	\$15.44

\$500,000		
AGE	35	45
M	\$16.10	\$32.63
F	\$13.92	\$26.54

Sample rates underwritten by Lincoln National Life Insurance Company.



ImageRank

- Relevance Ranking
- Importance Ranking
- Combined Ranking

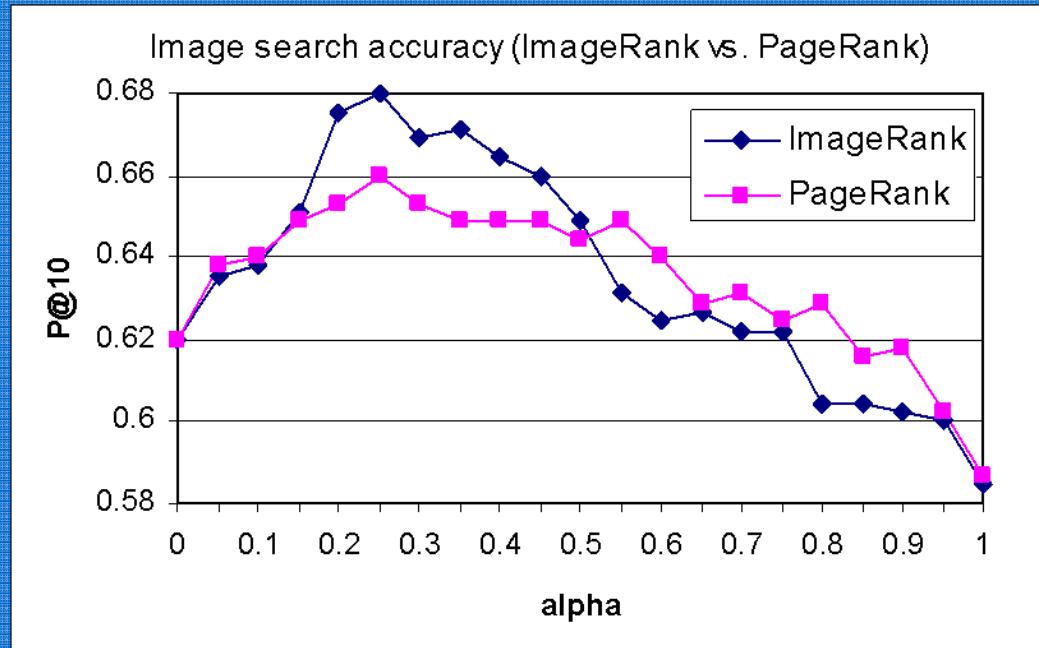


ImageRank vs. PageRank

- Dataset
 - 26.5 millions web pages
 - 11.6 millions images
- Query set
 - 45 hot queries in Google image search statistics
- Ground truth
 - Five volunteers were chosen to evaluate the top 100 results re-turned by the system (iFind)
- Ranking method

$$s(\mathbf{x}) = \alpha \cdot rank_{importance}(\mathbf{x}) + (1 - \alpha) \cdot rank_{relevance}(\mathbf{x})$$

ImageRank vs PageRank



- **Image search accuracy using ImageRank and PageRank. Both of them achieved their best results at $\alpha=0.25$.**

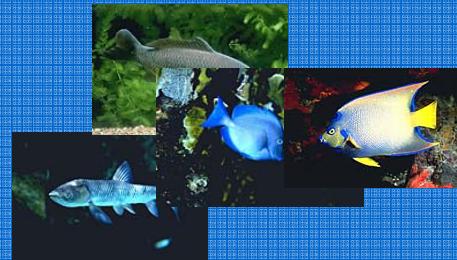
Example on Image Clustering & Embedding

1710 JPG images in 1287 pages are crawled within the website
<http://www.yahooligans.com/content/animals/>

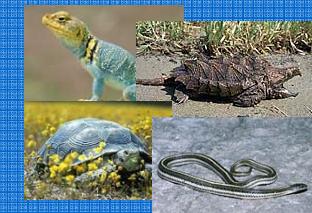
Six Categories



Mammal



Fish



Reptile



Bird



Amphibian



Insect

Yahoo!ligans! Animals

Search for Animals: Search

Mammals Fishes Insects Birds Amphibians Reptiles

Home > Animals > Fishes > Great Barracuda

Fishes

Great Barracuda
Sphyraena barracuda

Smaller Great Barracudas can be found in shallow inshore waters over sandy bottoms. Larger individuals are more often found offshore and are usually solitary. Great Barracudas feed mainly on fishes and occasionally on small shrimp. They are curious fish, and often follow snorkelers or divers. Attacks on humans are rare and probably occur when barracudas try to take upended fish as prey.

Look For: A slender fish with 2 dorsal fins and a large mouth. Gray above, silvery sides. Dark spots above anal fin.

Length: 6'

Habitat: Warm coastal waters, open ocean. Juveniles often near shore.

Range: Pacific and Atlantic coasts. Caution: Known to attack swimmers.

Learn more about Fishes: Select a topic:

Related Species:

- Saltfish** *Ictalurus punctatus*
- Atlantic Mackerel** *Scomber scombrus*
- Yellowtail** *Thunnus albacares*

Get the Big Picture

- Animals
- Anthropology and Archaeology
- Astronomy and Space News
- Environment and Nature News
- All Stories

ADVERTISEMENT

PHOTO CONTEST

Kellogg's WIN MOVIE TICKETS

Great Barracuda 

Red-tailed Hawk 

Red-tailed Hawk *Buteo jamaicensis*

The Red-tail divides its time between perching in trees and soaring, always looking for small mammals, birds, or reptiles. Like other buteos (soaring hawks), it drifts in wide circles in the sky.

Look For: Brown above, often with dark streaks on belly. May be all brown in West. The tail is brown in juveniles, orangish in adults.

Length: 19-25"

Habitat: Open country, forests.

Range: Alaska and Canada (mainly only in summer) and south throughout U.S.

Learn more about birds: Select a topic:

Related Species:

- Red-tailed Hawk** *Buteo lineatus*
- Northern Harrier** *Circus cyaneus*
- Peregrine Falcon** *Falco peregrinus*

Get the Big Picture

- Animals
- Anthropology and Archaeology
- Astronomy and Space News
- Environment and Nature News
- All Stories

ADVERTISEMENT

PHOTO CONTEST

Rubber Rick 

HIT US WITH YOUR BEST SHOT!

PHOTO CONTEST

ADVERTISEMENT

IMPRESS YOUR PARENTS

YAHOO!ligans! Animals

Search for Animals: Search

Mammals Fishes Insects Birds Amphibians Reptiles

Home > Animals > Fishes > Fishes

Fishes

Insects

Birds

Amphibians

Arctic Fox 

Mammals

Arctic Fox *Alopex lagopus*

The Arctic Fox is well suited to its subzero habitat: it has a compact body with short legs and ears (body heat is lost through long ears and legs), dense fur, and thick hair on the footpads, which insulates against the cold and provides traction on ice. Winter fur develops in October: The coat thickens, and the new hairs are much lighter, providing camouflage against snow and ice. Sadly, this fox has been heavily hunted for its beautiful fur coat.

Look For: A fox of the extreme north, pure white in winter, brownish-gray in summer.

Length: Body 19-22" long.

Habitat: Tundra and sea ice.

Range: Alaska and northern Canada.

Get the Big Picture

- Animals
- Anthropology and Archaeology
- Astronomy and Space News
- Environment and Nature News
- All Stories

Learn more about mammals: Select a topic:

Related Species:

- Common Gray Fox** *Urocyon cinereoargenteus* 
- Red Fox** *Vulpes vulpes* 
- Kit Fox** *Vulpes macrotis* 

Get the Big Picture

- Animals
- Anthropology and Archaeology
- Astronomy and Space News
- Environment and Nature News
- All Stories

ADVERTISEMENT

PHOTO CONTEST

Kellogg's WIN MOVIE TICKETS

YAHOO!ligans! Animals

Search for Animals: Search

Mammals Fishes Insects Birds Amphibians Reptiles

Home > Animals > Mammals > Common Gray Fox

Mammals

Common Gray Fox *Urocyon cinereoargenteus* 

Although it is a member of the dog family, the Common Gray Fox is a good tree climber and often hides in trees. This fox feeds on cottontail rabbits, mice, weasels, shrews, small birds, birds, insects, and plant material, including corn, apples, persimmons, nuts, cherries, grapes, grass, and blackberries. Grasshoppers and crickets are often a very important part of the diet in late summer and autumn.

Look For: A gray fox with a black-and-white face and red around the ears, neck, chest, and lower sides. Tail black on top and at tip.

Length: Body 24" long.

Habitat: Woodlands and brushy areas.

Range: Most of the U.S., but not in Rockies or parts of Great Plains.

Get the Big Picture

Related Species:

- Arctic Fox** *Alopex lagopus* 

Get the Big Picture

- Animals
- Anthropology and Archaeology
- Astronomy and Space News
- Environment and Nature News
- All Stories

ADVERTISEMENT

PHOTO CONTEST

Kellogg's WIN MOVIE TICKETS

12/26/2012

Data Mining - Principles and

Algorithms

Web Image Search Result Presentation



- Two different topics in the search result
- A possible solution:
 - Cluster search results into different semantic groups

Three kinds of WWW image representation

- Visual Feature Based Representation
 - Traditional CBIR
- Textual Feature Based Representation
 - Surrounding text in image block
- Link Graph Based Representation
 - Image graph embedding

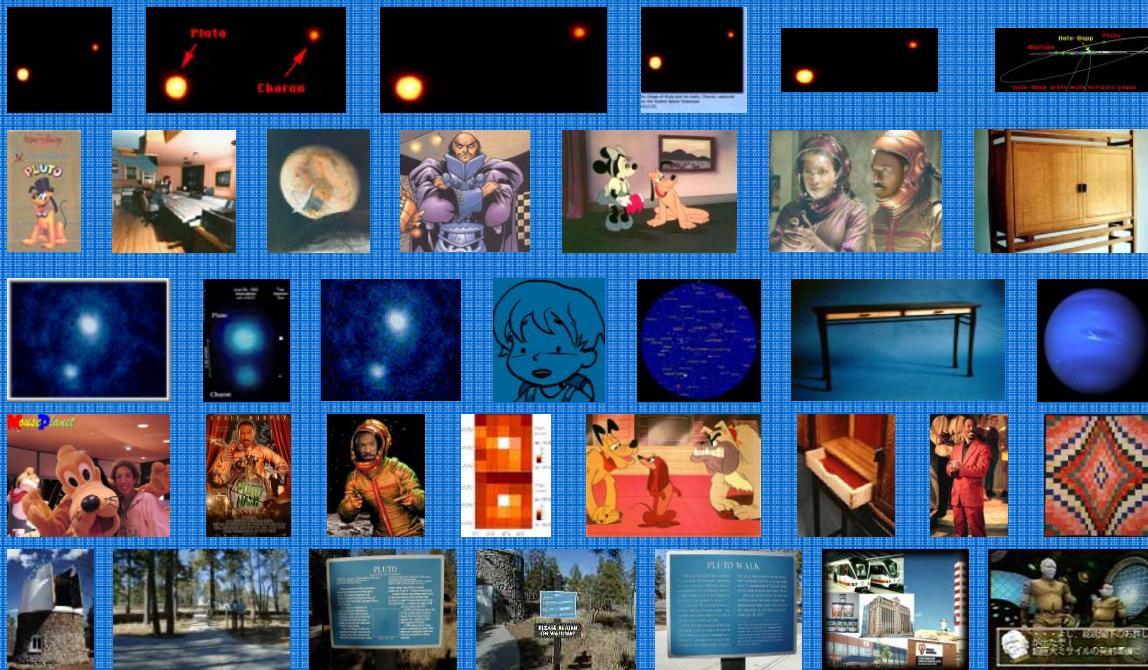
Hierarchical Clustering

- Clustering based on three representations
 - Visual feature
 - Hard to reflect the semantic meaning
 - Textual feature
 - Semantic
 - Sometimes the surrounding text is too little
 - Link graph:
 - Semantic
 - Many disconnected sub-graph (too many clusters)
- Two Steps:
 - Using texts and link information to get semantic clusters
 - For each cluster, using visual feature to re-organize the images to facilitate user's browsing

Our System

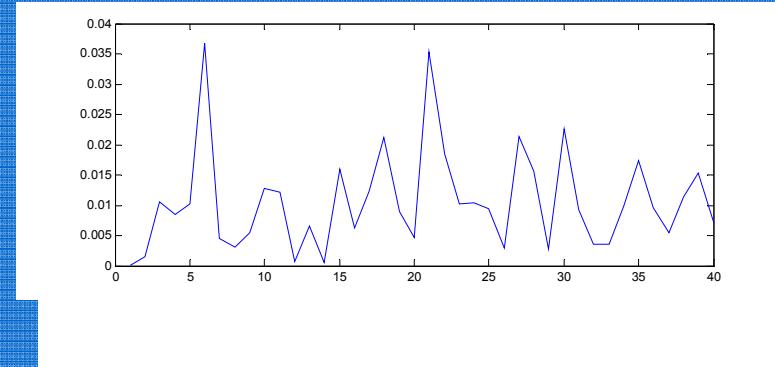
- Dataset
 - 26.5 millions web pages
http://dir.yahoo.com/Arts/Visual_Arts/Photography/Museums_and_Galleries/
 - 11.6 millions images
 - Filter images whose ratio between width and height are greater than 5 or smaller than 1/5
 - Removed images whose width and height are both smaller than 60 pixels
- Analyze pages and index images
 - VIPS: Pages → Blocks
 - Surrounding texts used to index images
- An illustrative example
 - Query “Pluto”
 - Top 500 results

Clustering Using Visual Feature



- From the perspectives of color and texture, the clustering results are quite good. Different clusters have different colors and textures. However, from semantic perspective, these clusters make little sense.

Clustering Using Textual Feature



- Six semantic categories are correctly identified if we choose $k = 6$.

Summary

- More improvement on web search can be made by mining webpage Layout structure
- Leverage visual cues for web information analysis & information extraction
- Demos:
 - <http://www.ews.uiuc.edu/~dengcai2>
 - Papers
 - VIPS demo & dll

Review Questions

- Define special data mining?
- What is document rank based on the context of text mining?
- Can we construct a special data warehouse?
- List the two types of measures in a special data cube?
- Enlist the two types of multi media indexing and retrieval system?
- Give a note on multimedia data cube?
- What is information retrieval?
- List the methods for information retrieval?
- What is meant by authoritative web page?
- What is web usage mining?

Bibliography

- Data mining concepts and Techniques by Jiawei Han and Micheline Kamber