Advanced Frequent Pattern Mining
Sequences and Partial Orders

Sequential patterns:
- CHK → MMK → MORT → RESP
- CHK → RRSP → MORT → RESP
- CHK → RRSP → MORT → BROK
- CHK → RRSP → MORT → RESP
- CHK → RRSP → MORT → BROK

<table>
<thead>
<tr>
<th>Cid</th>
<th>Sequence of account opening</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHK → MMK → RRSP → MORT → RESP → BROK</td>
</tr>
<tr>
<td>2</td>
<td>CHK → RRSP → MMK → MORT → RESP → BROK</td>
</tr>
<tr>
<td>3</td>
<td>MMK → CHK → BROK → RESP → RRSP</td>
</tr>
<tr>
<td>4</td>
<td>CHK → MMK → RRSP → MORT → BROK → RESP</td>
</tr>
</tbody>
</table>

Account code | Account type
-------------|--------------|
CHK           | Checking account|
MMK           | Money market  |
RRSP          | Retirement Savings Plan|
MORT          | Mortgage      |
RESP          | Registered Education Savings Plan|
BROK          | Brokerage     |
Why Frequent Orders?

• Frequent orders capture more thorough information than sequential patterns
• Many important applications
  – Bioinformatics: order-preserving clustering of microarray data
  – Web mining and market basket analysis: modeling customer purchase behaviors
  – Network management and intrusion detection: frequent routing paths, signatures for intrusions
  – Preference-based services: partial orders from ranking data
Why Mining Orders Difficult?

- Use sequential patterns to assemble frequent partial orders?
  - One frequent closed partial order may summarize a few sequential patterns
  - Assembling can be costly

Sequential patterns:
- CHK $\rightarrow$ MMK $\rightarrow$ MORT $\rightarrow$ RESP
- CHK $\rightarrow$ MMK $\rightarrow$ MORT $\rightarrow$ BROK
- CHK $\rightarrow$ RRSP $\rightarrow$ MORT $\rightarrow$ RESP
- CHK $\rightarrow$ RRSP $\rightarrow$ MORT $\rightarrow$ BROK
Model

• A sequence \( s \) induces a full order \( R_1 \), if \( R_1 \rightarrow R_2 \), where \( R_2 \) is a partial order, then \( R_1 \) is said to support \( R_2 \)

• The support of a partial order \( R \) in a sequence database is the number of sequences supporting \( R \) in the database

• An order \( R \) is closed if there exists no any \( R' \rightarrow R \) and \( \text{sup}(R)=\text{sup}(R') \)

• Given a minimum support threshold, order \( R \) is a frequent closed partial order if it is closed and passes the support threshold
Ideas

• Depth-first search to generate frequent closed partial orders in transitive reduction
  – Transitive reduction is a succinct representation of partial orders
• Pruning infrequent items, edges and partial orders
• Pruning forbidden edges
• Extracting transitive reductions of frequent partial orders directly
Interesting Orders

(b) A pattern in data set BreastCancer (support=224)

(a) A pattern in data set Yeast (support=80)

(e) Another pattern in data set Snake (support=80).
Freq pat Mining: Achievements

• An important task in data mining
• Frequent pattern mining methodology
  – Candidate generation & test vs. frequent-pattern growth
  – Vertical vs. horizontal format
  – Various optimization methods: database partition, scan reduction, hash tree, sampling, border computation, clustering, etc.
Search Strategies

• Breadth-first vs. depth-first search
  – Apriori-based vs. pattern growth methods
  – Pattern-growth methods have good performance in large and dense databases

• How to search?
  – Reduce recursion/counting cost
  – Compress database
Mining Various Patterns

• Mining closed frequent itemsets and max-patterns
• Mining multi-level, multi-dimensional frequent patterns with flexible support constraints
• Constraint pushing for mining optimization
• From frequent patterns to sequential patterns, correlation and causality
Applications

- Association-based classification
- Iceberg cube computation
- Mining sequential patterns
- Mining partial periodicity, cyclic associations, etc.
- Mining frequent structures, trends, etc.
What Is the Next Step?

• New applications of frequent pattern mining
  – Extensions
  – Use frequent pattern mining ideas to solve other problems – great!
• New types of frequent patterns
Resources for FPM Research

• Data sets:
  – UCI data sets and some real data sets
  – Archived at FIMI workshop website

• Implementations
  – Some source codes are available on the web, but unnecessarily are good implementations