Have You Ever ...

- Wondered how products are placed in supermarket isles?
- Had your application for a no-interest-for-6-months Titanium credit card rejected?
- Puzzled over the two-hour phone call to Belize on your phone bill?
- Gazed at the sky and wondered if that bright star is a white dwarf?
- Data mining has the answers!!!

What is Data Mining?

- Finding “interesting” patterns in large amounts of data.
- Data mining encompasses several areas:
  - Machine learning (AI)
  - Statistics
  - Databases

Data Mining Needs Databases

- Machine learning and statistics often make the following assumptions:
  - small amount of data (or sample)
  - data fits in main memory
  - CPU time is crucial
- The reality:
  - huge amounts data
  - data on secondary storage
  - data management (disk I/O) is crucial

Data Mining Techniques

- Classification (supervised learning)
  - Build and train classifiers (decision trees, neural nets, etc.)
- Clustering (unsupervised learning)
  - Partition the data into groups with similar characteristics.
- Sequence and stream analysis
- Association rule-mining

Association-Rule Mining

- Flagship of data mining with database flavor.
- Find correlations among data without building a complete predictive or descriptive model.
- Data-centric approach.
Market Basket Data

- Consider supermarket customers.
- At the checkout each customer has a basket of items.
- Find correlation among the contents of baskets.
- The model works for many domains:
  - Online/offline shopping
  - Web surfing
  - Text analysis.

Association Rules

- Find rules of the form:
  - People who buy X tend to buy Y.

Mythical Association Rule

A Lesson in Marketing

- Suppose we know that people buy bread and milk frequently. So what?
  - Stock them together.
  - Stock them apart.
  - Run sales on one and up the price of the other.
- Amazon’s recommendations are based on association rules.
  - Order size went up 20% in the first week after recommendations were introduced.

Schema of Market Basket Data

- Several models possible depending on the application.
- Simplest, most general schema: `Baskets(basketID, item)`
- Applicable to many different scenarios, online and offline.

Market Basket Example

<table>
<thead>
<tr>
<th>basketID</th>
<th>item</th>
</tr>
</thead>
<tbody>
<tr>
<td>11111</td>
<td>beer</td>
</tr>
<tr>
<td>11111</td>
<td>chips</td>
</tr>
<tr>
<td>11111</td>
<td>salsa</td>
</tr>
<tr>
<td>22222</td>
<td>vodka</td>
</tr>
<tr>
<td>22222</td>
<td>caviar</td>
</tr>
</tbody>
</table>
Support and Confidence

- Formally, we associate two numbers with every rule:
  - support
  - confidence
- Example: Beer → Diapers
  - Support is the fraction of all baskets that contain both beer and diapers.
  - Confidence is the fraction of baskets which contain beer that also contain diapers.

Thresholds

- Find association rules with high support and high confidence.
- Typically, high support means > 0.1% and high confidence means > 50%.
- Thresholds depend on the application.

Main Challenge

- Too many item combinations:
  - 100s of thousands of items
  - millions of transactions
- Direct approach too slow:
  - 100 million baskets, 20 items/basket
  - 19 billion pairs, 100+ billion triples,...

Two-Phase Approach

- Phase 1: Find all itemsets with high support.
  - These itemsets are called frequent.
- Phase 2: Construct rules with high confidence.
  - The computational cost of phase 1 dominates the total cost.
  - Focus on finding frequent itemsets.

Find All Frequent Pairs

- Write query in SQL:

The \textit{A-Priori} Technique

- Key observation: a pair of items is frequent only if each item is frequent.
  - If \{bread, cheese\} is frequent then \{bread\} and \{cheese\} must be frequent.
- Levelwise pruning:
  - Consider \{bread, milk, cheese\} only if \{bread, milk\}, \{bread, cheese\}, \{milk, cheese\} are frequent
A-Priori in SQL

```sql
INSERT INTO Baskets1(bid, item)
SELECT * FROM Baskets
WHERE item IN (SELECT item
FROM Baskets
GROUP BY item
HAVING COUNT(*) >= s);
```

- Rewrite join using Basket1 instead of Basket.

Extending Association Rules

- Causality vs. association
  - much trickier
  - hidden variables outside the domain
- More detailed associations:
  - Find items that are bought together frequently, in a particular region, in a particular month.
  - Additional information is already available at the data warehouse.

Example Data Warehouse

![Data Warehouse Diagram]

Need for Data Warehousing

- Integrated, company-wide view of high-quality information.
- Separation of operational and analytical systems and data.

Operational vs. Analytical Data

<table>
<thead>
<tr>
<th>Data Differences</th>
<th>Technical Differences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typical Time-Horizon: Days/Months</td>
<td>Typical Time-Horizon: Years</td>
</tr>
<tr>
<td>Detailed</td>
<td>Summarized (and/or Detailed)</td>
</tr>
<tr>
<td>Current</td>
<td>Values over time (Snapshots)</td>
</tr>
<tr>
<td>Can be Updated</td>
<td>Read (and Append) Only</td>
</tr>
<tr>
<td>Control of Update: Major Issue</td>
<td>Control of Update: No Issue</td>
</tr>
<tr>
<td>Small Amounts used in a Process</td>
<td>Large Amounts used in a Process</td>
</tr>
<tr>
<td>Non-Redundant</td>
<td>Redundancy not an Issue</td>
</tr>
<tr>
<td>High frequency of Access</td>
<td>Low/Most frequency of Access</td>
</tr>
<tr>
<td>Purpose Differences</td>
<td></td>
</tr>
<tr>
<td>For &quot;Clerical Community&quot;</td>
<td>For &quot;Managerial Community&quot;</td>
</tr>
<tr>
<td>Supports Day-to-Day Operations</td>
<td>Supports Manageral Needs</td>
</tr>
<tr>
<td>Application Oriented</td>
<td>Subject Oriented</td>
</tr>
</tbody>
</table>

Application vs. Subject Oriented

<table>
<thead>
<tr>
<th>Application: Health Club Members-Visit Database</th>
<th>Subject: Health Club Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEALTHCLUBMEMBERS</td>
<td></td>
</tr>
<tr>
<td>MembId   Name    MembLevel</td>
<td>DatePayed</td>
</tr>
<tr>
<td>111      Joe       A            01/01/2000</td>
<td></td>
</tr>
<tr>
<td>222      Sue       B            01/01/2000</td>
<td></td>
</tr>
<tr>
<td>333      Pat       A            01/01/2000</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

DAILYVISITORSFROMNONMEMBERS

<table>
<thead>
<tr>
<th>Trid</th>
<th>VisitType</th>
<th>VisitDate</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>11xx22</td>
<td>YP</td>
<td>01/01/2000</td>
<td></td>
</tr>
<tr>
<td>11xx23</td>
<td>NP</td>
<td>02/01/2000</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MEMBERSHIPLEVELS

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Gold $100</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Basic $60</td>
<td></td>
</tr>
</tbody>
</table>

VISITLEVELS

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>NF</td>
<td>With Pool Usage $15</td>
<td></td>
</tr>
<tr>
<td>NP</td>
<td>Without Pool Usage $10</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Application: Health Club Members-Fact Database</th>
<th>Subject: Health Club Revenue</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEALTHCLUBMEMBERS</td>
<td></td>
</tr>
<tr>
<td>Rid     Date         Name    MembLevel</td>
<td>Amount</td>
</tr>
<tr>
<td>7235   01/01/2000  NonMember         $15</td>
<td></td>
</tr>
<tr>
<td>7236   01/01/2000  Member            $100</td>
<td></td>
</tr>
<tr>
<td>7237   01/01/2000  Member            $50</td>
<td></td>
</tr>
<tr>
<td>7238   02/01/2000  Member            $100</td>
<td></td>
</tr>
<tr>
<td>7239   02/01/2000  Member            $10</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>
### Application vs. Subject Oriented

**Application:**
- Health Club Members-Visit Database

**Subject:**
- Health Club Revenue

---

**HEALTHCLUBMEMBERS**

<table>
<thead>
<tr>
<th>MembId</th>
<th>Name</th>
<th>MembLevel</th>
<th>DatePaid</th>
</tr>
</thead>
<tbody>
<tr>
<td>111</td>
<td>Joe</td>
<td>A</td>
<td>01/01/2000</td>
</tr>
<tr>
<td>222</td>
<td>Sue</td>
<td>B</td>
<td>01/01/2000</td>
</tr>
<tr>
<td>333</td>
<td>Pat</td>
<td>A</td>
<td>01/01/2000</td>
</tr>
</tbody>
</table>

**DAILYVISITSFROMNONMEMBERS**

<table>
<thead>
<tr>
<th>Trid</th>
<th>VisitType</th>
<th>VisitDate</th>
</tr>
</thead>
<tbody>
<tr>
<td>11xx22</td>
<td>YP</td>
<td>01/01/2000</td>
</tr>
<tr>
<td>11xx23</td>
<td>NP</td>
<td>02/01/2000</td>
</tr>
<tr>
<td>11xx24</td>
<td>YP</td>
<td>02/01/2000</td>
</tr>
</tbody>
</table>

**MEMBERSHIPLEVELS**

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Gold</td>
<td>$100</td>
</tr>
<tr>
<td>B</td>
<td>Basic</td>
<td>$50</td>
</tr>
</tbody>
</table>

**VISITLEVELS**

<table>
<thead>
<tr>
<th>ID</th>
<th>Type</th>
<th>Fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>YP</td>
<td>With Pool Usage</td>
<td>$15</td>
</tr>
<tr>
<td>NP</td>
<td>Without Pool Usage</td>
<td>$10</td>
</tr>
</tbody>
</table>

**REVENUE**

<table>
<thead>
<tr>
<th>Rid</th>
<th>DateGenerated</th>
<th>CustomerType</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>7235</td>
<td>01/01/2000</td>
<td>NonMember</td>
<td>$15</td>
</tr>
<tr>
<td>7236</td>
<td>01/01/2000</td>
<td>Member</td>
<td>$100</td>
</tr>
<tr>
<td>7237</td>
<td>01/01/2000</td>
<td>Member</td>
<td>$50</td>
</tr>
<tr>
<td>7238</td>
<td>01/01/2000</td>
<td>Member</td>
<td>$100</td>
</tr>
<tr>
<td>7239</td>
<td>02/01/2000</td>
<td>NonMember</td>
<td>$10</td>
</tr>
<tr>
<td>7240</td>
<td>02/01/2000</td>
<td>NonMember</td>
<td>$15</td>
</tr>
</tbody>
</table>

---

### Standard ARM Question:

**What products are frequently bought together?**

- **Location**
  - LocationKey (PK)
  - StoreID
  - Region
  - City

- **Customer**
  - CustomerKey (PK)
  - CustomerID
  - Gender
  - Zip

- **Product**
  - ProductKey (PK)
  - SKU
  - Brand

- **Calendar**
  - CalendarKey (PK)
  - FullDate
  - Month

- **Sales fact table**
  - CalendarKey (FK)
  - ProductKey (FK)
  - CustomerKey (FK)
  - LocationKey (FK)
  - TransactionID

### Analyst may want to know:

**What products are frequently bought together in a particular region and in a particular month?**

- **Location**
  - LocationKey (PK)
  - StoreID
  - Region
  - City

- **Customer**
  - CustomerKey (PK)
  - CustomerID
  - Gender
  - Zip

- **Product**
  - ProductKey (PK)
  - SKU
  - Brand

- **Calendar**
  - CalendarKey (PK)
  - FullDate
  - Month

- **Sales fact table**
  - CalendarKey (FK)
  - ProductKey (FK)
  - CustomerKey (FK)
  - LocationKey (FK)
  - TransactionID

---

### New Challenges

- Interactive mining
- Collaborative/distributed mining
- Peer to peer systems
- Beyond relational data:
  - Text
  - XML
  - Audio
  - Video