Sequential Pattern Mining

Riccardo Guidotti

Revisited slides from Lecture Notes for Chapter 5 “Introduction to Data Mining”, 2nd Edition by Tan, Steinbach, Karpatne, Kumar
Examples of Sequence

• Sequence of different transactions by a customer at an online store:
  < {Digital Camera,iPad} {memory card} {headphone,iPad cover} >

• Sequence of events causing the nuclear accident at 3-mile Island:
  (http://stellar-one.com/nuclear/staff_reports/summary_SOE_the_initiating_event.htm)
  < {clogged resin} {outlet valve closure} {loss of feedwater}
    {condenser polisher outlet valve shut} {booster pumps trip}
    {main waterpump trips} {main turbine trips} {reactor pressure increases} >

• Sequence of books checked out at a library:
  <{Fellowship of the Ring} {The Two Towers} {Return of the King}>
From Itemsets to Sequences

- Frequent itemsets and association rules focus on transactions and the items that appear there
- Databases of transactions usually have a temporal information
  - Sequential patterns exploit it
- Example data:
  - Market basket transactions
  - Web server logs
  - Tweets
  - Workflow production logs
Frequent Patterns

- Events or combinations of events that appear frequently in the data
- E.g. items bought by customers of a supermarket
Frequent Patterns

- **Frequent itemsets** w.r.t. minimum threshold

- E.g. with Min_freq = 5
Frequent Patterns in Complex Domains

• Frequent sequences (a.k.a. Sequential patterns)
• Input: sequences of events (or of groups)
Frequent Patterns in Complex Domains

- Objective: identify sequences that occur frequently
- Sequential pattern: \{ 🍎🍎\} → 🍎
Sequential Pattern Discovery: Examples

• In telecommunications alarm logs,
  – Inverter_Problem:
    (Excessive_Line_Current) (Rectifier_Alarm) --> (Fire_Alarm)

• In point-of-sale transaction sequences,
  – Computer Bookstore:
    (Intro_To_Visual_C) (C++_Primer) --> (Perl_for_dummies,Tcl_Tk)
  – Athletic Apparel Store:
    (Shoes) (Racket, Racketball) --> (Sports_Jacket)
## Sequence Data and Terminology

<table>
<thead>
<tr>
<th>Sequence Database</th>
<th>Sequence</th>
<th>Element (Transaction)</th>
<th>Event (Item)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td>Purchase history of a given customer</td>
<td>A set of items bought by a customer at time t</td>
<td>Books, diary products, CDs, etc</td>
</tr>
<tr>
<td>Web Data</td>
<td>Browsing activity of a particular Web visitor</td>
<td>A collection of files viewed by a Web visitor after a single mouse click</td>
<td>Home page, index page, contact info, etc</td>
</tr>
<tr>
<td>Event data</td>
<td>History of events generated by a given sensor</td>
<td>Events triggered by a sensor at time t</td>
<td>Types of alarms generated by sensors</td>
</tr>
<tr>
<td>Genome sequences</td>
<td>DNA sequence of a particular species</td>
<td>An element of the DNA sequence</td>
<td>Bases A,T,G,C</td>
</tr>
</tbody>
</table>

**Diagram:**

- **Sequence (E1 E2 E3)**
  - **Element (Transaction) (E1 E2)**
  - **Event (Item) (E3)**
Sequence Data

Sequence Database:

<table>
<thead>
<tr>
<th>Object</th>
<th>Timestamp</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10</td>
<td>2, 3, 5</td>
</tr>
<tr>
<td>A</td>
<td>20</td>
<td>6, 1</td>
</tr>
<tr>
<td>A</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>11</td>
<td>4, 5, 6</td>
</tr>
<tr>
<td>B</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>21</td>
<td>7, 8, 1, 2</td>
</tr>
<tr>
<td>B</td>
<td>28</td>
<td>1, 6</td>
</tr>
<tr>
<td>C</td>
<td>14</td>
<td>1, 8, 7</td>
</tr>
</tbody>
</table>
Formal Definition of a Sequence

- A sequence is an ordered list of transactions
  \[ s = < e_1 e_2 e_3 ... > \]
  - Each transaction is attributed to a specific time or location
  - Each transaction contains a collection of items
    \[ e_i = \{ i_1, i_2, ..., i_k \} \]

- We “measure” sequences with two different notions:
  - Length of a sequence: \(|s|\) is given by the number of transactions of the sequence
  - Size of a sequence: a k-sequence is a sequence that contains k items
Formal Definition of a Sequence

- Example

\[ s = \langle \{A, B\}, \{B, E, F\}, \{A\}, \{E, F, H\} \rangle \]

- Cardinality of \( s \): \(|s| = 4\) transactions
- \( s \) is a 9-sequence as it contains 9 items
- Times associated to elements:
  - \( \{A, B\} \rightarrow \) time=0
  - \( \{B, E, F\} \rightarrow \) time = 120
  - \( \{A\} \rightarrow \) time = 130
  - \( \{E, F, H\} \rightarrow \) time = 200
Sequences without Explicit Time Info

• Default: time of element = position in the sequence

• Example

\[ S = < \{A,C\}, \{E\}, \{A,F\}, \{E,G,H\} > \]

• Default times associated to transactions:
  • \{A,C\} \rightarrow \text{time} = 0
  • \{E\} \rightarrow \text{time} = 1
  • \{A,F\} \rightarrow \text{time} = 2
  • \{E,G,H\} \rightarrow \text{time} = 3
Examples of Sequence

• Web sequence:
  
  `< {Homepage} {Electronics} {Digital Cameras} {Canon Digital Camera} {Shopping Cart} {Order Confirmation} {Return to Shopping} >`

• Sequence of events causing the nuclear accident at 3-mile Island:
  (http://stellar-one.com/nuclear/staff_reports/summary_SOE_the_initiating_event.htm)
  `< {clogged resin & outlet valve closure} {loss of feedwater}
  {condenser polisher outlet valve shut} {booster pumps trip}
  {main waterpump trips & main turbine trips & reactor pressure increases}>`

• Sequence of books checked out at a library:
  `<{Fellowship of the Ring} {The Two Towers} {Return of the King}>`
Formal Definition of a Subsequence

- A sequence $<a_1 a_2 \ldots a_n>$ is contained in another sequence $<b_1 b_2 \ldots b_m>$ ($m \geq n$) if there exist integers $i_1 < i_2 < \ldots < i_n$ such that $a_1 \subseteq b_{i_1}$, $a_2 \subseteq b_{i_1}$, $\ldots$, $a_n \subseteq b_{i_n}$.

<table>
<thead>
<tr>
<th>Data sequence</th>
<th>Subsequence</th>
<th>Contain?</th>
</tr>
</thead>
<tbody>
<tr>
<td>$&lt; {2,4} {3,5,6} {8}&gt;$</td>
<td>$&lt;{2} {3,5}&gt;$</td>
<td>Yes</td>
</tr>
<tr>
<td>$&lt;{1,2} {3,4}&gt;$</td>
<td>$&lt;{1} {2}&gt;$</td>
<td>No</td>
</tr>
<tr>
<td>$&lt;{2,4} {2,4} {2,5}&gt;$</td>
<td>$&lt;{2} {4}&gt;$</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Formal Definition of Sequential Pattern

- The **support** of a subsequence $w$ is the fraction of data sequences that contain $w$

  subsequence $w$: \{A\}, \{B,C\}, \{D\}

  Input sequences:

  - {D}
  - {A,C}
  - {A,B,C}
  - {F}
  - {D}
  - {A}
  - {A,B,D}
  - {D}
  - {A,C}
  - {B,C}
  - {D}
  - {B,E}
  - {B,C}
  - {D}
  - {F}

  support of $w$: $2/4 = 0.50$ (50%)

- A **sequential pattern**
  - is a **frequent** subsequence
  - i.e., a subsequence whose support is $\geq$ **minsup**
Formal Definition of Sequential Pattern

- Remark: a subsequence (i.e. a candidate pattern) might be mapped into a sequence in several different ways
  - Each mapping is an **instance** of the subsequence
  - In mining sequential patterns we need to find only one instance

\[
\begin{aligned}
I_1 &= 1, I_2 = 2, I_3 = 5 \\
\{D\} &\quad \{A, C\} &\quad \{A, B, C\} &\quad \{F\} &\quad \{B, E\} &\quad \{D\} \\
I_1 &= 1, I_2 = 4, I_3 = 5 \\
\{D\} &\quad \{A, C\} &\quad \{A, B, C\} &\quad \{F\} &\quad \{B, E\} &\quad \{D\} \\
I_1 &= 2, I_2 = 4, I_3 = 5 \\
\{D\} &\quad \{A, C\} &\quad \{A, B, C\} &\quad \{F\} &\quad \{B, E\} &\quad \{D\}
\end{aligned}
\]
Exercise 1

- find instances/occurrence of the following patterns

  \(<\{C\} \{H\} \{C\}\>

  \(<\{A\} \{F\} >\)

  \(<\{A\} \{A\} \{D\}\>

  \(<\{A\} \{A,B\} \{F\}\>

- in the input sequence below

  \(<\{A,C\} \{C,D\} \{F,H\} \{A,B\} \{B,C,D\} \{E\} \{A,B,D\} \{F\} >\)
Exercise 2

• find instances/occurrence of the following patterns

  < {C} {H} {C} >

  < {A} {B} >

  < {C} {C} {E} >

  < {A} {E} >

• in the input sequence below

  < {A,C} {C,D,E} {F} {A,H} {B,C,D} {E} {A,B,D} >

  t=0   t=1   t=2   t=3   t=4   t=5   t=6
Sequential Pattern Mining: Definition

• Given:
  • a database of sequences
  • a user-specified minimum support threshold, $minsup$

• Task:
  • Find all subsequences with support $\geq minsup$
Sequential Pattern Mining: Challenge

- Trivial approach: generate all possible k-subsequences, for k=1,2,3,... and compute support
- Combinatorial explosion!
  - With frequent itemsets mining we had:
    - N. of k-subsets = \( \binom{n}{k} \)
    - \( n = n. \) of distinct items in the data
  - With sequential patterns:
    - N. of k-subsequences = \( n^k \)
    - The same item can be repeated:
      - \(< \{A\} \{A\} \{B\} \{A\} \ldots >\)
Sequential Pattern Mining: Challenge

- Even if we generate them from input sequences
  - E.g.: Given a n-sequence: `<{a b} {c d e} {f} {g h i}>`
    - Examples of subsequences:
      - `<{a} {c d} {f} {g}>`, `<{c d e}>`, `<{b} {g}>`, etc.
    - Number of k-subsequences can be extracted from it

\[
\begin{align*}
\langle \{a\ b\} \{c\ d\ e\} \{f\} \{g\ h\ i\}\rangle & \quad n = 9 \\
\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\
\langle \{a\} \quad \{d\ e\} \quad \{i\}\rangle \\
\end{align*}
\]

\[k=4: \quad \text{Y \_ \_ \_ \_ Y Y \_ \_ \_ Y}\]

Answer:
\[
\binom{n}{k} = \binom{9}{4} = 126
\]
Sequential Pattern Mining: Example

<table>
<thead>
<tr>
<th>Object</th>
<th>Timestamp</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1,2,4</td>
</tr>
<tr>
<td>A</td>
<td>2</td>
<td>2,3</td>
</tr>
<tr>
<td>A</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1,2</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>2,3,4</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1,2</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>2,3,4</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>2,4,5</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>D</td>
<td>2</td>
<td>3,4</td>
</tr>
<tr>
<td>D</td>
<td>3</td>
<td>4,5</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>1,3</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2,4,5</td>
</tr>
</tbody>
</table>

*Minsup* = 50%

**Examples of Frequent Subsequences:**

- `<{1,2}> s=60%`
- `<{2,3}> s=60%`
- `<{2,4}> s=80%`
- `<{3} {5}> s=80%`
- `<{1} {2}> s=80%`
- `<{2} {2}> s=60%`
- `<{1} {2,3}> s=60%`
- `<{2} {2,3}> s=60%`
- `<{1,2} {2,3}> s=60%`
Generalized Sequential Pattern
Generalized Sequential Pattern (GSP)

- **Follows the same structure of Apriori**
  - Start from short patterns and find longer ones at each iteration

- **Based on “Apriori principle” or “anti-monotonicity of support”**
  - If one sequence $S_1$ is contained in sequence $S_2$, then the support of $S_2$ cannot be larger than that of $S_1$:
    $$S_1 \subseteq S_2 \Rightarrow \sup(S_1) \geq \sup(S_2)$$

- **Intuitive proof**
  - Any input sequence that contains $S_2$ will also contain $S_1$

---

Input sequence:

- $\{D\}$
- $\{A,C\}$
- $\{A,B,C\}$
- $\{F\}$
- $\{B,E\}$
- $\{D\}$

Patterns:

- $\{B\}$
- $\{D\}$
- $\{B,C\}$
- $\{D\}$

- $S_1$ starts with $\{B\}$ and continues with $\{D\}$.
- $S_2$ starts with $\{A\}$ and continues with $\{B,C\}$ and $\{D\}$.
Generalized Sequential Pattern (GSP)

- **Follows the same structure of Apriori**
  - Start from short patterns and find longer ones at each iteration

- **Step 1:**
  - Make the first pass over the sequence database D to yield all the 1-transaction frequent sequences

- **Step 2:** Repeat until no new frequent sequences are found:
  - **Candidate Generation:**
    - Merge pairs of frequent subsequences found in the \((k-1)th\) pass to generate candidate sequences that contain \(k\) items
  - **Candidate Pruning:**
    - Prune candidate \(k\)-sequences that contain infrequent \((k-1)\)-subsequences
  - **Support Counting:**
    - Make a new pass over the sequence database D to find the support for these candidate sequences
  - **Candidate Elimination:**
    - Eliminate candidate \(k\)-sequences whose actual support is less than \(\text{minsup}\)
Extracting Sequential Patterns

• Given \( n \) items: \( i_1, i_2, i_3, \ldots, i_n \)
  • Candidate 1-subsequences:
    \(<\{i_1\}>, <\{i_2\}>, <\{i_3\}>, \ldots, <\{i_n\}>\)
  • Candidate 2-subsequences:
    \(<\{i_1, i_2\}>, <\{i_1, i_3\}>, \ldots, <\{i_1\} \{i_1\}>, <\{i_1\} \{i_2\}>, \ldots, <\{i_{n-1}\} \{i_n\}>\)
  • Candidate 3-subsequences:
    \(<\{i_1, i_2, i_3\}>, <\{i_1, i_2, i_4\}>, \ldots, <\{i_1, i_2\} \{i_1\}>, <\{i_1, i_2\} \{i_2\}>, \ldots, <\{i_1\} \{i_1\} \{i_3\}>, <\{i_1\} \{i_1\} \{i_2\}>, \ldots\)

• Remark: items within a transaction are ordered
  • YES: \(<\{i_1, i_2, i_3\}>\)  • NO: \(<\{i_3, i_1, i_2\}>\)
Candidate Generation

- Base case (k=2):
  - Merging two frequent 1-sequences \(<\{i_1\}\>\) and \(<\{i_2\}\>\) will produce two candidate 2-sequences: \(<\{i_1\} \{i_2\} \rangle\) and \(<\{i_1 \ i_2\}\>\)
  - Special case: \(i_1\) can be merged with itself: \(<\{i_1\} \{i_1\}>\)

- General case (k>2):
  - A frequent \((k-1)\)-sequence \(w_1\) is merged with another frequent \((k-1)\)-sequence \(w_2\) to produce a candidate \(k\)-sequence if the subsequence obtained by removing the first event in \(w_1\) is the same as the one obtained by removing the last event in \(w_2\).
    - The resulting candidate after merging is given by the sequence \(w_1\) extended with the last event of \(w_2\).
      - Simplest case: \(<\{d\}{a}\{b\}> + <\{a\}{b}\{c\}> = <\{d\}{a}\{b\}{c}\>\)
      - If last two events in \(w_2\) belong to the same element => last event in \(w_2\) becomes part of the last element in \(w_1\): \(<\{d\}{a}\{b\}> + <\{a\}{b, c}\> = <\{d\}{a}\{b, c\}>\)
      - Otherwise, the last event in \(w_2\) becomes a separate element appended to the end of \(w_1\): \(<\{a, d\}{b}\> + <\{d\}{b}\{c\}> = <\{a, d\}{b}{c}\>\)
  - Special case: check if \(w_1\) can be merged with itself
    - Works when it contains only one event type: \(<\ {a}\ \{a\}> + <\{a\} \{a\}> = <\{a\} \{a\} \{a\}>\)
Candidate Generation Examples

• Merging the sequences \( w_1 = \langle 1 \rangle \{2 3\} \{4\} \) and \( w_2 = \langle 2 3 \rangle \{4 5\} \) will produce the candidate sequence \( \langle 1 \rangle \{2 3\} \{4 5\} \) because the last two items in \( w_2 \) (4 and 5) belong to the same transaction.

• Merging the sequences \( w_1 = \langle 1 \rangle \{2 3\} \{4\} \) and \( w_2 = \langle 2 3 \rangle \{4 \} \{5\} \) will produce the candidate sequence \( \langle 1 \rangle \{2 3\} \{4 \} \{5\} \) because the last two items in \( w_2 \) (4 and 5) do not belong to the same transaction.

• Can we merge \( w_1 = \langle 1 \rangle \{2 6\} \{4\} \) and \( w_2 = \langle 1 \rangle \{2\} \{4 5\} \) ?
  We do not have to merge the sequences \( w_1 = \langle 1 \rangle \{2 6\} \{4\} \) and \( w_2 = \langle 1 \rangle \{2\} \{4 5\} \) to produce the candidate \( \langle 1 \rangle \{2 6\} \{4 5\} \).
  Notice that if the latter is a viable candidate, it will be obtained by merging \( w_1 \) with \( \langle 2 6\} \{4 5\} \).
Candidate Pruning

• Based on Apriori principle:
  • If a k-sequence W contains a (k-1)-subsequence that is not frequent, then W is not frequent and can be pruned

• Method:
  • Enumerate all (k-1)-subsequence:
    • \{a,b\}{c}{d} \rightarrow \{b\}{c}{d}, \{a\}{c}{d}, \{a,b\}{d}, \{a,b\}{c}
  • Each subsequence generated by cancelling 1 item in W
    • Number of (k-1)-subsequences = k
  • Remark: candidates are generated by merging two “mother” (k-1)-subsequences that we know to be frequent
    • Correspond to remove the first event or the last one
    • Number of significant (k-1)-subsequences to test = k – 2
    • Special cases: at step k=2 the pruning has no utility, since the only (k-1)-subsequences are the “mother” ones
GSP Example

**Frequent 3-sequences**

- `< {1} {2} {3} >`
- `< {1} {2} {5} >`
- `< {1} {5} {3} >`
- `< {2} {3} {4} >`
- `< {2} {5} {3} >`
- `< {3} {4} {5} >`
- `< {5} {3} {4} >`

**Candidate Generation**

- `< {1} {2} {3} {4} >`
- `< {1} {2} {5} {3} >`
- `< {1} {5} {3} {4} >`
- `< {2} {3} {4} {5} >`
- `< {2} {5} {3} {4} >`

**Candidate Pruning**

- `< {1} {2} {5} {3} >`
GSP Exercise

• Given the following dataset of sequences

<table>
<thead>
<tr>
<th>ID</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ab → a → b</td>
</tr>
<tr>
<td>2</td>
<td>b → a → cd</td>
</tr>
<tr>
<td>3</td>
<td>a → b</td>
</tr>
<tr>
<td>4</td>
<td>a → a → bd</td>
</tr>
</tbody>
</table>

• Generate sequential patterns if min_sup = 35%
## GSP Exercise - Solution

<table>
<thead>
<tr>
<th>ID</th>
<th>Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a b → a → b</td>
</tr>
<tr>
<td>2</td>
<td>b → a → c d</td>
</tr>
<tr>
<td>3</td>
<td>a → b</td>
</tr>
<tr>
<td>4</td>
<td>a → a → b d</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sequential pattern</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>100 %</td>
</tr>
<tr>
<td>b</td>
<td>100 %</td>
</tr>
<tr>
<td>d</td>
<td>50 %</td>
</tr>
<tr>
<td>a → a</td>
<td>50 %</td>
</tr>
<tr>
<td>a → b</td>
<td>75 %</td>
</tr>
<tr>
<td>a → d</td>
<td>50 %</td>
</tr>
<tr>
<td>b → a</td>
<td>50 %</td>
</tr>
<tr>
<td>a → a → b</td>
<td>50 %</td>
</tr>
</tbody>
</table>
Timing Constraints

• Motivation by examples:
  • Sequential Pattern \{milk} \rightarrow \{cookies\}
    • It might suggest that cookies are bought to better enjoy milk
    • Yet, we might obtain it even if all customers by milk and after 6 months buy cookies, in which case our interpretation is wrong
  • \{cheese A\} \rightarrow \{cheese B\}
    • Does it mean that buying and eating cheese A induces the customer to try also cheese B (e.g. by the same brand)?
    • Maybe, yet if they are bought within 20 minutes it is like that they were to be bought together (and the customer forgot it)
  • \{buy PC\} \rightarrow \{buy printer\} \rightarrow \{ask for repair\}
    • Is it a good or bad sign?
    • It depends on how much time the whole process took:
      • Short time => issues, Long time => OK, normal life cycle
Timing Constraints

- Define 3 types of constraint on the instances to consider
  - E.g. ask that the pattern instances last no more than 30 days

\[
\begin{align*}
\{A & \ B\} \quad \{C\} \quad \{D & \ E\} \\
\leq x_g & \quad > n_g & \leq m_s \\
\end{align*}
\]

- \(x_g\): max-gap \(\Rightarrow\) Each transaction of the pattern instance must be at most \(x_g\) time after the previous one
- \(n_g\): min-gap \(\Rightarrow\) Each transaction of the pattern instance must be at least \(n_g\) time after the previous one
- \(m_s\): maximum span \(\Rightarrow\) The overall duration of the pattern instance must be at most \(m_s\)

\(x_g = 2, \ n_g = 0, \ m_s = 4\) \(\Rightarrow\) consecutive transactions at most distance 2 & overall duration at most 4 time units

<table>
<thead>
<tr>
<th>Data sequence</th>
<th>Subsequence</th>
<th>Contain?</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt; {2,4} {3,5,6} {4,7} {4,5} {8}&gt;)</td>
<td>(&lt; {6} {5}&gt;)</td>
<td>Yes</td>
</tr>
<tr>
<td>(&lt; {1} {2} {3} {4} {5}&gt;)</td>
<td>(&lt; {1} {4}&gt;)</td>
<td>No</td>
</tr>
<tr>
<td>(&lt; {1} {2,3} {3,4} {4,5}&gt;)</td>
<td>(&lt; {2} {3} {5}&gt;)</td>
<td>Yes</td>
</tr>
<tr>
<td>(&lt; {1,2} {3} {2,3} {3,4} {2,4} {4,5}&gt;)</td>
<td>(&lt; {1,2} {5}&gt;)</td>
<td>No</td>
</tr>
</tbody>
</table>

Each transaction of the pattern instance must be at most \(x_g\) time after the previous one.
Each transaction of the pattern instance must be at least \(n_g\) time after the previous one.
The overall duration of the pattern instance must be at most \(m_s\).
Mining Sequential Patterns with Timing Constraints

• Approach 1:
  • Mine sequential patterns without timing constraints
  • Postprocess the discovered patterns
  • Dangerous: might generate billions of sequential patterns to obtain only a few time-constrained ones

• Approach 2:
  • Modify GSP to directly prune candidates that violate timing constraints
  • Question: Does Apriori principle still hold?
Apriori Principle with Time Constraints

- **Case 1: max-span**
- **Intuitive check**
  - Does any input sequence that contains $S_2$ will also contain $S_1$?

![Diagram showing input sequences and spans]

- **Span for $S_2$**: $\text{Span} = 4$
- **Span for $S_1$**: $\text{Span} = 1$

- **When $S_1$ has less transactions, $S_1$ span can (only) decrease**
  - If $S_2$ span is OK, then also $S_1$ span is OK
Apriori Principle with Time Constraints

- **Case 2: min-gap**
- **Intuitive check**
  - Does any input sequence that contains S2 will also contain S1?

```
S_1
\{A\} \{D\}

S_2
\{A\} \{B,C\} \{D\}

Input sequence
\{D\} \{A,C\} \{A,B,C\} \{F\} \{B,E\} \{D\}
```

- **Gaps for S_2**:
  - Gap = 1
  - Gap = 3

- **Gaps for S_1**:
  - Gap = 4

- **When S1 has less transactions, gaps for S1 can (only) increase**
  - If S2 gaps are OK, they are OK also for S1
Apriori Principle with Time Constraints

- **Case 3: max-gap**
- **Intuitive check**
  - Does any input sequence that contains $S_2$ will also contain $S_1$?

\[
\begin{align*}
S_1 & \quad \{A\} \quad \{D\} \\
S_2 & \quad \{A\} \quad \{B,C\} \quad \{D\} \\
\text{Input sequence} & \quad \{D\} \quad \{A,C\} \quad \{A,B,C\} \quad \{F\} \quad \{B,E\} \quad \{D\}
\end{align*}
\]

- **Gaps for $S_2$:**
  - Gap = 1
  - Gap = 3

- **Gaps for $S_1$:**
  - Gap = 4

- **When $S_1$ has less transactions, gaps for $S_1$ can (only) increase**
  - **Happens when $S_1$ has lost an internal element w.r.t. $S_2$**
  - Even if $S_2$ gaps are OK, $S_1$ gaps might grow too large w.r.t. max-gap
### Apriori Principle for Sequence Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Timestamp</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1,2,4</td>
</tr>
<tr>
<td>A</td>
<td>2</td>
<td>2,3</td>
</tr>
<tr>
<td>A</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1,2</td>
</tr>
<tr>
<td>B</td>
<td>2</td>
<td>2,3,4</td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>1,2</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>2,3,4</td>
</tr>
<tr>
<td>C</td>
<td>3</td>
<td>2,4,5</td>
</tr>
<tr>
<td>D</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>D</td>
<td>2</td>
<td>3,4</td>
</tr>
<tr>
<td>D</td>
<td>3</td>
<td>4,5</td>
</tr>
<tr>
<td>E</td>
<td>1</td>
<td>1,3</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2,4,5</td>
</tr>
</tbody>
</table>

Suppose:
- $x_g = 1$ (max-gap)
- $n_g = 0$ (min-gap)
- $m_s = 5$ (maximum span)
- $\text{minsup} = 60\%$

- ${\{2}\ {5}\} \text{ support} = 40\%$
- but
- ${\{2}\ {3}\ {5}\} \text{ support} = 60\%$

Problem exists because of max-gap constraint
No such problem if max-gap is infinite
Contiguous Subsequences

• s is a contiguous subsequence of
  \[ w = \langle e_1 \rangle \langle e_2 \rangle \ldots \langle e_k \rangle \]
  if any of the following conditions hold:
  1. s is obtained from w by deleting an item from either \( e_1 \) or \( e_k \)
  2. s is obtained from w by deleting an item from any element \( e_i \) that contains more than 2 items
  3. s is a contiguous subsequence of \( s' \) and \( s' \) is a contiguous subsequence of w (recursive definition)

• Examples: \( s = \langle \{1\} \{2\} \rangle \)
  • is a contiguous subsequence of
    \( \langle \{1\} \{2\} \{3\} \rangle \), \( \langle \{1\} \{2\} \{3\} \rangle \), and \( \langle \{3\} \{4\} \{1\} \{2\} \{3\} \{4\} \rangle \)
  • is not a contiguous subsequence of
    \( \langle \{1\} \{3\} \{2\} \rangle \) and \( \langle \{2\} \{1\} \{3\} \{2\} \rangle \)

Key point: avoids internal "jumps"
Not interesting for our usage
Modified Candidate Pruning Step

- **Without maxgap constraint:**
  - A candidate $k$-sequence is pruned if at least one of its $(k-1)$-subsequences is infrequent

- **With maxgap constraint:**
  - A candidate $k$-sequence is pruned if at least one of its contiguous $(k-1)$-subsequences is infrequent

- **Remark:** the “pruning power” is now reduced
  - Less subsequences to test for “killing” the candidate
Other kinds of patterns for sequences

• In some domains, we may have only one very long time series
  • Example:
    • monitoring network traffic events for attacks
    • monitoring telecommunication alarm signals

• Goal is to find frequent sequences of events in the time series
  • Now we have to count “instances”, but which ones?
  • This problem is also known as frequent episode mining

Pattern: \(<E1> <E3>\)
References

- Sequential Pattern Mining. Chapter 7. Introduction to Data Mining.
Exercises SPM
## Sequential Pattern – Exercise 1

### a) (3 points) Given the following input sequence

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>{A}</td>
<td>{B,F}</td>
<td>{E}</td>
<td>{A,B}</td>
<td>{A,C,D}</td>
<td>{F}</td>
<td>{B,E}</td>
</tr>
<tr>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td>t=4</td>
<td>t=5</td>
<td>t=6</td>
<td>t=7</td>
</tr>
</tbody>
</table>

Show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering min-gap = 1 (i.e., gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

<table>
<thead>
<tr>
<th>Example</th>
<th>Occurrences</th>
<th>Occurrences with min-gap = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ex.: &lt;{B}{E}&gt;</strong></td>
<td>(&lt;1,2&gt; &lt;1,6&gt; &lt;3,6&gt;)</td>
<td>(&lt;1,6&gt; &lt;3,6&gt;)</td>
</tr>
<tr>
<td>(w_1 = &lt;{A} {B} {E}&gt;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_2 = &lt;{B}{D}&gt;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_3 = &lt;{F}{E}{C,D}&gt;)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Sequential Pattern – Exercise 1

**a) (3 points)** Given the following input sequence

<table>
<thead>
<tr>
<th></th>
<th>t=0</th>
<th>t=1</th>
<th>t=2</th>
<th>t=3</th>
<th>t=4</th>
<th>t=5</th>
<th>t=6</th>
<th>t=7</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{B,F}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{E}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{A,B}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{A,C,D}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{F}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{B,E}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>{C,D}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering min-gap = 1 (i.e. gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

### Answer:

<table>
<thead>
<tr>
<th>Subsequence</th>
<th>Occurrences</th>
<th>Occurrences with min-gap = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;{B}{E}&gt;)</td>
<td>(&lt;1,2&gt; &lt;1,6&gt; &lt;3,6&gt;)</td>
<td>(&lt;1,6&gt;&lt;3,6&gt;)</td>
</tr>
<tr>
<td>(w_1 = \langle{A} {B} {E}\rangle)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_2 = \langle{B}{D}\rangle)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_3 = \langle{F}{E}{C,D}\rangle)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
a) **(3 points)** Given the following input sequence

\[
\begin{array}{cccccccc}
\text{t=0} & \{A\} & \text{t=1} & \{B,F\} & \text{t=2} & \{E\} & \text{t=3} & \{A,B\} & \text{t=4} & \{A,C,D\} & \text{t=5} & \{F\} & \text{t=6} & \{B,E\} & \text{t=7} & \{C,D\}
\end{array}
\]

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column); the second time considering min-gap = 1 (i.e. gap > 1, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

**Answer:**

<table>
<thead>
<tr>
<th>Subsequence</th>
<th>Occurrences</th>
<th>Occurrences with min-gap = 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>ex.: (&lt;{B}{E}&gt;)</td>
<td>(&lt;1,2,6,3,6&gt;)</td>
<td>(&lt;1,6,3,6&gt;)</td>
</tr>
<tr>
<td>(w_1 = {A} {B} {E}&gt;)</td>
<td>(&lt;0,1,2,0,1,6,0,3,6&gt;)</td>
<td>(&lt;0,3,6&gt;)</td>
</tr>
<tr>
<td>(w_2 = {B} {D}&gt;)</td>
<td>(&lt;1,4,1,7,3,4,3,7,6,7&gt;)</td>
<td>(&lt;1,4,1,7,3,7&gt;)</td>
</tr>
<tr>
<td>(w_3 = {F} {E} {C,D}&gt;)</td>
<td>(&lt;1,2,4,1,2,7,1,6,7,5,6,7&gt;)</td>
<td><strong>none</strong></td>
</tr>
</tbody>
</table>
Sequential Pattern – Exercise 2

a) (3 points) Given the following input sequence

\[
\begin{array}{cccccccc}
\text{t=0} & \{\text{B,F}\} & \text{t=1} & \{\text{A}\} & \text{t=2} & \{\text{A,B}\} & \text{t=3} & \{\text{C,D,F}\} & \text{t=4} & \{\text{E}\} & \text{t=5} & \{\text{B,E}\} & \text{t=6} & \{\text{C,D}\}
\end{array}
\]

Show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column); the second time considering max-gap = 4 (i.e. gap <= 4, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

<table>
<thead>
<tr>
<th>Subsequence</th>
<th>Occurrences</th>
<th>Occurrences with max-gap = 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w_1 = &lt;{B} {E}&gt;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_2 = &lt;{B}{D}&gt;)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(w_3 = &lt;{F}{B}{C,D}&gt;)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Sequential Pattern – Exercise 2 – Solution

a) **(3 points)** Given the following input sequence

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>B,F</td>
<td>A</td>
<td>A,B</td>
<td>C,D,F</td>
<td>E</td>
<td>B,E</td>
<td>C,D</td>
</tr>
<tr>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td>t=4</td>
<td>t=5</td>
<td>t=6</td>
</tr>
</tbody>
</table>

show all the occurrences (there can be more than one or none, in general) of each of the following subsequences in the input sequence above. Repeat the exercise twice: the first time considering no temporal constraints (left column): the second time considering max-gap = 4 (i.e. gap <= 4, right column). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).  

**Answer:**

<table>
<thead>
<tr>
<th>Subsequence</th>
<th>Occurrences</th>
<th>Occurrences with max-gap =4</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;{B} {E}&gt;)</td>
<td>(&lt;0,4&gt;  &lt;0,5&gt;  &lt;2,4&gt;  &lt;2,5&gt;)</td>
<td>(&lt;0,4&gt;  &lt;2,4&gt;  &lt;2,5&gt;)</td>
</tr>
<tr>
<td>(&lt;{B}{D}&gt;)</td>
<td>(&lt;0,3&gt;  &lt;0,6&gt;  &lt;2,3&gt;  &lt;2,6&gt;  &lt;5,6&gt;)</td>
<td>(&lt;0,3&gt;  &lt;2,3&gt;  &lt;2,6&gt;  &lt;5,6&gt;)</td>
</tr>
<tr>
<td>(&lt;{F} {B}{C,D}&gt;)</td>
<td>(&lt;0,2,3&gt;  &lt;0,2,6&gt;  &lt;0,5,6&gt;  &lt;3,5,6&gt;)</td>
<td>(&lt;0,2,3&gt;  &lt;0,2,6&gt;  &lt;3,5,6&gt;)</td>
</tr>
</tbody>
</table>
Given the input sequences listed in the table below (column 1), show for each of them all the occurrences of subsequences \( \{A\} \rightarrow \{D\} \) and \( \{A\} \rightarrow \{C,D\} \), and finally write its total support. Repeat the exercise twice: the first time considering no temporal constraints (columns 2 and 4); the second time considering min-gap = 1 (i.e. gap > 1) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

<table>
<thead>
<tr>
<th>column 1</th>
<th>column 2</th>
<th>column 3</th>
<th>column 4</th>
<th>column 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>(&lt;{A,B,F} {C} {C,D,F} {E} {C,D}&gt;)</td>
<td>{A} → {D}</td>
<td>{B} → {C,D}</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>(&lt;t=0\ t=1\ t=2\ t=3\ t=4&gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,B} {C} {A,B} {C,D}&gt;)</td>
<td>{A} → {D}</td>
<td>{B} → {C,D}</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>(&lt;t=0\ t=1\ t=2\ t=3&gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{F} {A,B,F} {A,B,C,D} {D} {E} {C}&gt;)</td>
<td>{A} → {D}</td>
<td>{B} → {C,D}</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>(&lt;t=0\ t=1\ t=2\ t=3\ t=4&gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,F} {B,C} {A,B} {E} {D}&gt;)</td>
<td>{A} → {D}</td>
<td>{B} → {C,D}</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>(&lt;t=0\ t=1\ t=2\ t=3&gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,B,F} {A,C} {A,B,D} {C} {C,D}&gt;)</td>
<td>{A} → {D}</td>
<td>{B} → {C,D}</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>(&lt;t=0\ t=1\ t=2\ t=3\ t=4&gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total support:
Sequential Pattern – Exercise 3 – Solution

Given the input sequences listed in the table below (column 1), show for each of them all the occurrences of subsequences \{A\} → \{D\} and \{A\} → \{C,D\}, and finally write its total support. Repeat the exercise twice: the first time considering no temporal constraints (columns 2 and 4); the second time considering min-gap = 1 (i.e. gap > 1) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

<table>
<thead>
<tr>
<th>column 1</th>
<th>column 2</th>
<th>column 3</th>
<th>column 4</th>
<th>column 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A} → {D}</td>
<td>{A} → {C,D}</td>
<td>{B} → {C,D}</td>
<td>{B} → {C,D}</td>
<td>{B} → {C,D}</td>
</tr>
<tr>
<td>(&lt;{A,B,F} {C} {C,D,F} {E} {C,D} &gt;)</td>
<td>No constraints</td>
<td>min-gap = 1</td>
<td>No constraints</td>
<td>min-gap = 1</td>
</tr>
<tr>
<td>t=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,B} {C} {A,B} {C,D} &gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{F} {A,B,F} {A,B,C,D} {D} {E} {C} &gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,F} {B,C} {A,B} {E} {D} &gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(&lt;{A,B,F} {A,C} {A,B,D} {C} {C,D} &gt;)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t=4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total support:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 (100%)</td>
<td>5 (100%)</td>
<td>4 (80%)</td>
<td>3 (60%)</td>
<td></td>
</tr>
</tbody>
</table>
Given the input sequences listed in the table below (column 1), show for each of them **all the occurrences** of subsequences \( \{A\} \rightarrow \{A\} \rightarrow \{D\} \) and \( \{B\} \rightarrow \{C,D\} \), and finally write its total support. Repeat the exercise twice: the first time **considering no temporal constraints** (columns 2 and 4); the second time **considering max-gap = 2** (i.e. gap <= 2) (columns 3 and 5). Each occurrence should be represented by its corresponding list of time stamps, e.g.: \(<0,2,3> = <t=0, t=2, t=3>\).

<table>
<thead>
<tr>
<th>column 1</th>
<th>column 2</th>
<th>column 3</th>
<th>column 4</th>
<th>column 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A} → {A} → {D}</td>
<td>{B} → {C,D}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>No constraints</td>
<td>max-gap = 2</td>
<td>No constraints</td>
<td>max-gap = 2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>&lt; {A,B,F} {C} {A,C,D,F} {E} {C,D} &gt;</th>
<th>t=0</th>
<th>t=1</th>
<th>t=2</th>
<th>t=3</th>
<th>t=4</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; {A,B} {C} {A,B} {C,D} &gt;</td>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td></td>
</tr>
<tr>
<td>&lt; {F} {A,F} {A,C} {D} {A,E} {C} &gt;</td>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td>t=4</td>
</tr>
<tr>
<td>&lt; {A,F} {B,C,D} {A,B} {B,E} {D} &gt;</td>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td>t=4</td>
</tr>
<tr>
<td>&lt; {A,B} {A,D} {A} {C} {A} {C,D} &gt;</td>
<td>t=0</td>
<td>t=1</td>
<td>t=2</td>
<td>t=3</td>
<td>t=4</td>
</tr>
</tbody>
</table>

Total support:
b) **(3 points)** Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

\[
\begin{align*}
\{A\} & \rightarrow \{B \ C\} \rightarrow \{C\} \rightarrow \{D\} \\
\{A \ C\} & \rightarrow \{B\} \rightarrow \{C\} \rightarrow \{C\} \\
\{D\} & \rightarrow \{C\} \rightarrow \{B\} \rightarrow \{C \ D\} \\
\{A \ B\} & \rightarrow \{D\} \rightarrow \{C\} \rightarrow \{C \ D\} \rightarrow \{E\}
\end{align*}
\]
b) (3 points) Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

{ A } -> { B C } -> { C } -> { D }
{ A C } -> { B } -> { C } -> { C }
{ D } -> { C } -> { B } -> { C D }
{ A B } -> { D } -> { C } -> { C D } -> { E }
b) (3 points) Simulate the execution of the GSP algorithm on the following dataset of sequences, assuming a minimum support threshold of 60%.

\[
\begin{align*}
\{\text{A}\} & \rightarrow \{\text{B}\} \rightarrow \{\text{C}\} \rightarrow \{\text{D}\} \\
\{\text{AC}\} & \rightarrow \{\text{B}\} \rightarrow \{\text{C}\} \rightarrow \{\text{C}\} \\
\{\text{D}\} & \rightarrow \{\text{C}\} \rightarrow \{\text{B}\} \rightarrow \{\text{CD}\} \\
\{\text{AB}\} & \rightarrow \{\text{D}\} \rightarrow \{\text{C}\} \rightarrow \{\text{CD}\} \rightarrow \{\text{E}\}
\end{align*}
\]

2-seq (2-transactions)

<table>
<thead>
<tr>
<th>1-seq</th>
<th>2-seq (1-transaction)</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A}</td>
<td>{A} \rightarrow {B}</td>
</tr>
<tr>
<td>{B}</td>
<td>{B} \rightarrow {C}</td>
</tr>
<tr>
<td>{C}</td>
<td>{C} \rightarrow {D}</td>
</tr>
<tr>
<td>{BC}</td>
<td>{C} \rightarrow {B}</td>
</tr>
<tr>
<td>{AC}</td>
<td>{C} \rightarrow {C}</td>
</tr>
<tr>
<td>{CD}</td>
<td>{C} \rightarrow {D}</td>
</tr>
<tr>
<td>{AB}</td>
<td>{D} \rightarrow {B}</td>
</tr>
</tbody>
</table>

\[\{\text{B}\} \rightarrow \{\text{C}\} \rightarrow \{\text{CD}\} \rightarrow \{\text{E}\}\]

(pruning)
b) **(3 points)** Running the GSP algorithm on a dataset of sequences, at the end of the second iteration it found the frequent 3-sequences on the left, and at the next iteration it generated (among the others) the candidate 4-sequences on the right. Which of the candidates will be **pruned**, and why?

### Frequent 3-sequences

| A B } → { C } | A } → { D } → { C } |
| { A B } → { D } | { B } → { C } → { C } |
| { A } → { C D } | { B } → { C } → { D } |
| { B } → { C D } | { B } → { D } → { C } |
| { A } → { C } → { C } | { D } → { C } → { C } |
| { A } → { C } → { D } | { D } → { C } → { D } |

### Candidates

1. { A B } → { C D } 
2. { A } → { D } → { C } → { D } 
3. { B } → { D } → { C } → { D } 
4. { A B } → { D } → { C } 
5. { A B } → { C } → { D }
b) (3 points) Running the GSP algorithm on a dataset of sequences, at the end of the second iteration it found the frequent 3-sequences on the left, and at the next iteration it generated (among the others) the candidate 4-sequences on the right. Which of the candidates will be pruned, and why?

<table>
<thead>
<tr>
<th>Frequent 3-sequences</th>
<th>Candidates</th>
</tr>
</thead>
<tbody>
<tr>
<td>{A B } → {C}</td>
<td>1. {A B} → {C D}</td>
</tr>
<tr>
<td>{A B } → {D}</td>
<td>2. {A} → {D} → {C} → {D}</td>
</tr>
<tr>
<td>{A} → {CD}</td>
<td>3. {B} → {D} → {C} → {D}</td>
</tr>
<tr>
<td>{B} → {CD}</td>
<td>4. {A B} → {D} → {C}</td>
</tr>
<tr>
<td>{A} → {C} → {C}</td>
<td>5. {A B} → {C} → {D}</td>
</tr>
<tr>
<td>{A} → {C} → {D}</td>
<td></td>
</tr>
</tbody>
</table>

**Answer:**

Candidates

1. {A B} → {C D}
2. {A} → {D} → {C} → {D} ← PRUNED
3. {B} → {D} → {C} → {D} ← PRUNED
4. {A B} → {D} → {C}
5. {A B} → {C} → {D}

Missing from frequent 3-sequences

- A → D → D
- B → D → D