

An Introduction to Data Mining

Chapter#1



Why Data Mining

⌘ Credit ratings/targeted marketing:

- ☒ Given a database of 100,000 names, which persons are the least likely to default on their credit cards?
- ☒ Identify likely responders to sales promotions

⌘ Fraud detection

- ☒ Which types of transactions are likely to be fraudulent, given the demographics and transactional history of a particular customer?

⌘ Customer relationship management:

- ☒ Which of my customers are likely to be the most loyal, and which are most likely to leave for a competitor? :

Data Mining helps extract such information

Data mining



- ⌘ Process of semi-automatically analyzing large databases to find patterns that are:
 - ☑ valid: hold on new data with some certainty
 - ☑ novel: non-obvious to the system
 - ☑ useful: should be possible to act on the item
 - ☑ understandable: humans should be able to interpret the pattern
- ⌘ Also known as Knowledge Discovery in Databases (KDD)

Applications



- ⌘ Banking: loan/credit card approval
 - ☑ predict good customers based on old customers
- ⌘ Customer relationship management:
 - ☑ identify those who are likely to leave for a competitor.
- ⌘ Targeted marketing:
 - ☑ identify likely responders to promotions
- ⌘ Fraud detection: telecommunications, financial transactions
 - ☑ from an online stream of event identify fraudulent events
- ⌘ Manufacturing and production:
 - ☑ automatically adjust knobs when process parameter changes

Applications (continued)



⌘ Medicine: disease outcome, effectiveness of treatments

☑ analyze patient disease history: find relationship between diseases

⌘ Molecular/Pharmaceutical: identify new drugs

⌘ Scientific data analysis:

☑ identify new galaxies by searching for sub clusters

⌘ Web site/store design and promotion:

☑ find affinity of visitor to pages and modify layout

The KDD process

⌘ Problem formulation

⌘ Data collection

☒ subset data: sampling might hurt if highly skewed data

☒ feature selection: principal component analysis, heuristic search

⌘ Pre-processing: cleaning

☒ name/address cleaning, different meanings (annual, yearly), duplicate removal, supplying missing values

⌘ Transformation:

☒ map complex objects e.g. time series data to features e.g. frequency

⌘ Choosing mining task and mining method:

⌘ Result evaluation and Visualization:

Knowledge discovery is an iterative process

Relationship with other fields



- ⌘ Overlaps with machine learning, statistics, artificial intelligence, databases, visualization but more stress on
 - ☑ scalability of number of features and instances
 - ☑ stress on algorithms and architectures whereas foundations of methods and formulations provided by statistics and machine learning.
 - ☑ automation for handling large, heterogeneous data

Some basic operations



⌘ Predictive:

- ☑ Regression
- ☑ Classification
- ☑ Collaborative Filtering

⌘ Descriptive:

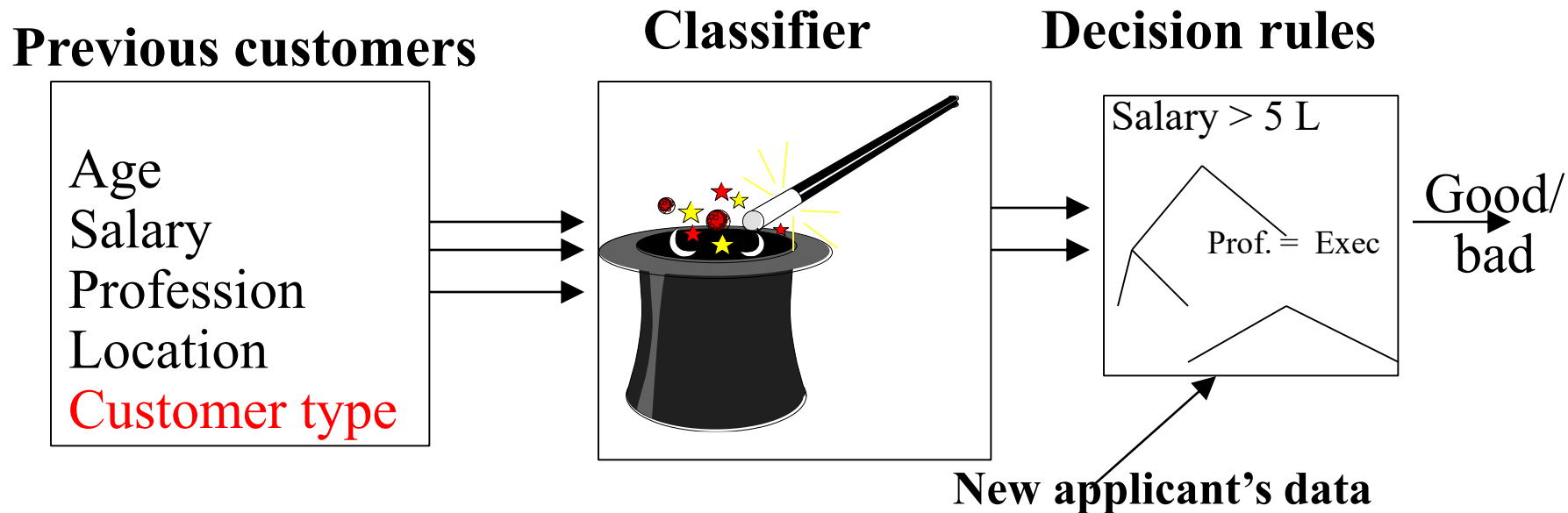
- ☑ Clustering / similarity matching
- ☑ Association rules and variants
- ☑ Deviation detection



Classification (Supervised learning)

Classification

⌘ Given old data about customers and payments, predict new applicant's loan eligibility.



Classification methods

- ⌘ **Goal:** Predict class $C_i = f(x_1, x_2, \dots, X_n)$
- ⌘ Regression: (linear or any other polynomial)
 - ⌘ $a \cdot x_1 + b \cdot x_2 + c = C_i$.
- ⌘ Nearest neighbour
- ⌘ Decision tree classifier: divide decision space into piecewise constant regions.
- ⌘ Probabilistic/generative models
- ⌘ Neural networks: partition by non-linear boundaries

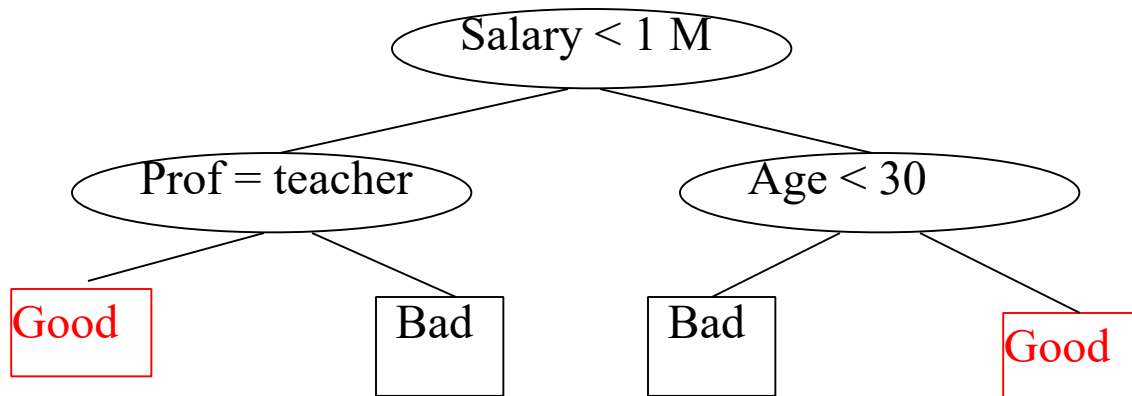
Nearest neighbor



- ⌘ Define proximity between instances, find neighbors of new instance and assign majority class
- ⌘ Case based reasoning: when attributes are more complicated than real-valued.
 - Pros
 - + Fast training
 - Cons
 - Slow during application.
 - No feature selection.
 - Notion of proximity vague

Decision trees

- Tree where internal nodes are simple decision rules on one or more attributes and leaf nodes are predicted class labels.



Decision tree classifiers

- ⌘ Widely used learning method
- ⌘ Easy to interpret: can be re-represented as if-then-else rules
- ⌘ Approximates function by piece wise constant regions
- ⌘ Does not require any prior knowledge of data distribution, works well on noisy data.
- ⌘ Has been applied to:
 - ⊞ classify medical patients based on the disease,
 - ⊞ equipment malfunction by cause,
 - ⊞ loan applicant by likelihood of payment.

Pros and Cons of decision trees



· Pros

- + Reasonable training time
- + Fast application
- + Easy to interpret
- + Easy to implement
- + Can handle large number of features

More information:

<http://www.stat.wisc.edu/~limt/treeprogs.html>

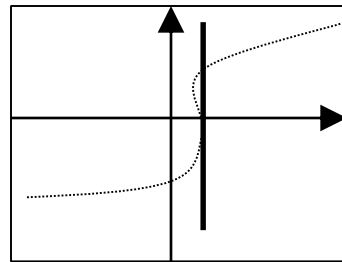
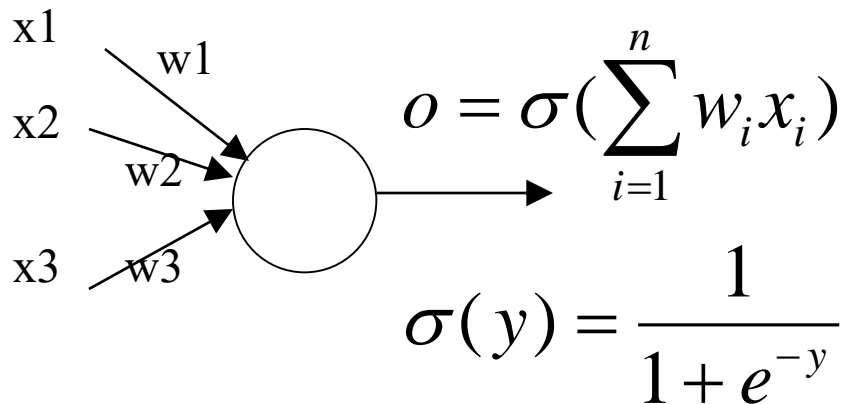
· Cons

- Cannot handle complicated relationship between features
- simple decision boundaries
- problems with lots of missing data

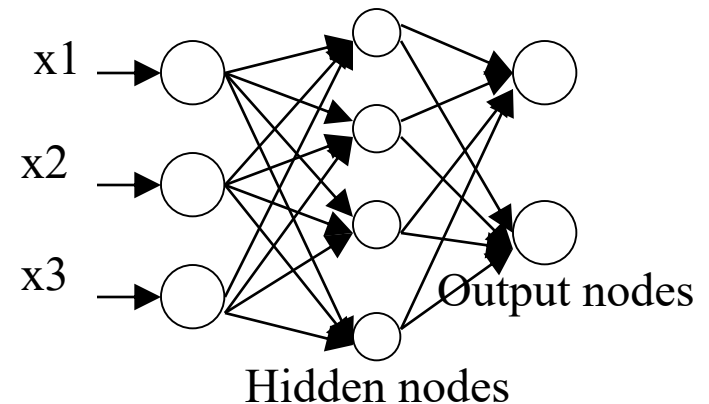
Neural network

⌘ Set of nodes connected by directed weighted edges

Basic NN unit



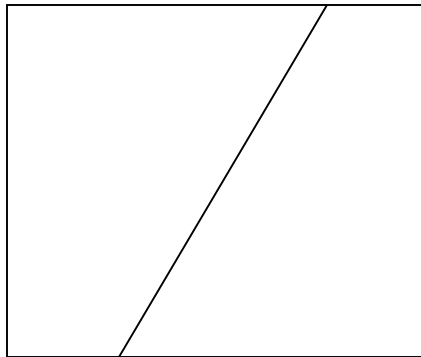
A more typical NN



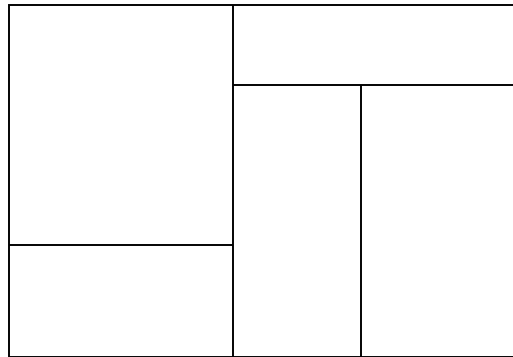
Neural networks

⌘ Useful for learning complex data like handwriting, speech and image recognition

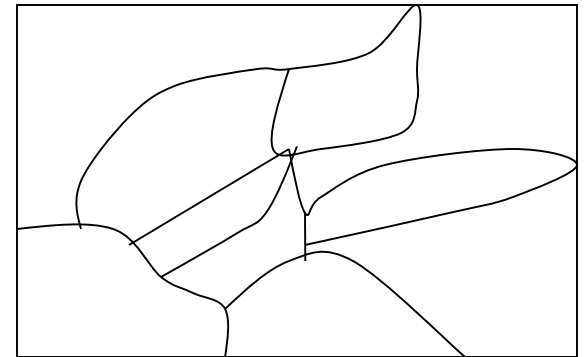
Decision boundaries:



Linear regression



Classification tree



Neural network

Pros and Cons of Neural Network



· Pros

- + Can learn more complicated class boundaries
- + Fast application
- + Can handle large number of features

· Cons

- Slow training time
- Hard to interpret
- Hard to implement: trial and error for choosing number of nodes


Conclusion: Use neural nets only if decision-trees/NN fail.

Bayesian learning

- ⌘ Assume a probability model on generation of data.
- ⌘ predicted class : $c = \max_{c_j} p(c_j | d) = \max_{c_j} \frac{p(d | c_j) p(c_j)}{p(d)}$
- ⌘ Apply bayes theorem to find most likely class as:

$$c = \max_{c_j} \frac{p(c_j)}{p(d)} \prod_{i=1}^n p(a_i | c_j)$$

- ⌘ Naïve bayes: Assume attributes conditionally independent given class value
- ⌘ Easy to learn probabilities by counting,
- ⌘ Useful in some domains e.g. text



Clustering or Unsupervised Learning

Clustering



- ⌘ Unsupervised learning when old data with class labels not available e.g. when introducing a new product.
- ⌘ Group/cluster existing customers based on time series of payment history such that similar customers in same cluster.
- ⌘ Key requirement: Need a good measure of similarity between instances.
- ⌘ Identify micro-markets and develop policies for each

Applications



- ⌘ Customer segmentation e.g. for targeted marketing
 - ☑ Group/cluster existing customers based on time series of payment history such that similar customers in same cluster.
 - ☑ Identify micro-markets and develop policies for each
- ⌘ Collaborative filtering:
 - ☑ group based on common items purchased
- ⌘ Text clustering
- ⌘ Compression

Distance functions

- ⌘ Numeric data: euclidean, manhattan distances
- ⌘ Categorical data: 0/1 to indicate presence/absence followed by
 - ☒ Hamming distance (# dissimilarity)
 - ☒ Jaccard coefficients: $\# \text{similarity in 1s} / (\# \text{ of 1s})$
 - ☒ data dependent measures: similarity of A and B depends on co-occurrence with C.
- ⌘ Combined numeric and categorical data:
 - ☒ weighted normalized distance:

Clustering methods



⌘ Hierarchical clustering

- ☑ agglomerative Vs divisive
- ☑ single link Vs complete link

⌘ Partitional clustering

- ☑ distance-based: K-means
- ☑ model-based: EM
- ☑ density-based:

Partitional methods: K-means

⌘ Criteria: minimize sum of square of distance

- ⊗ Between each point and centroid of the cluster.

- ⊗ Between each pair of points in the cluster

⌘ Algorithm:

- ⊗ Select initial partition with K clusters: random, first K, K separated points

- ⊗ Repeat until stabilization:

- ⊗ Assign each point to closest cluster center

- ⊗ Generate new cluster centers

- ⊗ Adjust clusters by merging/splitting

Collaborative Filtering

- ⌘ Given database of user preferences, predict preference of new user
- ⌘ Example: predict what new movies you will like based on
 - ☑ your past preferences
 - ☑ others with similar past preferences
 - ☑ their preferences for the new movies
- ⌘ Example: predict what books/CDs a person may want to buy
 - ☑ (and suggest it, or give discounts to tempt customer)

Collaborative recommendation

- Possible approaches:

- Average vote along columns [Same prediction for all]
- Weight vote based on similarity of likings [GroupLens]

	Rangeela	QSQT	100 day	Anand	Sholay	Deewar	Vertigo
Smita							
Vijay							
Mohan							
Rajesh							
Nina							
Nitin	?	?		?	?	?	?

Cluster-based approaches

- ⌘ External attributes of people and movies to cluster

- ⊞ age, gender of people

- ⊞ actors and directors of movies.

- ⊞ [May not be available]

- ⌘ Cluster people based on movie preferences

- ⊞ misses information about similarity of movies

- ⌘ Repeated clustering:

- ⊞ cluster movies based on people, then people based on movies, and repeat

- ⊞ ad hoc, might smear out groups

Example of clustering

	Anand QSQT		Rangeela	100 days	Vertigo	Deewar	Sholay
Vijay							
Rajesh							
Mohan							
Nina							
Smita							
Nitin	?	?	?		?	?	?

Model-based approach

- ⌘ People and movies belong to unknown classes
- ⌘ P_k = probability a random person is in class k
- ⌘ P_l = probability a random movie is in class l
- ⌘ P_{kl} = probability of a class- k person liking a class- l movie
- ⌘ Gibbs sampling: iterate
 - ⊞ Pick a person or movie at random and assign to a class with probability proportional to P_k or P_l
 - ⊞ Estimate new parameters
 - ⊞ Need statistics background to understand details



Association Rules

Association rules

T

- ⌘ Given set T of groups of items
- ⌘ Example: set of item sets purchased
- ⌘ Goal: find all rules on itemsets of the form $a \rightarrow b$ such that
 - ⌘ support of a and b $>$ user threshold s
 - ⌘ conditional probability (confidence) of b given a $>$ user threshold c
- ⌘ Example: Milk \rightarrow bread
- ⌘ Purchase of product A \rightarrow service B

Milk, cereal
Tea, milk
Tea, rice, bread
cereal

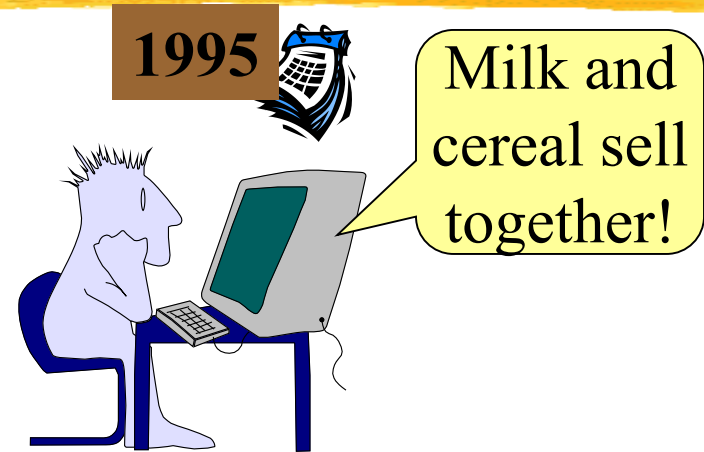
Variants



- ⌘ High confidence may not imply high correlation
- ⌘ Use correlations. Find expected support and large departures from that interesting..
 - 📄 see statistical literature on contingency tables.
- ⌘ Still too many rules, need to prune...

Prevalent \neq Interesting

- ⌘ Analysts already know about prevalent rules
- ⌘ Interesting rules are those that *deviate* from prior expectation
- ⌘ Mining's payoff is in finding *surprising* phenomena



What makes a rule surprising?

⌘ Does not match prior expectation

☑ Correlation between milk and cereal remains roughly constant over time

⌘ Cannot be trivially derived from simpler rules

☑ Milk 10%, cereal 10%

☑ Milk and cereal 10% ... surprising

☑ Eggs 10%

☑ Milk, cereal and eggs 0.1% ... surprising!

☑ Expected 1%

Applications of fast itemset counting



Find correlated events:

- ⌘ Applications in medicine: find redundant tests
- ⌘ Cross selling in retail, banking
- ⌘ Improve predictive capability of classifiers that assume attribute independence
- ⌘ New similarity measures of categorical attributes [**Mannila et al, KDD 98**]



Data Mining in Practice

Application Areas



Industry

Finance

Insurance

Telecommunication

Transport

Consumer goods

Data Service providers

Utilities

Application

Credit Card Analysis

Claims, Fraud Analysis

Call record analysis

Logistics management

promotion analysis

Value added data

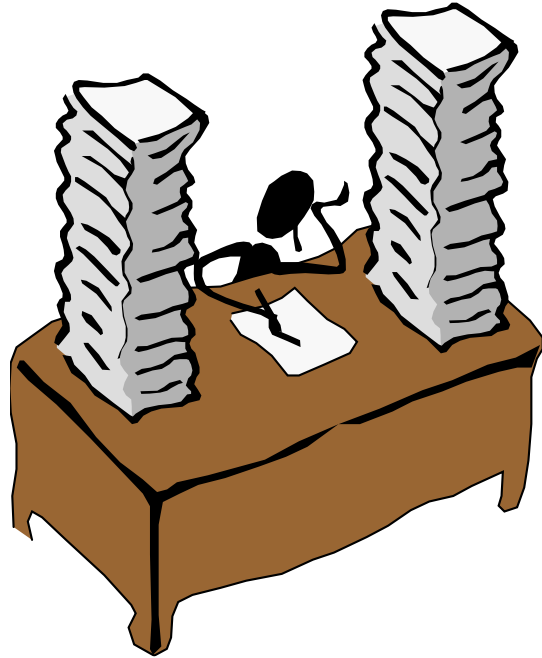
Power usage analysis

Why Now?



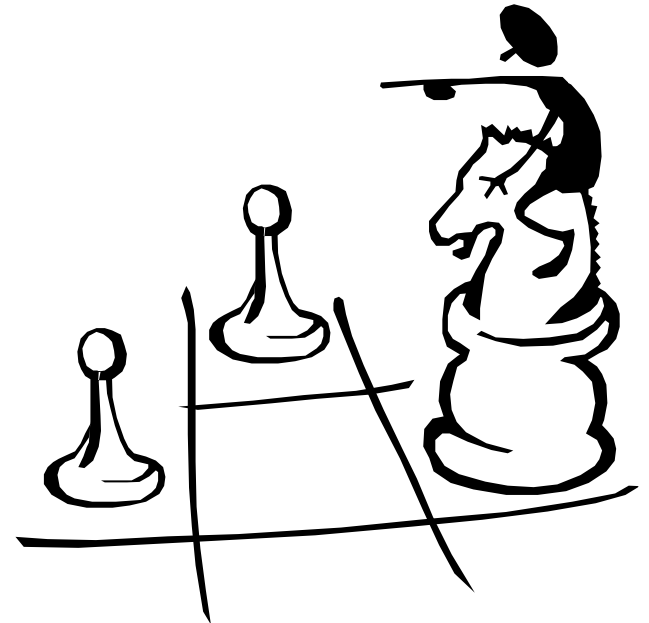
- ⌘ Data is being produced
- ⌘ Data is being warehoused
- ⌘ The computing power is available
- ⌘ The computing power is affordable
- ⌘ The competitive pressures are strong
- ⌘ Commercial products are available

Data Mining works with Warehouse Data



⌘ Data Warehousing provides the Enterprise with a memory

Ñ Data Mining provides the Enterprise with intelligence



Usage scenarios



⌘ Data warehouse mining:

- ☑ assimilate data from operational sources

- ☑ mine static data

⌘ Mining log data

⌘ Continuous mining: example in process control

⌘ Stages in mining:

- ☑ data selection → pre-processing: cleaning

 - transformation → mining → result

 - evaluation → visualization

Mining market



⌘ Around 20 to 30 mining tool vendors

⌘ Major tool players:

- ☒ Clementine,
- ☒ IBM's Intelligent Miner,
- ☒ SGI's MineSet,
- ☒ SAS's Enterprise Miner.

⌘ All pretty much the same set of tools

⌘ Many embedded products:

- ☒ fraud detection:
- ☒ electronic commerce applications,
- ☒ health care,
- ☒ customer relationship management: Epiphany

Vertical integration:

Mining on the web

⌘ Web log analysis for site design:

- ☑ what are popular pages,
- ☑ what links are hard to find.

⌘ Electronic stores sales enhancements:

- ☑ recommendations, advertisement:
- ☑ **Collaborative filtering**: Net perception, Wisewire
- ☑ Inventory control: what was a shopper looking for and could not find..

OLAP Mining integration

⌘ OLAP (On Line Analytical Processing)

- ☑ Fast interactive exploration of multidim. aggregates.

- ☑ Heavy reliance on manual operations for analysis:

- ☑ Tedious and error-prone on large multidimensional data

- ⌘ Ideal platform for vertical integration of mining but needs to be interactive instead of batch.

State of art in mining OLAP integration

- ⌘ Decision trees [**Information discovery**, Cognos]
 - ☑ find factors influencing high profits
- ⌘ Clustering [Pilot software]
 - ☑ segment customers to define hierarchy on that dimension
- ⌘ Time series analysis: [Seagate's Holos]
 - ☑ Query for various shapes along time: eg. spikes, outliers
- ⌘ Multi-level Associations [Han et al.]
 - ☑ find association between members of dimensions
- ⌘ Sarawagi [VLDB2000]

Data Mining in Use



- ⌘ The US Government uses Data Mining to track fraud
- ⌘ A Supermarket becomes an information broker
- ⌘ Basketball teams use it to track game strategy
- ⌘ Cross Selling
- ⌘ Target Marketing
- ⌘ Holding on to Good Customers
- ⌘ Weeding out Bad Customers

Some success stories

- ⌘ Network intrusion detection using a combination of sequential rule discovery and classification tree on 4 GB DARPA data
 - ⊞ Won over (manual) knowledge engineering approach
 - ⊞ <http://www.cs.columbia.edu/~sal/JAM/PROJECT/> provides good detailed description of the entire process
- ⌘ Major US bank: customer attrition prediction
 - ⊞ First segment customers based on financial behavior: found 3 segments
 - ⊞ Build attrition models for each of the 3 segments
 - ⊞ 40-50% of attritions were predicted == factor of 18 increase
- ⌘ Targeted credit marketing: major US banks
 - ⊞ find customer segments based on 13 months credit balances
 - ⊞ build another response model based on surveys
 - ⊞ increased response 4 times -- 2%