Lecture 13
Data Mining for Customer Relationship Management

Zhou Shuigeng

June 4, 2006
Business Intelligence

- Business Intelligence is a process of turning data into knowledge and knowledge into action for business gain.
  - Data Warehouse Institute
The Purpose for Business Intelligence

- To support decision making at all levels of business management based on the facts and (scientific) predictions of current and future business situations that are obtained from intelligent analysis of historical business data.

- Business decisions made with BI support are more
  - Correct
  - Accurate
  - Objective
  - Timely
What to Do to Enable Business Intelligence?

- Data integration
  - Integrate data to support front end operations such as CRM
  - Integrate data to support business analytics

- Business analysis
  - Customer segmentation
  - Customer profiling
  - Customer classification
  - Churn prediction
  - Risk scoring
  - Customer life time value
  - Channel analysis
  - Supplier evaluation
  - Sales forecasting
  - Partner contribution evaluation
  - Clickstream analysis
  - Personalization and recommendation
Core Technologies for Business Intelligence

- Data warehousing
  - Store and manage integrated customer data
- OLAP
  - Interactive data queries
  - Reporting
  - Ad hoc reporting
- Data mining
  - Insights into the data
  - Discovery of correlations and patterns
  - Segmentation and classification
  - Prediction and forecasting
Business Intelligence Architecture

Business Intelligence Applications

Decision Support Tools
- Query and Reporting
- OLAP
- Information Mining

Access enablers
- Application Interfaces
- Middleware Services

Data Management
- Global Warehouse
- Departmental warehouses (datamarts)
- Other Information Stores

Warehouse modeling and Construction Tools

Operational and External Data

Administration

Metadata Management
## BI Applications by Industry

<table>
<thead>
<tr>
<th>Analytic Applications</th>
<th>Banking</th>
<th>Finance</th>
<th>Insur.</th>
<th>Retail</th>
<th>Telco</th>
<th>Mfg</th>
<th>Govt.</th>
<th>T&amp;T</th>
<th>Products</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRM-analytics: segmentation, cross sell</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>DO, ME, S</td>
</tr>
<tr>
<td>Customer Churn</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>DO, ME, S</td>
</tr>
<tr>
<td>CRM direct mail</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>DO, ME, S</td>
</tr>
<tr>
<td>Click Stream analysis</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>D, ME</td>
</tr>
<tr>
<td>Materials Mgmt &amp; Logistics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Demand Planning</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Pricing Analytics</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Procurement Analytics</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Risk Mgmt</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>D, O, ME, S</td>
</tr>
<tr>
<td>Warranty &amp; maintenance</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>D, O, ME</td>
</tr>
<tr>
<td>Supplier scorecards</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D, O, E</td>
</tr>
<tr>
<td>Bus. Perf. Monitoring</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D, O, E</td>
</tr>
<tr>
<td>Budget planning</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td>D, O, E</td>
</tr>
<tr>
<td>Profit analysis</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Workforce analysis</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>D, O, E</td>
</tr>
<tr>
<td>Fraud Detection</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>D, O, ME, S</td>
</tr>
<tr>
<td>Billing analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>D, E</td>
</tr>
<tr>
<td>Network capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

---

D = DB2; O = DB2 OLAP Server; M = Intelligent Miner; E = DB2 Warehouse Manager/DataStage; S = Spatial
## Top BI Implementations by Industry

<table>
<thead>
<tr>
<th>Analytic Applications</th>
<th>Bank/Fin</th>
<th>Insur</th>
<th>Retail/CPG</th>
<th>Telco</th>
<th>Mfg</th>
<th>Govt</th>
<th>Products</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer Loyalty &amp; Churn</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>D,O,M,E,S</td>
</tr>
<tr>
<td>CRM direct mail</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>D,O,M,E,S</td>
</tr>
<tr>
<td>Click Stream analysis</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>D,M,E</td>
</tr>
<tr>
<td>Demand Planning</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>D,O,E</td>
</tr>
<tr>
<td>Network Capacity</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td>D,O,E</td>
</tr>
<tr>
<td>Billing analysis</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D,E</td>
</tr>
<tr>
<td>Procurement Analytics</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D,O,E</td>
</tr>
<tr>
<td>Supplier scorecards</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D,O,E</td>
</tr>
<tr>
<td>Risk Mgmt</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>D,O,M,E,S</td>
</tr>
<tr>
<td>Bus. Perf. Monitoring</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D,O,E</td>
</tr>
<tr>
<td>Budget planning</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>D,O,E</td>
</tr>
<tr>
<td>Profit analysis</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>D,O,E</td>
</tr>
<tr>
<td>Homeland Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>D,M,E,S</td>
</tr>
</tbody>
</table>

D = DB2; O = DB2 OLAP Server; M = Intelligent Miner; E = DB2 Warehouse Manager/DataStage; S = Spatial
Customer Relationship Management

Customer Relationship Management: focus on customer satisfaction to improve profit

Two kinds of CRM

- **Enabling CRM**: Infrastructure, multiple touch point management, data integration and management, ...
  - Oracle, IBM, PeopleSoft, Siebel Systems, SAS...
- **Intelligent CRM**: data mining and analysis, customer marketing, customization, employee analysis
  - Vendors/products (see later)
  - Services!!
The Business Problem: Marketing

- **Improve** customer relationship
  - Actions (promotion, communication) ⇒ changes
- What **actions** should your Enterprise take to change your customers from an undesired status to a desired one
  - How to cross-sell?
  - How to segment customer base?
  - How to formulate direct marketing plans?
- Data mining can help!
## Data Mining Software

<table>
<thead>
<tr>
<th>Data mining tools you regularly use: [495 votes, 858 tools]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clementine (156)</td>
</tr>
<tr>
<td>SPSS/AnswerTree (135)</td>
</tr>
<tr>
<td>SAS (104)</td>
</tr>
<tr>
<td>CART/MARS (97)</td>
</tr>
<tr>
<td>SAS EM (55)</td>
</tr>
<tr>
<td>Megaputer (52)</td>
</tr>
<tr>
<td>MATLAB (45)</td>
</tr>
<tr>
<td>Angoss (29)</td>
</tr>
<tr>
<td>IBM I-Miner (29)</td>
</tr>
<tr>
<td>Statistica (16)</td>
</tr>
<tr>
<td>Oracle Darwin (14)</td>
</tr>
<tr>
<td>SGI Mineset (14)</td>
</tr>
<tr>
<td>Model 1 (10)</td>
</tr>
<tr>
<td>Gainsmarts (6)</td>
</tr>
<tr>
<td>Xaffinity (3)</td>
</tr>
<tr>
<td>Other (93)</td>
</tr>
</tbody>
</table>

- Clementine: 18%
- SPSS/AnswerTree: 16%
- SAS: 12%
- CART/MARS: 11%
- SAS EM: 6%
- Megaputer: 6%
- MATLAB: 5%
- Angoss: 3%
- IBM I-Miner: 3%
- Statistica: 2%
- Oracle Darwin: 2%
- SGI Mineset: 2%
- Model 1: 1%
- Gainsmarts: 1%
- Xaffinity: 0%
- Other: 11%
Direct Marketing

Two approaches to promotion:

Mass marketing
Use mass media to broadcast message to the public without discrimination.
Become less effective due to low response rate.

Direct marketing

Direct Marketing

Direct marketing
A process of identifying likely buyers of certain products and promoting the products accordingly
Studies customers’ characteristics.
selects certain customers as the target.

Data mining—provide an effective tool for direct marketing
Case Study 1: Attrition/Churn In the Mobile Phone Industry

- Each year, an average of 27% of the customers churn in the US.
  - Overtime, 90% of the customers in cell phone industry have churned at least once in every five year period.
  - It takes $300 to $600 to acquire a new customer in this industry.

- Example

<table>
<thead>
<tr>
<th>Avg Bill (1995-96)</th>
<th>Churn Rate</th>
<th># customers</th>
<th>Acquire New Customer</th>
</tr>
</thead>
<tbody>
<tr>
<td>$52/month</td>
<td>25%/year</td>
<td>1,000,000</td>
<td>$400</td>
</tr>
</tbody>
</table>

Thus, if we reduce churn by 5%, then we can save Company $5,000,000.00 per year!
Lightbridge

Lightbridge, a US mobile-phone company, applied the CART algorithm to their customer database
  to identify a segment of their customer base that held 10% of the customers,
  but with a 50% churn rate.

This segment is highly predictive in terms of customer attrition.
  This segment is then said to have a lift of five.
The Lightbridge Experience

From the CART Tree, it is found

- Subscribers who call customer service are more loyal to the company, and are less likely to churn!
- The first year anniversary seems to be a very vulnerable time for the customers.
- After the customers enter the second year, they do not churn.
Case Study 2: A UK Telecom

- A UK Company,
  - the CART model was applied to 260,000 customers
  - to study why the churn rate (40%) is so high.
- Method
  - Using March 1998 data as training set, and
  - April 1998 data as test set
- CART generated 29 segments.
Case Study 3: A Bank in Canada

- The bank wants to sell a mutual fund
  - Database contains two types of customers
  - After a Mass Marketing Campaign,
    - Group1 bought the fund
    - Group2 has not bought the fund
  - Often, Group1 << Group2
    - Group1 is usually 1%
- Question: what are the patterns of group1?
- How to select a subgroup from Group 2, such that they are likely to buy the mutual fund?
WorkFlow of Case 3:

1. Get the database of customers (1%)
2. Data cleaning: transform address and area codes, deal with missing values, etc.
3. Split database into training set and testing set
4. Applying data mining algorithms to the training set
5. Evaluate the patterns found in the testing set
6. Use the patterns found to predict likely buyers among the current non-buyers
7. Promote to likely buyers (rollout plan)
Specific problems

Extremely imbalanced class distribution
E.g. only 1% are positive (buyers), and the rest are negative (non-buyers).

Evaluation criterion for data mining process
The predictive accuracy is no longer suitable.
The training set with a large number of variables can be too large.
 Efficient learning algorithm is required.
Solutions

- **Rank** training and testing examples
  - We require learning algorithms to produce probability estimation or confidence factor.

- Use **lift** as the evaluation criterion
  - A lift reflects the redistribution of responders in the testing set after ranking the testing examples.
Solution I: Learning algorithms

- Naïve Bayes algorithm
  - Can produce probability in order to rank the testing examples.
  - Has efficient and good performance

- Decision tree with certainty factor (CF)
  - Modify C4.5 to produce CF.
  - \[ CF = \frac{\text{the number of examples of the majority class}}{\text{the total number of examples in that leaf}} \]
Solution I : Learning algorithms (cont.)

- Ada-boosting
  1. Initialize different weights across the training set to be uniform.
  2. Select a training set by sampling according to these weights and train component classifier $C_k$.
  3. Increase weights of patterns misclassified and decrease weights of patterns correctly classified by $C_k$.
  4. $k \leftarrow k + 1$, then skip to 2
Solution II: lift index for evaluation

- A typical lift table

<table>
<thead>
<tr>
<th></th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
<th>10%</th>
</tr>
</thead>
<tbody>
<tr>
<td>410</td>
<td>190</td>
<td>130</td>
<td>76</td>
<td>42</td>
<td>32</td>
<td>35</td>
<td>30</td>
<td>29</td>
<td>26</td>
<td></td>
</tr>
</tbody>
</table>

- Use a weighted sum of the items in the lift table over the total sum — lift index

  - **Definition**
    
    \[ S_{\text{lift}} = \frac{(1 \times S_1 + 0.9 \times S_2 + \ldots + 0.1 \times S_{10})}{\sum S_i} \]

  - **E.g.**
    \[ S_{\text{lift}} = \frac{(1 \times 410 + 0.9 \times 190 + \ldots + 0.1 \times 2.6)}{1000} = 81.1\% \]
Solution II: lift index for evaluation (cont.)

- Lift index is independent to the number of the responders
  - 50% for random distribution
  - Above 50% for better than random distribution
  - below 50% for worse than random distribution
Solutions: summary

- Two algorithms:
  - Ada-boosted Naïve Bayes
  - Ada-boosted C4.5 with CF

- Three datasets:
  - Bank
  - Life insurance
  - Bonus program

- Training and testing set with equal size
Solutions: summary (cont.)

Procedure:
- Training → learned results
- Rank the testing examples
- Calculate lift index and compare classifiers
- Repeat 10 times for each dataset to obtain an average lift index
Results

Average lift index on three datasets using boosted Naïve Bayes

<table>
<thead>
<tr>
<th>Positive/negative</th>
<th>Bank</th>
<th>Life Insurance</th>
<th>Bonus Program</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/0.25</td>
<td>66.4%</td>
<td>72.3%</td>
<td>78.7%</td>
</tr>
<tr>
<td>1/0.5</td>
<td>68.8 %</td>
<td>74.3%</td>
<td>80.3%</td>
</tr>
<tr>
<td>1/1</td>
<td>70.5 %</td>
<td>75.2%</td>
<td>81.3%</td>
</tr>
<tr>
<td>1/2</td>
<td>70.4 %</td>
<td>75.3%</td>
<td>81.2%</td>
</tr>
<tr>
<td>1/4</td>
<td>69.4%</td>
<td>75.4%</td>
<td>81.0%</td>
</tr>
<tr>
<td>1/8</td>
<td>69.1 %</td>
<td>75.5%</td>
<td>80.4%</td>
</tr>
</tbody>
</table>
### Comparison of Results

<table>
<thead>
<tr>
<th></th>
<th>Mass mailing</th>
<th>Direct mailing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of customers mailed</td>
<td>600,000</td>
<td>(20%)120,000</td>
</tr>
<tr>
<td>Cost of mailing ($0.71 each)</td>
<td>$426,000</td>
<td>$85,200</td>
</tr>
<tr>
<td>Cost of data mining</td>
<td>$0</td>
<td>$40,000</td>
</tr>
<tr>
<td>Total promotion cost</td>
<td>$426,000</td>
<td>$125,200</td>
</tr>
<tr>
<td>Response rate</td>
<td>1.0%</td>
<td>3.0%</td>
</tr>
<tr>
<td>Number of sales</td>
<td>6,000</td>
<td>3,600</td>
</tr>
<tr>
<td>Profit from sale ($70 each)</td>
<td>$420,000</td>
<td>$252,000</td>
</tr>
<tr>
<td>Net profit from promotion</td>
<td>-$6,000</td>
<td>$126,800</td>
</tr>
</tbody>
</table>

- The mailing cost is reduced
- But the response rate is improved.
- The net profit is increased dramatically.
Results (cont.)

- Net profit in direct marketing
Improvement

Probability estimation model
Rank customers by the estimated probability of response and mail to some top of the list.

Drawback of probability model
The actual value of individual customers is ignored in the ranking.
An inverse correlation between the likelihood to buy and the dollar amount to spend.
The goal of direct marketing
To maximize
\[ \sum (\text{actual profit} - \text{mailing cost}) \]
over the contacted customers
Idea: Push algorithm
probability estimation \rightarrow \text{profit estimation}

Challenges

The inverse correlation often occurs. Most probable to buy ≠ most money to spend
The high dimensionality of the dataset. “Transparent” prediction model is desirable.
Wish to devise campaign strategies based on the characteristics of generous expense.
Case Study 4: Direct Marketing for Charity in USA

- KDD Cup 98 Dataset
- 191,779 records in the database
- Each record is described by 479 non-target variables and two target variables
  - The class: “response” or “not response”
  - The actual donation in dollars
- The dataset was split in half, one for training and one for validation
Push Algorithm: Wang, Zhou et al
ICDE 2003

**Input**: the learning dataset

**Methods**
- **Step1**: rule generating
- **Step2**: model building
- **Step3**: model pruning

**Output**: A model for predicting the donation amount

The algorithm outline
Step 1: Rule Generating

Objective: To find all Focused association rules that captures features of responders.

\[ A_{i_1} = a_{i_1}, \ldots, A_{i_k} = a_{i_k} \rightarrow C = \text{respond} \]

**FAR**: a respond_rule that satisfies specified minimum R-support and maximum N-support.

**R-support of a respond_rule** is the percentage of the respond records that contain both sides of the rule.

**N-support of a respond_rule** is the largest N-support of the data items in the rule.

N-support of a data item is the percentage of the records in non-respond records.
Step 2: Model building

- Compute *Observed average profit*
  
  \[ O_{\text{avg}}(r) = \frac{\sum profit(r, t)}{N} \]
  
  for each rule.

- Build prediction model: assign the prediction rule with the largest possible \( O_{\text{avg}} \) to each customer record.

Given a record \( t \), a rule \( r \) is the prediction rule of \( t \) if \( r \) matches \( t \) and has the highest possible rank.
Step3: The model pruning

- Build up prediction tree based on prediction rules.
- Simplify the tree by pruning overfitting rules that do not generalize to the whole population.
The prediction

- The customer will be contacted if and only if $r$ is a respond_rule and

\[ E_{\text{avg}}(r) > 0 \]

Where $E_{\text{avg}}(r)$ is the estimated average profit.
Validation

Comparison with the top 5 contestants of the KDD-CUP-98

<table>
<thead>
<tr>
<th>Participants</th>
<th>Sum of actual profit</th>
<th># mailed</th>
<th>Average profit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Our method</strong></td>
<td>$24,621.00</td>
<td>27,550</td>
<td>0.89</td>
</tr>
<tr>
<td>GainSmarts (winner)</td>
<td>$14,712.24</td>
<td>56,330</td>
<td>0.26</td>
</tr>
<tr>
<td>SAS/Enterprise Miner</td>
<td>$14,662.43</td>
<td>55,838</td>
<td>0.26</td>
</tr>
<tr>
<td>Quadstone</td>
<td>$13,954.47</td>
<td>57,836</td>
<td>0.24</td>
</tr>
<tr>
<td>ARIA CARRL</td>
<td>$13,824.77</td>
<td>55,650</td>
<td>0.25</td>
</tr>
<tr>
<td>Amodocs/KDD Suite</td>
<td>$13,794.24</td>
<td>51,906</td>
<td>0.27</td>
</tr>
</tbody>
</table>

The approach generates 67% more total profit and 242% more average profit per mail than the winner of the competition.
Cross-Selling with Collaborative Filtering

Instructor: Zhou Shuigeng

May 3, 2004
Motivation

Question:
- A user bought some products already
- what *other products* to recommend to a user?
- Collaborative Filtering (CF)
  - Automates “circle of advisors”.
Collaborative Filtering

“..people collaborate to help one another perform filtering by recording their reactions...” (Tapestry)

- Finds users whose taste is similar to you and uses them to make recommendations.
- Complimentary to IR/IF.
  - IR/IF finds similar documents - CF finds similar users.
Example

Which movie would Sammy watch next?

Ratings 1--5

- If we just use the average of other users who voted on these movies, then we get
  - \( \text{Matrix} = 3; \quad \text{Titanic} = 14/4 = 3.5 \)
  - Recommend Titanic!
  - But, is this reasonable?
Types of Collaborative Filtering Algorithms

- **Collaborative Filters**
  - Statistical Collaborative Filters
  - Probabilistic Collaborative Filters [PHLO00]
  - Bayesian Filters [BP99][BHK98]
  - Association Rules [Agrawal, Han]

- **Open Problems**
  - Sparsity, First Rater, Scalability
### Statistical Collaborative Filters

- Users annotate items with numeric ratings.
- Users who rate items “similarly” become mutual advisors.
- Recommendation computed by taking a weighted aggregate of advisor ratings.
Basic Idea

- Nearest Neighbor Algorithm
- Given a user $a$ and item $i$
  - First, find the most similar users to $a$,
    - Let these be $Y$
  - Second, find how these users ($Y$) ranked $i$,
  - Then, calculate a predicted rating of $a$ on $i$
    - based on some average of all these users $Y$
      - *How to calculate the similarity and average?*
Statistical Filters

- **GroupLens [Resnick et al 94, MIT]**
  - Filters UseNet News postings
  - Similarity: *Pearson correlation*
  - Prediction: Weighted deviation from mean

\[
P_{a,i} = \bar{r}_a + \frac{1}{\alpha} \sum (r_{u,i} - \bar{r}_u) \cdot w_{a,u}
\]
Pearson Correlation

### Pearson Correlation

<table>
<thead>
<tr>
<th>User</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>C</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>

![Graph showing Pearson Correlation](image-url)

- **Items**: Item 1, Item 2, Item 3, Item 4, Item 5
- **Rating** range from 0 to 7
- **Users**: User A, User B, User C

![User A](image-url)
![User B](image-url)
![User C](image-url)
**Pearson Correlation**

- **Weight between users** $a$ and $u$
  - Compute similarity matrix between users
    - Use Pearson Correlation (-1, 0, 1)
    - Let $items$ be all items that users rated

\[
w_{a,u} = \frac{1}{|items|} \sum_{i \in items} \frac{(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in items} (r_{a,i} - \bar{r}_a)^2} \sqrt{\sum_{i \in items} (r_{u,i} - \bar{r}_u)^2}}
\]

<table>
<thead>
<tr>
<th>User</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1</td>
<td>-1</td>
</tr>
<tr>
<td>C</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
</tr>
</tbody>
</table>
**Prediction Generation**

- Predicts how much a user $a$ likes an item $i$

- Generate predictions using weighted deviation from the mean

$$P_{a,i} = \bar{r}_a + \frac{1}{\alpha} \sum (r_{u,i} - \bar{r}_u) \cdot w_{a,u}$$  \hspace{1cm} (1)

- $\alpha$: sum of all weights  
$$\alpha = \sum_{Y_{a,u}} w_{a,u}$$
Error Estimation

- **Mean Absolute Error (MAE) for user \( a \)**

  \[
  MAE_a = \frac{\sum_{i=1}^{N} |P_{a,i} - r_{a,i}|}{N}
  \]

- **Standard Deviation of the errors**

  \[
  \sigma = \sqrt{\frac{\sum_{a=1}^{K} (MAE_a - \overline{MAE})^2}{K}}
  \]
Example

<table>
<thead>
<tr>
<th>Users</th>
<th>Sammy</th>
<th>Dylan</th>
<th>Mathew</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sammy</td>
<td>1</td>
<td>1</td>
<td>-0.87</td>
</tr>
<tr>
<td>Dylan</td>
<td>1</td>
<td>1</td>
<td>0.21</td>
</tr>
<tr>
<td>Mathew</td>
<td>-0.87</td>
<td>0.21</td>
<td>1</td>
</tr>
</tbody>
</table>

\[
P_{Sammy,Matrix} = \bar{r}_{Sammy} + \left\{ (r_{Dylan,Matrix} - \bar{r}_{Dylan}) \cdot w_{Sammy,Dylan} + (r_{Mathew,Matrix} - \bar{r}_{Mathew}) \cdot w_{Sammy,Mathew} \right\} \cdot \frac{1}{|w_{Sammy,Dylan}| + |w_{Sammy,Mathew}|}
\]

\[
= 3.3 + \left\{ (3 - 3.4) \cdot 1 + (2 - 3.2) \cdot (-0.87) \right\} / (1 + 0.87)
\]

\[
= 3.6
\]

<table>
<thead>
<tr>
<th>Users</th>
<th>Prediction</th>
<th>Actual</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Matrix</td>
<td>Titanic</td>
<td></td>
</tr>
<tr>
<td>Sammy</td>
<td>3.6</td>
<td>2.8</td>
<td>0.9</td>
</tr>
<tr>
<td>Basil</td>
<td>4.6</td>
<td>4.1</td>
<td>0.75</td>
</tr>
</tbody>
</table>

\[
\overline{MAE} = 0.83
\]
Statistical Collaborative Filters

- Ringo [Shardanand and Maes 95 (MIT)]
  - Recommends music albums
    - Each user buys certain music artists' CDs
    - Base case: weighted average
  - Predictions
    - Mean square difference
      - First compute dissimilarity between pairs of users
      - Then find all users Y with dissimilarity less than L
      - Compute the weighted average of ratings of these users
    - Pearson correlation (Equation 1)
    - Constrained Pearson correlation (Equation 1 with weighted average of similar users (corr > L))
Open Problems in CF

- "Sparsity Problem"
  - CFs have poor accuracy and coverage in comparison to population averages at low rating density [GSK+99].

- "First Rater Problem"
  - The first person to rate an item receives no benefit. CF depends upon altruism. [AZ97]
Open Problems in CF

“Scalability Problem”

- CF is computationally expensive. Fastest published algorithms (nearest-neighbor) are $n^2$.
  - Any indexing method for speeding up?
- Has received relatively little attention.

References in CF:

Mining the network value of customers

by P. Domingos and M. Richardson
from KDD2002
Motivation

- Network value is **ignored** (Direct marketing).
- Examples:
  
  **Market to**
  
  - Low expected profit
  
  **Affected** (under the network effect)
  
  - High expected profit
  
  - High expected profit
Some Successful Case

- **Hotmail**
  - Grew from 0 to 12 million users in 18 months
  - Each email include a promotional URL of it.

- **ICQ**
  - Expand quickly
  - First appear, user addicted to it
  - Depend it to contact with friend
Introduction

- Incorporate the network value in maximizing the expected profit.
- Social networks: modeled by the Markov random field
- Probability to buy = Desirability of the item + Influence from others
- Goal = maximize the expected profit
Focus

- Making use of network value **practically** in recommendation

- Although the algorithm may be used in other applications, the focus is **NOT** a generic algorithm
Assumption

- Customer (buying) decision can be affected by other customer’s rating

- Market to people who is inclined to see the film

- One will not continue to use the system if he did not find its recommendations useful (natural elimination assumption)
Modeling

- View the markets as Social Networks
- Model the Social Network as Markov random field
- What is Markov random field?
  - An experiment with outcomes being functions of more than one continuous variable. [e.g. \( P(x,y,z) \)]
  - The outcome depends on the neighbors'.
Variable definition

- $X = \{X_1, \ldots, X_n\}$: a set of $n$ potential customer, $X_i = 1$ (buy), $X_i = 0$ (not buy)
- $X^k$ (known value), $X^u$ (unknown value)
- $N_i = \{X_{i,1}, \ldots, X_{i,n}\}$: neighbor of $X_i$
- $Y = \{Y_1, \ldots, Y_m\}$: a set of attribute to describe the product
- $M = \{M_1, \ldots, M_n\}$: a set of market action to each customer
Example (set of $Y$)

- Using EachMovie as example.
- $X_i$: Whether the person $i$ saw the movie?
- $Y$: The movie genre
- $R_i$: Rating to the movie by person $i$

It sets $Y$ as the movie genre,
- different problems can set different $Y$. 
Goal of modeling

- To find the market action ($M$) to different customer, to achieve best profit.
- Profit is called ELP (expected lift in profit)
- \[ ELP_i(X^k,Y,M) = r_1P(X_i=1|X^k,Y,f_i^1(M)) - r_0P(X_i=1|X^k,Y,f_i^0(M)) - c \]
  - $r_1$: revenue with market action
  - $r_0$: revenue without market action
Three different modeling algorithm

- Single pass
- Greedy search
- Hill-climbing search
Scenarios

- Customer {A, B, C, D}
- A: He/She will buy the product if someone suggest and discount (M=1)
- C, D: He/She will buy the product if someone suggest or discount (M=1)
- B: He/She will never buy the product

The best: M=1 M=1
Single pass

- For each $i$, set $M_i = 1$ if $\text{ELP}(X^k, Y, f_i^1(M_0)) > 0$, and set $M_i = 0$ otherwise.

- Adv: Fast algorithm, one pass only

- Disadv:
  - Some market action to the later customer may affect the previous customer
  - And they are ignored
Single Pass Example

\[ M = \{0,0,0,0\} \quad ELP(X^k, Y, f_0^1(M_0)) \leq 0 \]
\[ M = \{0,0,0,0\} \quad ELP(X^k, Y, f_1^1(M_0)) \leq 0 \]
\[ M = \{0,0,0,0\} \quad ELP(X^k, Y, f_2^1(M_0)) > 0 \]
\[ M = \{0,0,1,0\} \quad ELP(X^k, Y, f_3^1(M_0)) > 0 \]
\[ M = \{0,0,1,1\} \quad \text{Done} \]
Greedy Algorithm

- Set $M = M_0$.
- Loop through the $M_i$’s,
  - setting each $M_i$ to 1 if $\text{ELP}(\mathbf{x}_k, \mathbf{y}, f_i^1(M)) > \text{ELP}(\mathbf{x}_k, \mathbf{y}, M)$.
  - Continue until no changes in $M$.
- Adv: Later changes to the $M_i$’s will affect the previous $M_i$.
- Disadv: It takes much more computation time, several scans needed.
Greedy Example

A, B, C, D

- $M_0 = \{0,0,0,0\}$  First pass
- $M = \{0,0,1,1\}$  Second pass
- $M = \{1,0,1,1\}$  Third pass
- $M = \{1,0,1,1\}$  Done

Discount / suggest
Discount + suggest
Discount / suggest

M=1  M=1  M=1  M=1
Hill-climbing search

Set $M = M_0$. Set $M_{i_1} = 1$, where $i_1 = \arg\max_i \{ELP(X^k, Y, f_1(M))\}$.

Repeat

- Let $i = \arg\max_i \{ELP(X^k, Y, f_1(f_1(M)))\}$
- set $M_i = 1$

Until there is no $i$ for setting $M_i = 1$ with a larger ELP.

Adv:
- The best $M$ will be calculated, as each time the best $M_i$ will be selected.

Disadv: The most expensive algorithm.
Hill Climbing Example

- $M = \{0,0,0,0\}$ First pass
- $M = \{0,0,1,0\}$ Second pass
- $M = \{1,0,1,0\}$ Third pass
- $M = \{1,0,1,0\}$ Done

The best: $M=1$
Who Are the Neighbors?

- Mining Social Networks by Using Collaborative Filtering (CFinSC).
- Using Pearson correlation coefficient to calculate the similarity.
- The result in CFinSC can be used to calculate the Social networks.
- ELP and $M$ can be found by Social networks.
Who are the neighbors?

- Calculate the weight of every customer by the following equation:

\[ W_{ij} = \frac{\sum_k (R_{ik} - \bar{R}_i)(R_{jk} - \bar{R}_j)}{\sqrt{\sum_k (R_{ik} - \bar{R}_i)^2 \sum_k (R_{jk} - \bar{R}_j)^2}} \]
Neighbors’ Ratings for Product

- Calculate the Rating of the neighbor by the following equation.

\[ \hat{R}_{ik} = \bar{R}_i + \rho \sum_{X_j \in N_i} W_{ji} (R_{jk} - \bar{R}_j) \]

\[ \rho = \frac{1}{\sum_{X_j \in N_i} |W_{ij}|} \]

- If the neighbor did not rate the item, \( R_{jk} \) is set to mean of \( R_j \)
Estimating the Probabilities

- $P(X_i)$: Items rated by user $i$
- $P(Y_k|X_i)$: Obtained by counting the number of occurrences of each value of $Y_k$ with each value of $X_i$.
- $P(M_i|X_i)$: Select user in random, do market action to them, record their effect. (If data not available, using prior knowledge to judge)
Preprocessing

- Zero mean

- Prune people ratings cover too few movies (10)

- Non-zero standard deviation in ratings

- Penalize the Pearson correlation coefficient if both users rate very few movies in common

- Remove all movies which were viewed by < 1% of the people
Experiment Setup

- Data: Each movie
- Trainset & Testset (temporal effect)

![Diagram showing the timeline of data release and training/test set separation.](image-url)
Experiment Setup - cont.

- Target: 3 methods of searching an optimized marketing action VS baseline (direct marketing)

\[
\alpha = \sum_{Y_{a,u}} W_{a,u}
\]
Experiment Results

[Graphs showing experiment results for Free Movie, Advertising, Discounted Movie, and Discounted Movie Runtimes]
Proposed algorithms are much better than direct marketing.

Hill > (slight) greedy >> single-pass >> direct

Higher $\alpha$, better results!
References

Item Selection By “Hub-Authority” Profit Ranking

from ACM KDD2002

By Ke Wang, Ming-Yen Thomas Su
Ranking in Inter-related World

- Web pages
- Social networks
- Cross-selling
Item Ranking with Cross-selling Effect

What are the most profitable items?
The Hub/Authority Modeling

- **Hubs** i: “introductory” for sales of other items j (i→j).
- **Authorities** j: “necessary” for sales of other items i (i→j).
- **Solution**: model the mutual enforcement of hub and authority weights through links.
  - **Challenges**: Incorporate individual profits of items and strength of links, and ensure hub/authority weights converges.
Selecting Most Profitable Items

- **Size-constrained selection**
  - given a size $s$, find $s$ items that produce the most profit as a whole
  - solution: select the $s$ items at the top of ranking

- **Cost-constrained selection**
  - given the cost for selecting each item, find a collection of items that produce the most profit as a whole
  - solution: the same as above for uniform cost
Solution to const-constrained selection

- Estimated profit
- Selection cost

Optimal cutoff

# of items selected
Web Page Ranking Algorithm - HITS (Hyperlink-Induced Topic Search)

- Mutually reinforcing relationship
  - Hub weight: $h(i) = \sum a(j)$, for all page $j$ such that $i$ have a link to $j$
  - Authority weight: $a(i) = \sum h(j)$, for all page $j$ that have a link to $i$ $h(j)$

- $a$ and $h$ converge if normalized before each iteration
The Cross-Selling Graph

- Find frequent items and 2-itemsets
- Create a link $i \rightarrow j$ if $\text{Conf}(i \rightarrow j)$ is above a specified value ($i$ and $j$ may be same)
- “Quality” of link $i \rightarrow j$: $\text{prof}(i) \times \text{conf}(i \rightarrow j)$. Intuitively, it is the credit of $j$ due to its influence on $i$
Computing Weights in HAP

- For each iteration,
  - Authority weights: \( a(i) = \sum_{j \rightarrow i} \text{prof}(j) \times \text{conf}(j \rightarrow i) \times h(j) \)
  - Hub weights: \( h(i) = \sum_{i \rightarrow j} \text{prof}(i) \times \text{conf}(i \rightarrow j) \times a(i) \)
- Cross-selling matrix \( B \)
  - \( B[i, j] = \text{prof}(i) \times \text{conf}(i, j) \) for link \( i \rightarrow j \)
  - \( B[i, j] = 0 \) if no link \( i \rightarrow j \) (i.e. \( (i, j) \) is not frequent set)
- Compute weights iteratively or use eigen analysis
- Rank items using their authority weights
Example

Given frequent items, $X$, $Y$, and $Z$ and the table:

<table>
<thead>
<tr>
<th></th>
<th>$X$</th>
<th>$Y$</th>
<th>$Z$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X$</td>
<td>$5.0000$</td>
<td>$1.0000$</td>
<td>$4.0000$</td>
</tr>
<tr>
<td>$Y$</td>
<td>$0.0600$</td>
<td>$1.0000$</td>
<td>$0.5000$</td>
</tr>
<tr>
<td>$Z$</td>
<td>$0.0200$</td>
<td>$0.0375$</td>
<td>$0.1000$</td>
</tr>
</tbody>
</table>

We get the cross-selling matrix $B$:

- $\text{prof}(X) = $5
- $\text{conf}(X \rightarrow Y) = 0.2$
- $\text{conf}(Y \rightarrow X) = 0.06$
- $\text{prof}(Y) = $1
- $\text{conf}(X \rightarrow Z) = 0.8$
- $\text{conf}(Z \rightarrow X) = 0.2$
- $\text{prof}(Z) = $0.1
- $\text{conf}(Y \rightarrow Z) = 0.5$
- $\text{conf}(Z \rightarrow Y) = 0.375$

e.g. $B[X,Y] = \text{prof}(X) \times \text{conf}(X,Y) = 1.0000$
Example (con’t)

- \( \text{prof}(X) = $5, \text{prof}(Y) = $1, \text{prof}(Z) = $0.1 \)
- \( a(X) = 0.767, a(Y) = 0.166, a(Z) = 0.620 \)

- HAP Ranking is different from ranking the individual profit
  - The cross-selling effect increases the profitability of \( Z \)
Empirical Study

- Conduct experiments on two datasets
- Compare 3 selection methods: HAP, PROFSET [4, 5], and Naïve.
- HAP generates the highest estimated profit in most cases.
## Empirical Study

<table>
<thead>
<tr>
<th></th>
<th>Drug Store</th>
<th>Synthetic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transaction #</td>
<td>193,995</td>
<td>10,000</td>
</tr>
<tr>
<td>Item #</td>
<td>26,128</td>
<td>1,000</td>
</tr>
<tr>
<td>Avg. Trans length</td>
<td>2.86</td>
<td>10</td>
</tr>
<tr>
<td>Total profit</td>
<td>$1,006,970</td>
<td>$317,579</td>
</tr>
<tr>
<td>minsupp</td>
<td>0.1%</td>
<td>0.05%</td>
</tr>
<tr>
<td>Freq. items</td>
<td>332</td>
<td>999</td>
</tr>
<tr>
<td>Freq. pairs</td>
<td>39</td>
<td>115</td>
</tr>
</tbody>
</table>
Experiment Results

drug store dataset minsup=0.1%

estimated profit

# of selected items

HAP

PROFSET

Naïve