# 15 Introduction to Data Mining

15.1 Introduction to principle methods

15.2 Mining association rule

see also: A. Kemper, Chap. 17.4 , Kifer et al.: chap 17.7 ff

## 15.1 Introduction

"**Discovery** of useful, possibly unexpected **patterns in data**"
J. Ullman

"Data mining is the **process of discovering** meaningful new **correlations, patterns and trends** by sifting through **large amounts of data** stored in repositories, using pattern recognition technologies as well as statistical and mathematical techniques."
The Gartner Group



## Introduction

Example

- Large amount of data
  Assumption: 1 phone call per day per adults in Germany
  ⇒ 50 Mill / day, **15 Billion phone calls /year**
- Find "**hidden knowledge**" – e.g. correlations between attributes
  "Phone calls in the evening are longer than during the day"
  ... a DWH task!
  "Young people in the age group 20-25 with a new (< 6 month) mobile phone contract living in big cities make phone calls in the evening longer than on average".
  *Might be very interesting in order to optimize tariffs.*



## Examples

More examples

Personalized recommendations
- Recommend products to customers which 'similar' customers like (have purchased, ....)
- Data: all purchases, all shopping carts, click streams

Degree of credit worthiness
- Analyze potential borrower: income? profession? where living?...
- Data: borrowers in the past and their *known* worthiness

Estimate scholastic success
- Predict success of studies based on grades, social background, ...
- Data: historic data from Campus MGM ;)

**Typical: prediction of the future *for individuals* based on *history of others***



## Methods

All kinds of **statistical techniques**
- simple counting
- estimation of probabilities
- finding parameters of distributions, ....

**Challenge for DB technology**:
scalable algorithms for **very large data sets**

New challenges: **Real Time Data Mining** on streams



## (Un) Supervised learning

**Principle methods**

**(1) Learn a model** (supervised learning)
from a large set of retrospective data)

Very different models, e.g.
Association rules, parameters of probability distributions, decision trees, classes of "homogeneous" objects

**(2) Cluster data** (unsupervised learning)
group data according to some similarity criterion

⇒ **Machine Learning**

## Classification

**Methods of supervised Learning**

**Classification: the general problem**

A **training set** of classified records is available.
Infer a model, which predicts the class of a new record

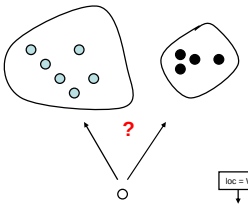
**Predict** attribute value x=c according to model F:
F(a1,...,an) = c

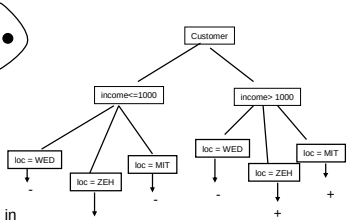
May sometimes be written as **classification rule** :

(age<40) ∧ (sex =`m´) ∧ (make=`Golf GTI´) ∧ (hp > 150)
⇨ (risk=´high´)



## Classification (2)




k=2 classes, **features** learned in training phase.

New objects assigned to according to their features.

Classification **by decision tree model** – which has to be learned.
Model: the decision tree



## Classification (3)

**Naïve Bayes Classification**

Suppose you are going to meet a person named *Gerit Robben* at the Amsterdam airport. Unfortunately you do not know if *Gerit* is a male or female first name...

Would like to know:
p(female | Gerit)

Probability, that a male is named Gerit

$$p(\text{female} \mid Gerit) = \frac{p(Gerit \mid \text{female})\, p(\text{female})}{p(Gerit)}$$

Probability of being named *Gerit*

Probability of being a female

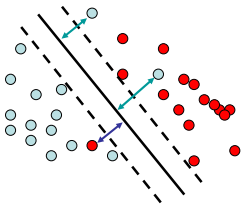
**Probabilistic Modeling**



## Classification (4)

**Support vector Machine (SVM)**

Find a hyperplane which separates classes in an optimal way

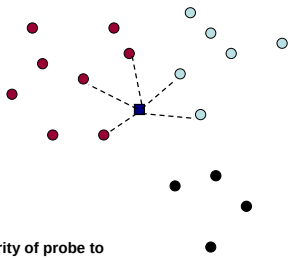


**Optimization problem**
**Model: ~ geometry**



## Classification (5)

k Nearest Neighbors



**Calculation of similarity of probe to its (classified!) neighbors.**
**Model: the classes**

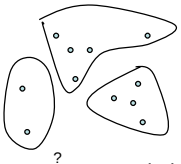


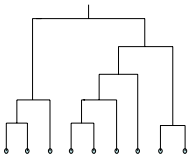
## Unsupervised Learning (1)

**Unsupervised learning methods**
**Clustering**

Group homogenous data into a cluster according to some similarity measure
e.g. find subsets of customers with similar shopping patterns

hierarchical / taxonomic



needed: **similarity measure**

## Similarity is …

… hard to define. "*We know it when we see it*"

form E. Keogh



## Unsupervised Learning(2)

**Association rules**

**Market basket analysis**:
customer transaction data: `tid, time, {articles}`
Find rules X ⇨Y , with particular confidence
e.g.
Those buying sauce, meat and spaghetti
buy red wine with high confidence

(whatever that may be)

Naiv algorithm: count how many spaghetti buyers also bought red wine



## Mining Process

**The Data mining process**

1. **Data gathering, joining, reformatting**
   e.g. Oracle: max 1000 attributes ⇨ transform into "transactional format": (id, attr_name, value)

2. **Data cleansing**
   - eliminate outliers
   - check correctness based on domain specific heuristics
   - check values in case of redundancy, ...
3. **Build model** (training phase).
4. **Apply** to [new] data

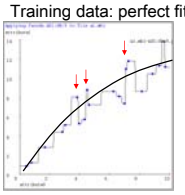


## Errors

**Training set error**
Check with records of training set if predicted value equals known value in record

Overfitting: Results are influenced by unimportant details, fit perfect to training set, but spoiled by noise.

Training data: perfect fit, but..





## 15.2 Association rules

**Applications**

Discover patterns of co-occuring products in a shopping basket

"How many customers by a printer together with a PC"

"Shopping basket" is just a metaphor...



## Association rules: examples

Baskets = **Web pages**; items = **words.**
Unusual words appearing together in a large number of documents, e.g., "British Petrol" and "Gulf" may indicate an interesting relationship.

Baskets = **Documents**, items = **sentences**
If number of co-occurrences of sentences is high: suspicion of plagiarism.

Common term for baskets, documents, or whatever:
**transaction**
Only slightly related to DB transactions in some, not all applications.

## Scale

**Problem scale** very different:

Kaufhof: ~ 100,000 products,
~ n*$10^9$ purchase transactions

Words in Web pages: ~ $10^6$
Web pages: $10^{11}$ ?



## Association rules

**The abstract problem**

Given a **set of objects** (items) $I = \{i_1,...,i_n\}$
and a **bag T** (multiset) **of non-empty subsets t** of I,
the transactions.

Find **association rules** $R_j$ : $\{i_{j1},...i_{jk}\} \rightarrow \{i_{j(k+1)},i_{k(k+l)}\}$
*Semantics:* if $i_{j1},...i_{jk} \in \mathbf{t}$, $\mathbf{t} \in \mathbf{T}$, then $\{i_{j(k+1)},i_{k(k+l)}\} \subseteq \mathbf{t}$
with some predefined probability.

But: **Probability not sufficient** - {a,b,c} occurs once in **T**,
and {a,b} is a subset of exactly two subsets of I:
$\Rightarrow$ Probability of $c \in \mathbf{t}$ if $\{a,b\} \subseteq \mathbf{t}$ is 1/2



## Association rules

**Measures:**

**Def: support** ( $A \rightarrow B$) = P(A,B), $A,B \subseteq I$
probability that A and B co-occur in the data set $\mathbf{t} \in \mathbf{T}$
e.g. 0.05 if 5 % of all customers bought a printer and a PC

**Def.: confidence** ( $A \rightarrow B$) = P (B | A)
fraction of transaction **t** containing B if **t** contains A,
e.g. 0.8: 4 of 5 bought also printer if they bought a PC

Find all rules r: $A \rightarrow B$ with
support (r) >= **minSupport** and
confidence(r) >= **minConfidence**



## Counting

**Base task** is **counting**!

r: $A \rightarrow B$ has support support (r) if there are
support(r) * |**T**| subsets **t** and $A \cup B \subseteq \mathbf{t}$

Example

| | |
|---|---|
| $t_1$ = {m, c, b} | $t_2$ = {m, p, j} |
| $t_3$ = {m, b} | $t_4$ = {c, j} |
| $t_5$ = {m, p, b} | $t_6$ = {m, c, b, j} |
| $t_7$ = {c, b, j} | $t_8$ = {b, c} |

Association rule r: {m, b} → {c}.
support(r) = P({m,b,c} $\subseteq$ t) = 1/4
confidence(r) = P({c} |{m,b}) = 2/4



## Counting

(2) **Given a frequent item set** F = {i1,..ik},
then for each $L \subset F$
r : $L \rightarrow F \setminus L$
is a rule with
**confidence (r) = support F / support L**

**since**
confidence ($L \rightarrow F \setminus L$) =P(F\L | L)
= P($F \subseteq t \wedge L \subseteq t$) / P($L \subseteq t$) = P(F) / P(L)
= support(F) / support(L)

Task: **Find frequent item sets**
... a trivial counting task??



## Counting...

Not so easy....

Suppose 1000 items.
Naïve approach: count all possible subsets...how many?
$\mathbf{2^{1000}-1}$

**Wanted:**
Clever strategies for finding frequent item sets (fis)
with 1, 2, 3, ... items from a **large number of transactions**.

## A Priori approach

Use a priori knowledge about frequent items.

**Monotony property**:
*each subset of a frequent item set is frequent*,
i.e. if $X \subseteq I$ is a frequent item set,
i. e. support(X) > minSupport
then $\forall X' \subseteq X$: support (X') > minSupport

Idea for **algorithm** to determine all Frequent Item Sets (FIS):
(1) Find FIS {F: |F| = m} having |F| > minSupport
(2) Generate candidates F'= F ∪ {i} ,
i∈I and support ({i}) > minsupport.
(3) Check each candidate with m+1 elements if frequent.



## A Priori Algorithm

```
for all items p {
  if p occurs more than minSupport make
  frequent item set with one element: F1^p = {p}  }
k = 1

repeat {
     for each Fk with k elements generate candidates
  Fk+1 with k+1 elements and Fk  ⊆ Fk+1.
    check in database, which candidates occur at least
  minSupport times;   (sequential scan of DB)
    k = k+1 }
until no new frequent item set found
```



## A Priori algorithm for finding associations

| Transactions | |
|---|---|
| TransID | Product |
| 111 | printer |
| 111 | paper |
| 111 | PC |
| 111 | toner |
| 222 | PC |
| 222 | scanner |
| 333 | printer |
| 333 | paper |
| 333 | toner |
| 444 | printer |
| 444 | PC |
| 555 | printer |
| 555 | paper |
| 555 | PC |
| 555 | scanner |
| 555 | toner |

5 transactions only!

**minSupport := 3/5 ; minCount:= minSupport*|T| = 3**
**confidence = 0.7**

```
Select product, count(*) from Transactions
group by product
having count(*) > minSupport;
```

k=1

| product | count(*) | |
|---|---|---|
| printer | 4 | |
| paper | 3 | |
| PC | 4 | |
| toner | 3 | |
| scanner | 2 | X |

example adapted from Kemper



## Priori-Algorithm

| Transactions | |
|---|---|
| TransID | Product |
| 111 | printer |
| 111 | paper |
| 111 | PC |
| 111 | toner |
| 222 | PC |
| 222 | scanner |
| 333 | printer |
| 333 | paper |
| 333 | toner |
| 444 | printer |
| 444 | PC |
| 555 | printer |
| 555 | paper |
| 555 | PC |
| 555 | scanner |
| 555 | toner |

| k=2 | | |
|---|---|---|
| FI-candidat | count | |
| {printer, paper} | 3 | |
| {printer, PC} | 3 | |
| {printer, scanner} | | |
| {printer, toner} | 3 | |
| {paper, PC} | 2 | X |
| {paper, toner} | 3 | |
| {paper, scanner} | | |
| {PC, scanner} | | |
| {PC,toner} | 2 | X |
| {scanner, toner} | | |
| {printer, paper, PC} | 2 | X |
| {printer, paper, toner} | 3 | |
| {printer, PC, toner} | 2 | X |
| {paper, PC, toner} | 2 | X |

k=3

## Generate association rules

**Given: set of FI of frequent items**

```
for each FI with support >= minSupport:
  { for each subset L  ⊂ FI
       define rule R :  L → FI  \  L
        confidence (R) = support FI / support L
        if confidence(R) >= minConfidence: keep L
    }
```

Example:
FI = {printer, paper, toner}
SupportCount = 3
Rule: {printer} ⇨ {paper, toner},
Confidence = SupportCount({printer, paper, toner}) / SupportCount({printer})
= (3/5) / (4/5)
= ¾ = 0.75

example adapted from Kemper



## Increase of confidence

**Increase of left hand side** ( i.e. decrease of right hand side) **of a rule increases confidence:**

$L \subset L^+$ , $R^- \subset R$ and $F = L \cup R = L^+ \cup R^-$

$\Rightarrow$ confidence($L \rightarrow R$) ≤ confidence($L^+ \rightarrow R^-$)

Rule: {printer} ⇨ {paper, toner}
confidence = support({printer, paper, toner}) / support({printer})
= (3/5) / (4/5)
= ¾ = 75%
Rule: {printer,paper} ⇨ {toner}
confidence = S({printer, paper, Toner}) / S({printer,paper})
= (3/5) / (3/5)
= 1 = 100%

example adapted from Kemper

## Summary data mining

Freie Universität Berlin

- Important **statistical technique**
- Basis algorithms from machine learning
- Many different methods and algorithms
- **Supervised** versus **unsupervised** learning
- Efficient implementation on **very large data sets** essential
- Enormous **commercial interest** (business transactions, web logs, ....)