Elements of Machine Learning & Data Science

Association Rules and Sequence Mining

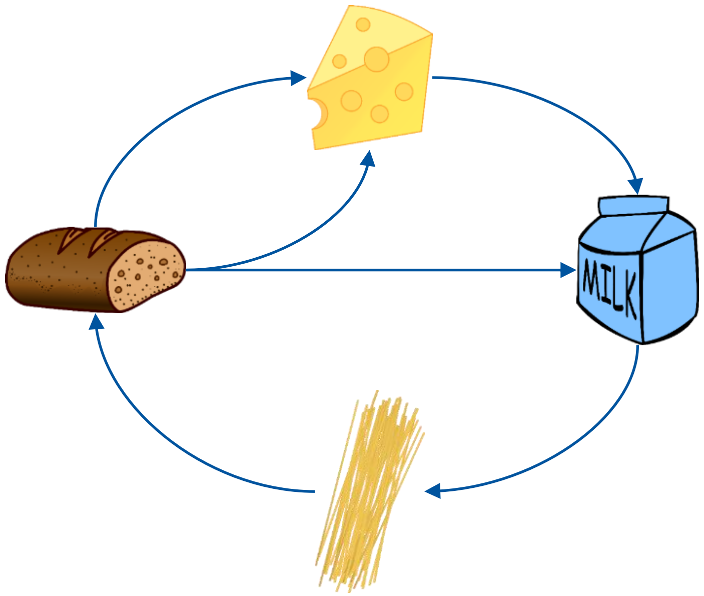
Lecture 10

Prof. Wil van der Aalst

Marco Pegoraro, M.Sc.
Christopher Schwanen, M.Sc.
Tsunghao Huang, M.Sc.

Association Rules

1. **Introduction**
2. Generating Association Rules
3. Applications
4. Evaluation
5. Simpson’s Paradox



From Frequent Itemsets to Association Rules

- Frequent Itemsets – a combinatorial explosion
- How to determine the interesting ones?
- How to turn itemsets into rules?

[1]

A Larger Supermarket May Have Up To 50000 Distinct Items

$$2^{50000} - 1 \approx 3.16 \times 10^{15051}$$

and the number of rules exceeds the number of possible itemsets . . .

Association Rules - Notation

- $\mathcal{I} = \{I_1, I_2, \ldots, I_D\}$ is the set of all possible items
- A transaction $\mathcal{T} \in \mathbb{P}(\mathcal{I}) \setminus \{\emptyset\}$ is a non-empty itemset
- A dataset $\mathcal{X} \in \mathbb{M}(\mathbb{P}(\mathcal{I}))$ $(\text{such that } \emptyset \notin \mathcal{X})$ is a multiset of transactions
(Here, $\mathbb{M}$ is the multiset and $\mathbb{P}$ is the powerset operator)

Same as before

- $\mathcal{A} \Rightarrow \mathcal{B}$ with $\mathcal{A} \subseteq \mathcal{I}, \mathcal{B} \subseteq \mathcal{I}$ and $\mathcal{A} \cap \mathcal{B} = \emptyset$ is an association rule
- For example, $\{\text{Cheese}, \text{Bread}\} \Rightarrow \{\text{Milk}\}$

$$\mathcal{A} \Rightarrow \mathcal{B}$$

Association Rules - Preview

- {Cheese, Bread} $\Rightarrow$ {Milk}
  People that buy Cheese and Bread also tend to buy Milk.
- {Track1, Track2} $\Rightarrow$ {Track3}
  Students that take the Track 1 and Track 2 modules of BridgingAI also tend to take the Track 3 courses. (We hope you do!)
- {Bitburger} $\Rightarrow$ {Heineken, Palm}
  People that buy Bitburger beer tend to buy both Heineken and Palm beer.
- {Carbonara, Margherita } $\Rightarrow$ {Espresso, Tiramisu}
  People that buy Carbonara and Margherita also tend to buy Espresso and Tiramisu.
- {part-245, part-345, part-456} $\Rightarrow$ {part-372}
  When Parts 245, 345, and 456 are replaced, then often also Part 372 is replaced.

Support and Confidence

- Support: fraction of instances containing all items in $\mathcal{A} \cup \mathcal{B}$

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A} \cup \mathcal{B}) = \frac{\text{support\_count}(\mathcal{A} \cup \mathcal{B})}{\text{support\_count}(\emptyset)} = \frac{|[\mathcal{T} \in \mathcal{X} \mid \mathcal{A} \cup \mathcal{B} \subseteq \mathcal{T}]|}{|\mathcal{X}|}$$

Support and Confidence

- Support: fraction of instances containing all items in $\mathcal{A} \cup \mathcal{B}$

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A} \cup \mathcal{B}) = \frac{\text{support\_count}(\mathcal{A} \cup \mathcal{B})}{\text{support\_count}(\emptyset)} = \frac{|[\mathcal{T} \in \mathcal{X} | \mathcal{A} \cup \mathcal{B} \subseteq \mathcal{T}]|}{|\mathcal{X}|}$$

- Confidence: fraction of instances containing items in $\mathcal{A}$ which contain items in $\mathcal{A} \cup \mathcal{B}$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\text{support}(\mathcal{A} \cup \mathcal{B})}{\text{support}(\mathcal{A})} = \frac{\text{support\_count}(\mathcal{A} \cup \mathcal{B})}{\text{support\_count}(\mathcal{A})} = \frac{|[\mathcal{T} \in \mathcal{X} | \mathcal{A} \cup \mathcal{B} \subseteq \mathcal{T}]|}{|[\mathcal{T} \in \mathcal{X} | \mathcal{A} \subseteq \mathcal{T}]|}$$

Support and Confidence - Example

ID	Bought Items
1	{Bread, Cheese, Milk, Pasta}
2	{Bread, Cheese, Chips}
3	{Cheese, Pasta, Milk}
4	{Bread, Cheese, Milk}
5	{Bread, Pasta}

All three items Bread, Cheese and Milk need to be in the transaction to count

$$\text{support}(\{\text{Bread}\} \Rightarrow \{\text{Cheese}, \text{Milk}\}) = \text{support}(\{\text{Bread}, \text{Cheese}, \text{Milk}\}) = \frac{2}{5}$$

Support and Confidence - Example

ID	Bought Items
1	{Bread, Cheese, Milk, Pasta}
2	{Bread, Cheese, Chips}
3	{Cheese, Pasta, Milk}
4	{Bread, Cheese, Milk}
5	{Bread, Pasta}

$$\text{support}(\{\text{Bread}\} \Rightarrow \{\text{Cheese}, \text{Milk}\}) = \text{support}(\{\text{Bread}, \text{Cheese}, \text{Milk}\}) = \frac{2}{5}$$

$$\text{support}(\{\text{Bread}\} \Rightarrow \{\text{Cheese}, \text{Milk}\}) = \text{support}(\{\text{Cheese}, \text{Milk}\} \Rightarrow \{\text{Bread}\})$$

$$\text{support}(\{\text{Bread}\} \Rightarrow \{\text{Cheese}, \text{Milk}\}) = \text{support}(\{\text{Bread}, \text{Cheese}\} \Rightarrow \{\text{Milk}\})$$

Symmetric: moving the item does not change the value

Support and Confidence - Example

ID	Bought Items
1	{Bread, Cheese, Milk, Pasta}
2	{Bread, Cheese, Chips}
3	{Cheese, Pasta, Milk}
4	{Bread, Cheese, Milk}
5	{Bread, Pasta}

$$\text{conf}(\{\text{Bread}\} \Rightarrow \{\text{Cheese}, \text{Milk}\}) = \frac{\text{support}(\{\text{Bread,Cheese,Milk}\})}{\text{support}(\{\text{Bread}\})} = \frac{2}{4}$$

Support and Confidence - Example

ID	Bought Items
1	{Bread, Cheese, Milk, Pasta}
2	{Bread, Cheese, Chips}
3	{Cheese, Pasta, Milk}
4	{Bread, Cheese, Milk}
5	{Bread, Pasta}

$$\mathrm{conf}(\{\mathrm{Bread}\} \Rightarrow \{\mathrm{Cheese}, \mathrm{Milk}\}) = \frac{\mathrm{support}(\{\mathrm{Bread}, \mathrm{Cheese}, \mathrm{Milk}\})}{\mathrm{support}(\{\mathrm{Bread}\})} = \frac{2}{4}$$

$$\mathrm{conf}(\{\mathrm{Cheese}, \mathrm{Milk}\} \Rightarrow \{\mathrm{Bread}\}) = \frac{\mathrm{support}(\{\mathrm{Bread}, \mathrm{Cheese}, \mathrm{Milk}\})}{\mathrm{support}(\{\mathrm{Cheese}, \mathrm{Milk}\})} = \frac{2}{3}$$

$$\mathrm{conf}(\{\mathrm{Bread}\} \Rightarrow \{\mathrm{Cheese}, \mathrm{Milk}\}) \neq \mathrm{conf}(\{\mathrm{Cheese}, \mathrm{Milk}\} \Rightarrow \{\mathrm{Bread}\})$$

Not symmetric
(equality holds only
in some rare cases)

Support and Confidence - Example

ID	Bought Items
1	{Bread, Cheese, Milk, Pasta}
2	{Bread, Cheese, Chips}
3	{Cheese, Pasta, Milk}
4	{Bread, Cheese, Milk}
5	{Bread, Pasta}

$$\mathrm{conf}(\{\mathrm{Bread}\} \Rightarrow \{\mathrm{Cheese}, \mathrm{Milk}\}) = \frac{\mathrm{support}(\{\mathrm{Bread},\mathrm{Cheese},\mathrm{Milk}\})}{\mathrm{support}(\{\mathrm{Bread}\})} = \frac{2}{4}$$

$$\mathrm{conf}(\{\mathrm{Bread}, \mathrm{Cheese}\} \Rightarrow \{\mathrm{Milk}\}) = \frac{\mathrm{support}(\{\mathrm{Bread},\mathrm{Cheese},\mathrm{Milk}\})}{\mathrm{support}(\{\mathrm{Bread},\mathrm{Cheese}\})} = \frac{2}{3}$$

General rule:

$$\mathrm{conf}(\{A, B\} \Rightarrow \{C\}) \geq \mathrm{conf}(\{A\} \Rightarrow \{B, C\})$$

Probabilistic Interpretation

- Support: probability that an instance contains $\mathcal{A} \cup \mathcal{B}$

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A} \cup \mathcal{B}) \approx P(\mathcal{A} \cup \mathcal{B})$$

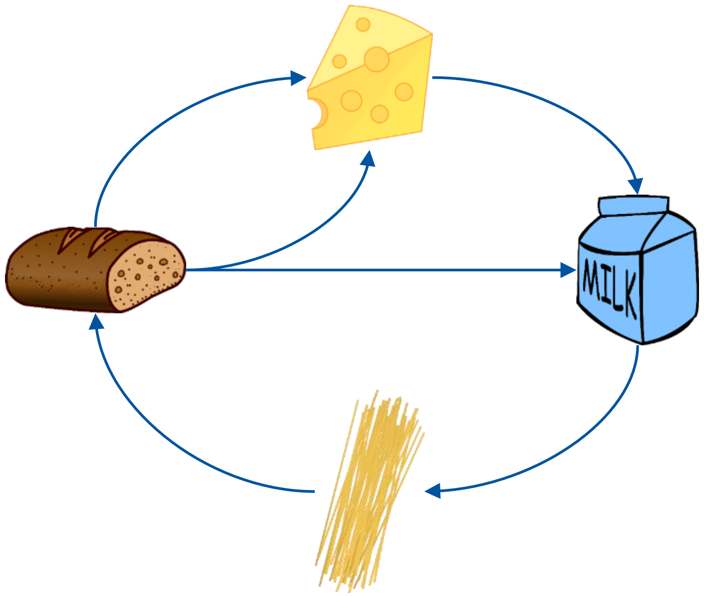
- Confidence: conditional probability that an instance contains items in $\mathcal{B}$, given that it contains items in $\mathcal{A}$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\text{support}(\mathcal{A} \cup \mathcal{B})}{\text{support}(\mathcal{A})} \approx P(\mathcal{B} \mid \mathcal{A})$$

Take ‘probability’ with a grain of salt - we are only considering a sample.

Association Rules

1. Introduction
2. **Generating Association Rules**
3. Applications
4. Evaluation
5. Simpson’s Paradox



From Frequent Itemsets to Association Rules

Given: a dataset $\mathcal{X} \in \mathbb{M}(\mathbb{P}(\mathcal{I})), \text{min\_sup}, \text{min\_conf}$

How to generate all association rules that have high support and high confidence?

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A} \cup \mathcal{B}) \geq \text{min\_sup}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\text{support}(\mathcal{A} \cup \mathcal{B})}{\text{support}(\mathcal{A})} \geq \text{min\_conf}$$

Ensuring $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) \geq \text{min\_sup}$

✓ Easy!

- Use frequent itemsets as a basis
- Consider frequent itemsets $\mathcal{C} = \mathcal{A} \cup \mathcal{B}$ such that $|\mathcal{C}| \geq 2$ and $\mathcal{C} \geq \text{min\_sup}$ (apply Apriori or FP-growth to generate such frequent itemsets)

Ensuring $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) \geq \text{min\_sup}$

✓ Easy!

- Use frequent itemsets as a basis
- Consider frequent itemsets $\mathcal{C} = \mathcal{A} \cup \mathcal{B}$ such that $|\mathcal{C}| \geq 2$ and $\mathcal{C} \geq \text{min\_sup}$
  (apply Apriori or FP-growth to generate such frequent itemsets)
- Generate candidate rules $\mathcal{A} \Rightarrow \mathcal{B}$ by considering all splits of $\mathcal{C}$ into two non-empty disjoint subsets
- However: the number of such candidate rules is $2^{|\mathcal{C}|} - 2$!

Ensuring $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) \geq \text{min\_sup}$

✓ Easy!

- Use frequent itemsets as a basis
- Consider frequent itemsets $\mathcal{C} = \mathcal{A} \cup \mathcal{B}$ such that $|\mathcal{C}| \geq 2$ and $\mathcal{C} \geq \text{min\_sup}$ (apply Apriori or FP-growth to generate such frequent itemsets)
- Generate candidate rules $\mathcal{A} \Rightarrow \mathcal{B}$ by considering all splits of $\mathcal{C}$ into two non-empty disjoint subsets
- However: the number of such candidate rules is $2^{|\mathcal{C}|} - 2$!

$$\mathcal{C} = \{\{\text{Bread}\}, \{\text{Cheese}\}, \{\text{Milk}\}, \{\text{Pasta}\}\}$$

$$\underbrace{\{\text{Bread}\}}_{\mathcal{A}}, \underbrace{\{\text{Cheese}\}, \{\text{Milk}\}, \{\text{Pasta}\}}_{\mathcal{B}}$$

$$\underbrace{\{\text{Bread}\}, \{\text{Cheese}\}}_{\mathcal{A}}, \underbrace{\{\text{Milk}\}, \{\text{Pasta}\}}_{\mathcal{B}}$$

$$\vdots$$

$$\underbrace{\{\text{Bread}\}, \{\text{Cheese}\}, \{\text{Milk}\}}_{\mathcal{B}}, \underbrace{\{\text{Pasta}\}}_{\mathcal{A}}$$

$|\mathcal{C}| = 4 \implies 2^4 - 2 = 14$ candidate rules

… and the number of candidate frequent itemsets was already exponential!

Ensuring $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) \geq \mathrm{min\_conf}$

No additional pass over the data needed

- Itemsets $\mathcal{A} \cup \mathcal{B}$ and $\mathcal{A}$ are frequent
  → their supports have already been computed when using Apriori or FP-growth
- Therefore, we can simply test every candidate rule and only return the ones that satisfy the criterion:

$$\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A})} \geq \mathrm{min\_conf}$$

Ensuring $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) \geq \mathrm{min\_conf}$

- Itemsets $\mathcal{A} \cup \mathcal{B}$ and $\mathcal{A}$ are frequent
  → their supports have already been computed when using Apriori or FP-growth
- Therefore, we can simply test every candidate rule and only return the ones that satisfy the criterion:

$$\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A})} \geq \mathrm{min\_conf}$$

But…
- There could be way too many association rules.
- Most are not interesting!

Confidence-Based Pruning

- Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$, and itemset $\mathcal{C}$ such that $\mathcal{C} \cap (\mathcal{A} \cup \mathcal{B}) = \emptyset$
- It holds that $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B} \cup \mathcal{C}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B} \cup \mathcal{C})}{\mathrm{support}(\mathcal{A})} \leq \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A})} = \mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B})$

recall that the support of a superset is lower or equal

Confidence-Based Pruning

- Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$, and itemset $\mathcal{C}$ such that $\mathcal{C} \cap (\mathcal{A} \cup \mathcal{B}) = \emptyset$
- It holds that $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B} \cup \mathcal{C}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B} \cup \mathcal{C})}{\mathrm{support}(\mathcal{A})} \leq \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A})} = \mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B})$

- Hence, if $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) \leq \mathrm{min\_conf}$ then $\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B} \cup \mathcal{C}) \leq \mathrm{min\_conf}$
- Adding $\mathcal{C}$ to the right part makes the rule stronger
- We can focus on the stronger rules meeting the confidence threshold

- This does not apply to $\mathrm{conf}(\mathcal{A} \cup \mathcal{C} \Rightarrow \mathcal{B}) \quad ?? \quad \mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B})$
- Additions to the left part of the rule may lead to an increase or decrease
  - $\{\text{Cheese}\} \Rightarrow \{\text{Wine}\}$ may have a confidence of 0.2
  - $\{\text{Cheese}, \text{Babyfood}\} \Rightarrow \{\text{Wine}\}$ may have a confidence of 0.1
  - $\{\text{Cheese}, \text{Chips}\} \Rightarrow \{\text{Wine}\}$ may have a confidence of 0.3

Removing Redundant Rules

- Consider two different association rules $\mathcal{A} \Rightarrow \mathcal{B}$ and $\mathcal{A}' \Rightarrow \mathcal{B}'$ with identical support and confidence, i.e.:
  - $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A}' \Rightarrow \mathcal{B}')$
  - $\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{conf}(\mathcal{A}' \Rightarrow \mathcal{B}')$

- $\mathcal{A}' \Rightarrow \mathcal{B}'$ is redundant if $\mathcal{A}' \subseteq \mathcal{A}$ and $\mathcal{B}' \subseteq \mathcal{B}$

- Using only closed frequent itemsets will avoid generating redundant rules

  (Recall: An itemset is closed if there is no proper superset that has the same support)

Avoiding Generation of Redundant Rules

1. Assume $\mathcal{A}' \Rightarrow \mathcal{B}'$ is redundant, i.e., there is another rule $\mathcal{A} \Rightarrow \mathcal{B}$ such that
   - $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A}' \Rightarrow \mathcal{B}')$
   - $\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{conf}(\mathcal{A}' \Rightarrow \mathcal{B}')$
   - $\mathcal{A}' \subseteq \mathcal{A}$
   - $\mathcal{B}' \subseteq \mathcal{B}$
   - It holds that $\mathcal{A}' \cup \mathcal{B}' \subset \mathcal{A} \cup \mathcal{B}$ (because the rules are different)

Avoiding Generation of Redundant Rules

1. Assume $\mathcal{A}' \Rightarrow \mathcal{B}'$ is redundant, i.e., there is another rule $\mathcal{A} \Rightarrow \mathcal{B}$ such that
   - $\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{support}(\mathcal{A}' \Rightarrow \mathcal{B}')$
   - $\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \text{conf}(\mathcal{A}' \Rightarrow \mathcal{B}')$
   - $\mathcal{A}' \subseteq \mathcal{A}$
   - $\mathcal{B}' \subseteq \mathcal{B}$
   - It holds that $\mathcal{A}' \cup \mathcal{B}' \subset \mathcal{A} \cup \mathcal{B}$ (because the rules are different)
2. Also, assume $\mathcal{A} \cup \mathcal{B}$ and $\mathcal{A}' \cup \mathcal{B}'$ are closed, i.e., there are no proper supersets with the same support
   - Hence, $\text{support}(\mathcal{A}' \Rightarrow \mathcal{B}') > \text{support}(\mathcal{A} \Rightarrow \mathcal{B})$ (cannot be equal, $\mathcal{A} \cup \mathcal{B}$ is closed)

Therefore, we find a contradiction. Closed itemsets cannot produce redundant rules.

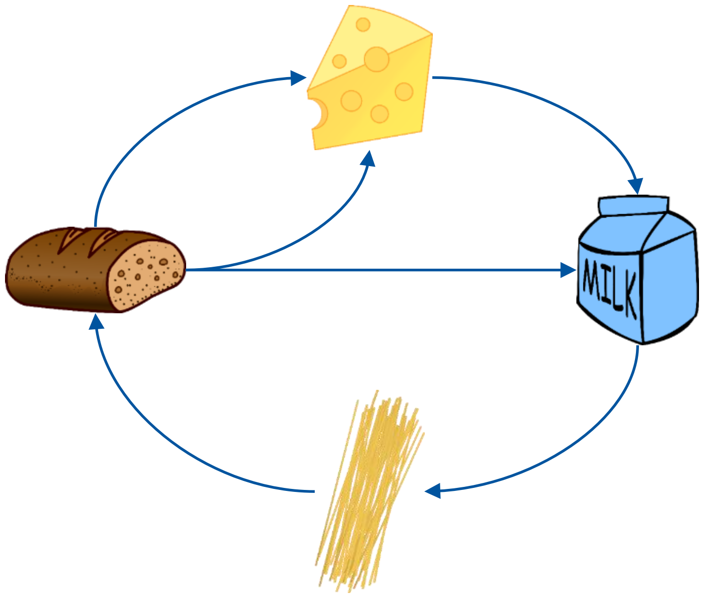
Summary

How to generate association rules that are interesting?

- We can generate candidate rules with high support based on frequent itemsets
- We can filter those candidates with high confidence without going back to the data
- We can prune the rules based on confidence: $\mathrm{min\_conf} \leq \mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B} \cup \mathcal{C}) \leq \cancel{\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B})}$
- We can focus on closed frequent itemsets to avoid redundant rules
- Not enough, we need additional concepts such as “surprisingness” (lift)

Association Rules

1. Introduction
2. Generating Association Rules
3. **Applications**
4. Evaluation
5. Simpson’s Paradox



Spotify

{Flowers(Miley Cyrus), Unholy(Sam Smith)} $\Rightarrow$ {Levitating(Dua Lipa)}

{One(Metallica),Trasher(Evile)} $\Rightarrow$ {Augen-Auf(Oomph),The Trooper(Iron Maiden)}

{Birds(Anouk),Irgendwo(Nena)} $\Rightarrow$ {Leiser(Lea),Klavier(Lea)}

- 456 million active listeners
- 195 million premium subscribers
- Over 80 million songs

(As of January 2023)

Amazon

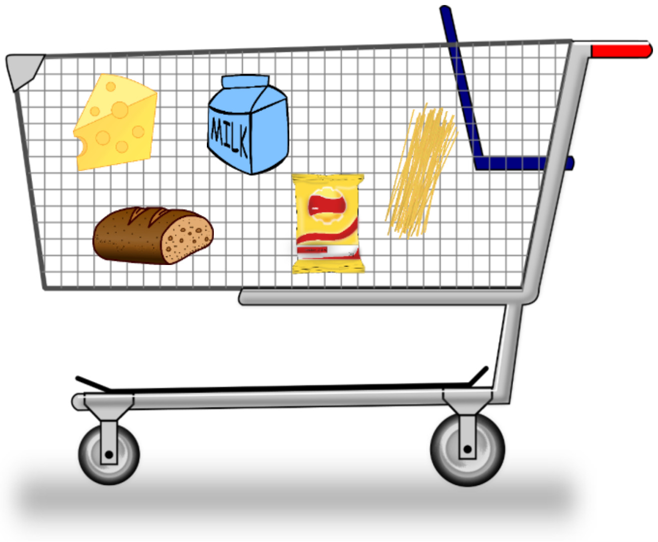
{Echo-Show-8,Fire-TV-Cube} $\Rightarrow$ {Kindle-Paperwhite}

{Fire-TV-Stick-8} $\Rightarrow$ {Fire-HD-8,Blink-Mini}

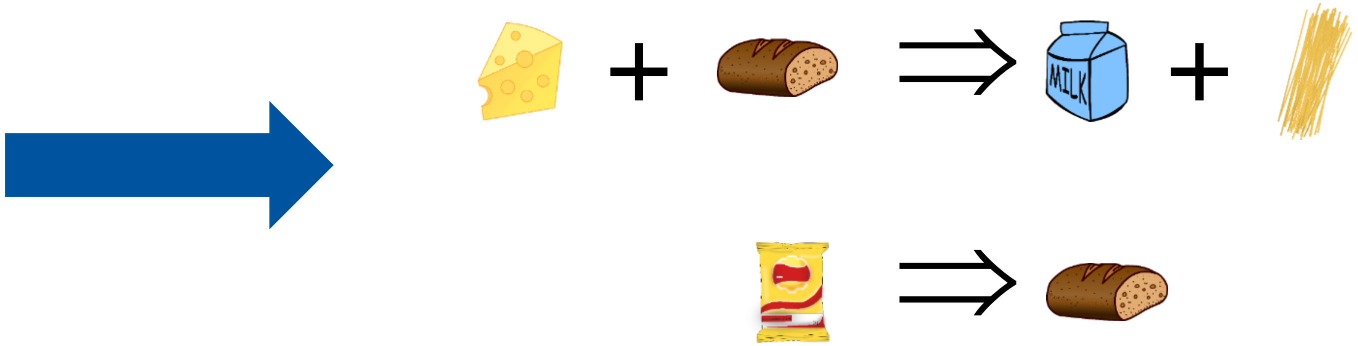
- 300 million active users
- Over 2 million third-party seller businesses
- Around 350 million items on the marketplace

(As of January 2023)

Supermarkets



support = 0.01
confidence = 0.85
lift = 1.67



support = 0.001
confidence = 0.15
lift = 1.2

Next to confidence and support, we will see other measures like lift

Using Features Values As Items And Instances As Itemsets

Rain	Wind	Temp	Play
Yes	Yes	15	No
No	No	34	Yes
Yes	No	23	Yes
Yes	Yes	20	Yes
No	Yes	28	No
…	…	…	…

- Examples consider items as products, services, etc.
- Items can also be normal features values and transactions normal instances
- This leads to itemsets of the form {f1=v1, f2=v2, … fn=vn} for each instance

Using Features Values As Items And Instances As Itemsets

Rain	Wind	Temp	Play
Yes	Yes	15	No
No	No	34	Yes
Yes	No	23	Yes
Yes	Yes	20	Yes
No	Yes	28	No
…	…	…	…

[{Rain=Yes, Wind=Yes, Temp=15, Play=No},
{Rain=No, Wind=No, Temp=34, Play=Yes},
{Rain=Yes, Wind=No, Temp=23, Play=Yes},
{Rain=Yes, Wind=Yes, Temp=20, Play=Yes},
{Rain=No, Wind=Yes, Temp=28, Play=No},
…]

- Examples consider items as products, services, etc.
- Items can also be normal features values and transactions normal instances
- This leads to itemsets of the form {f1=v1, f2=v2, … fn=vn} for each instance

Using Features Values As Items And Instances As Itemsets

Rain	Wind	Temp	Play
Yes	Yes	15	No
No	No	34	Yes
Yes	No	23	Yes
Yes	Yes	20	Yes
No	Yes	28	No
…	…	…	…

[{Rain=Yes, Wind=Yes, 10≤Temp<20, Play=No},
{Rain=No, Wind=No, 30≤Temp<40, Play=Yes},
{Rain=Yes, Wind=No, 20≤Temp<30, Play=Yes},
{Rain=Yes, Wind=Yes, 20≤Temp<30, Play=Yes},
{Rain=No, Wind=Yes, 20≤Temp<30, Play=No},
…]

- Items can also be ranges for continuous feature values
  - Temp≥25
  - Temp<25
  - 20≤Temp<30
  - Etc.
- Any dataset having instances and features can be converted into a multiset of transactions $\mathcal{X} \in \mathbb{M}(\mathbb{P}(\mathcal{I}))$

Using Features Values As Items And Instances As Itemsets

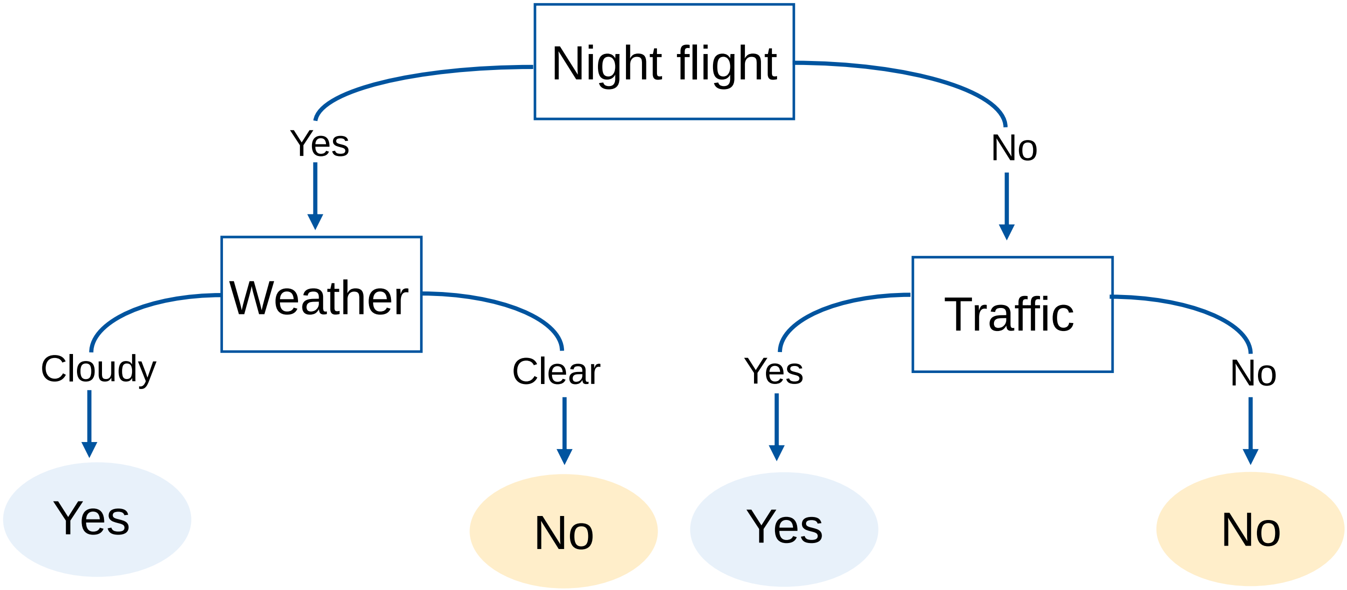
Rain	Wind	Temp	Play
Yes	Yes	15	No
No	No	34	Yes
Yes	No	23	Yes
Yes	Yes	20	Yes
No	Yes	28	No
…	…	…	…

- Any dataset having instances and features can be converted into a multiset of transactions $\mathcal{X} \in \mathbb{M}(\mathbb{P}(\mathcal{I}))$
- Hence, we can also have association rules of the form $\mathcal{A} \Rightarrow \mathcal{B}$ with $\mathcal{A} \subseteq \mathcal{I}, \mathcal{B} \subseteq \mathcal{I}$ and $\mathcal{A} \cap \mathcal{B} = \emptyset$

{Rain=Yes, Wind=Yes} $\Rightarrow$ {Play=No}
{Temp>30} $\Rightarrow$ {Rain=No, Wind=No}
{Temp>20,Play=Yes} $\Rightarrow$ {Wind=No}

Link To Classification and Decision Trees

Weather	Traffic	Night flight	Flight delayed
Cloudy	No	Yes	**Yes**
Cloudy	Yes	No	**Yes**
Cloudy	Yes	No	**Yes**
Clear	Yes	Yes	**No**
Clear	No	No	**No**
Clear	No	No	**No**



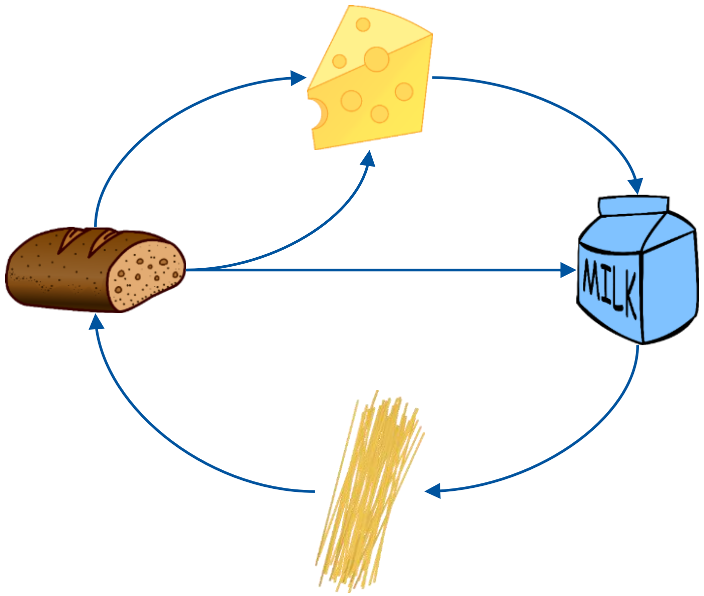
{Night_flight=Yes, Weather=Cloudy} $\Rightarrow$ {Flight_delayed=Yes}
{Night_flight=Yes, Weather=Clear} $\Rightarrow$ {Flight_delayed=No}
{Night_flight=No, Traffic=Yes} $\Rightarrow$ {Flight_delayed=Yes}
{Night_flight=No, Traffic=No} $\Rightarrow$ {Flight_delayed=No}

Summary

- Association rules can be learned for “normal itemsets” and itemsets based on feature values
- Classification rules can be expressed as association rules
- The challenge remains that there are exponentially many candidate rules
- Confidence and support are only part of the story
    - What if many rules meet the two thresholds?
    - How to select the most interesting ones?

Association Rules

1. Introduction
2. Generating Association Rules
3. Applications
4. **Evaluation**
5. Simpson's Paradox



Association rules $\mathcal{A} \Rightarrow \mathcal{B}$

{Cheese, Chips} $\Rightarrow$ {Wine, Beer}
{One(Metallica),Trasher(Evile)} $\Rightarrow$ {Augen-Auf(Oomph), The Trooper(Iron Maiden)}
{Temp>20,Play=Yes} $\Rightarrow$ {Wind=No}
{Night_flight=No, Traffic=Yes} $\Rightarrow$ {Flight_delayed=Yes}
{Gender=Male, Sport=Football} $\Rightarrow$ {Favorite_food=Currywurst, Age>40}
…

How to evaluate the quality of a rule?

Confusion matrix for association rules

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$\#\mathcal{A}\mathcal{B}$	$\#\mathcal{A}\overline{\mathcal{B}}$	$\#\mathcal{A}$
$\mathcal{A}$ is not included	$\#\overline{\mathcal{A}}\mathcal{B}$	$\#\overline{\mathcal{A}\mathcal{B}}$	$\#\overline{\mathcal{A}}$
	$\#\mathcal{B}$	$\#\overline{\mathcal{B}}$	$\#\mathrm{ALL}$

$$\mathrm{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\#\mathcal{A}\mathcal{B}}{\#ALL} \qquad \mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\#\mathcal{A}\mathcal{B}}{\#\mathcal{A}}$$

Confusion matrix for association rules

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$\#\mathcal{A}\mathcal{B}$	$\#\mathcal{A}\overline{\mathcal{B}}$	$\#\mathcal{A}$
$\mathcal{A}$ is not included	$\#\overline{\mathcal{A}}\mathcal{B}$	$\#\overline{\mathcal{A}\mathcal{B}}$	$\#\overline{\mathcal{A}}$
	$\#\mathcal{B}$	$\#\overline{\mathcal{B}}$	$\#\text{ALL}$

The lower the better

The higher the better

Not captured in any of the metrics

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\#\mathcal{A}\mathcal{B}}{\#ALL}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\#\mathcal{A}\mathcal{B}}{\#\mathcal{A}}$$

High Support and High Confidence

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	100	0	100
$\mathcal{A}$ is not included	0	0	0
	100	0	100

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{100}{100} \qquad \text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{100}{100}$$

Low Support and High Confidence

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	10	0	10
$\mathcal{A}$ is not included	40	50	90
	50	50	100

$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{10}{100}$

$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{10}{10}$

Low Support and Low Confidence

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	10	40	50
$\mathcal{A}$ is not included	25	25	50
	35	65	100

$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{10}{100}$

$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{10}{50}$

Support and Confidence Don't Tell The Full Story

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	80	10	90
$\mathcal{A}$ is not included	**0**	**10**	10
	80	20	100

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{80}{100}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{80}{90}$$

Seems to be a good rule because if $\mathcal{A}$ is not included, $\mathcal{B}$ is also never included

Support and Confidence Don't Tell The Full Story

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	80	10	90
$\mathcal{A}$ is not included	**10**	**0**	10
	90	10	100

Not captured in any of the metrics

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{80}{100}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{80}{90}$$

Same support and confidence, but seems to be a poor rule because if $\mathcal{A}$ is not included, $\mathcal{B}$ is always included

The distribution of counts in the second row does not influence support and confidence

We need Lift: How surprising?

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$\#\mathcal{A}\mathcal{B}$	$\#\mathcal{A}\overline{\mathcal{B}}$	$\#\mathcal{A}$
$\mathcal{A}$ is not included	$\#\overline{\mathcal{A}}\mathcal{B}$	$\#\overline{\mathcal{A}\mathcal{B}}$	$\#\overline{\mathcal{A}}$
	$\#\mathcal{B}$	$\#\overline{\mathcal{B}}$	$\#\mathrm{ALL}$

$$\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A}) \cdot \mathrm{support}(\mathcal{B})} = \frac{P(\mathcal{A} \cup \mathcal{B})}{P(\mathcal{A}) \cdot P(\mathcal{B})} = \frac{\frac{\#\mathcal{A}\mathcal{B}}{\#\mathrm{ALL}}}{\frac{\#\mathcal{A}}{\#\mathrm{ALL}} \cdot \frac{\#\mathcal{B}}{\#\mathrm{ALL}}} = \frac{\#\mathcal{A}\mathcal{B} \cdot \#\mathrm{ALL}}{\#\mathcal{A} \cdot \#\mathcal{B}}$$

We need Lift

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$\#\mathcal{A}\mathcal{B}$	$\#\mathcal{A}\overline{\mathcal{B}}$	$\#\mathcal{A}$
$\mathcal{A}$ is not included	$\#\overline{\mathcal{A}}\mathcal{B}$	$\#\overline{\mathcal{A}\mathcal{B}}$	$\#\overline{\mathcal{A}}$
	$\#\mathcal{B}$	$\#\overline{\mathcal{B}}$	$\#\mathrm{ALL}$

$$\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A}) \cdot \mathrm{support}(\mathcal{B})} = \frac{P(\mathcal{A} \cup \mathcal{B})}{P(\mathcal{A}) \cdot P(\mathcal{B})} = \frac{\frac{\#\mathcal{A}\mathcal{B}}{\#\mathrm{ALL}}}{\frac{\#\mathcal{A}}{\#\mathrm{ALL}} \cdot \frac{\#\mathcal{B}}{\#\mathrm{ALL}}}$$

If $\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) \approx 1$ then $\mathcal{A}$ and $\mathcal{B}$ are **independent** $P(\mathcal{A} \cup \mathcal{B}) \approx P(\mathcal{A}) \cdot P(\mathcal{B})$

If $\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) \ll 1$ then $\mathcal{A}$ and $\mathcal{B}$ are **negatively correlated** $P(\mathcal{A} \cup \mathcal{B}) \ll P(\mathcal{A}) \cdot P(\mathcal{B})$

If $\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) \gg 1$ then $\mathcal{A}$ and $\mathcal{B}$ are **positively correlated** $P(\mathcal{A} \cup \mathcal{B}) \gg P(\mathcal{A}) \cdot P(\mathcal{B})$

Is the Rule Surprising?

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	9	1	10
$\mathcal{A}$ is not included	81	9	90
	90	10	100

$$\text{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\text{support}(\mathcal{A} \cup \mathcal{B})}{\text{support}(\mathcal{A}) \cdot \text{support}(\mathcal{B})} = \frac{P(\mathcal{A} \cup \mathcal{B})}{P(\mathcal{A}) \cdot P(\mathcal{B})} = \frac{\frac{\#\mathcal{AB}}{\#\text{ALL}}}{\frac{\#\mathcal{A}}{\#\text{ALL}} \cdot \frac{\#\mathcal{B}}{\#\text{ALL}}}$$

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{100}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{10}$$

$$\text{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\frac{9}{100}}{\frac{10}{100} \cdot \frac{90}{100}} = 1$$

No suprise!

Is the Rule Surprising?

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	9	1	10
$\mathcal{A}$ is not included	**0**	**90**	90
	9	91	100

$$\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\mathrm{support}(\mathcal{A} \cup \mathcal{B})}{\mathrm{support}(\mathcal{A}) \cdot \mathrm{support}(\mathcal{B})} = \frac{P(\mathcal{A} \cup \mathcal{B})}{P(\mathcal{A}) \cdot P(\mathcal{B})} = \frac{\frac{\#\mathcal{A}\mathcal{B}}{\#\mathrm{ALL}}}{\frac{\#\mathcal{A}}{\#\mathrm{ALL}} \cdot \frac{\#\mathcal{B}}{\#\mathrm{ALL}}}$$

$$\mathrm{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{100}$$

$$\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{10}$$

$$\mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\frac{9}{100}}{\frac{10}{100} \cdot \frac{9}{100}} = 10$$

Surprise!

Is the Rule Surprising?

Consider association rule $\mathcal{A} \Rightarrow \mathcal{B}$

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	9	1	10
$\mathcal{A}$ is not included	**90**	**0**	90
	99	1	100

$$\text{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\text{support}(\mathcal{A} \cup \mathcal{B})}{\text{support}(\mathcal{A}) \cdot \text{support}(\mathcal{B})} = \frac{P(\mathcal{A} \cup \mathcal{B})}{P(\mathcal{A}) \cdot P(\mathcal{B})} = \frac{\frac{\#\mathcal{AB}}{\#\text{ALL}}}{\frac{\#\mathcal{A}}{\#\text{ALL}} \cdot \frac{\#\mathcal{B}}{\#\text{ALL}}}$$

$$\text{support}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{100}$$

$$\text{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{9}{10}$$

$$\text{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\frac{9}{100}}{\frac{10}{100} \cdot \frac{99}{100}} = \frac{10}{11}$$

a little bit …

Selecting Association rules

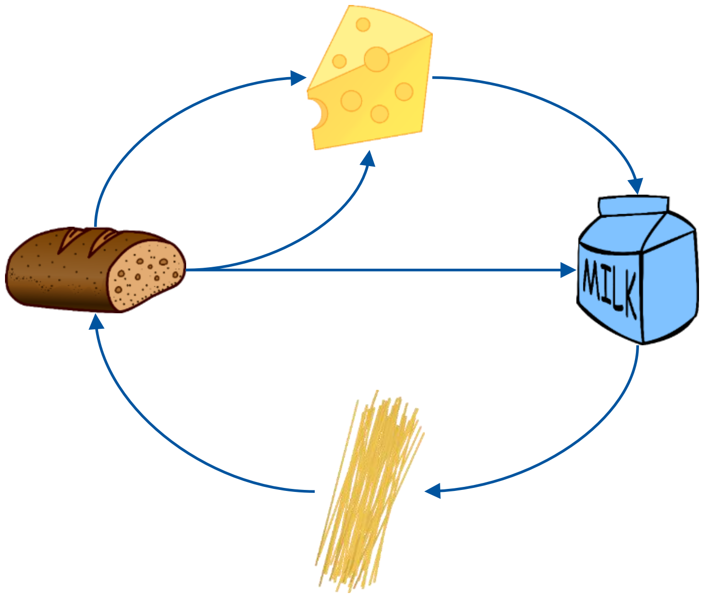
1. Set thresholds for minimal support and confidence
2. Evaluate lift and possibly other metrics for the rules remaining
3. Sort and prune based on any of the quality criteria (support, confidence, lift, etc.)

It is hard to predict the number of rules beforehand

There are many other measures of quality (conviction, leverage, collective strength, etc.)

Association Rules

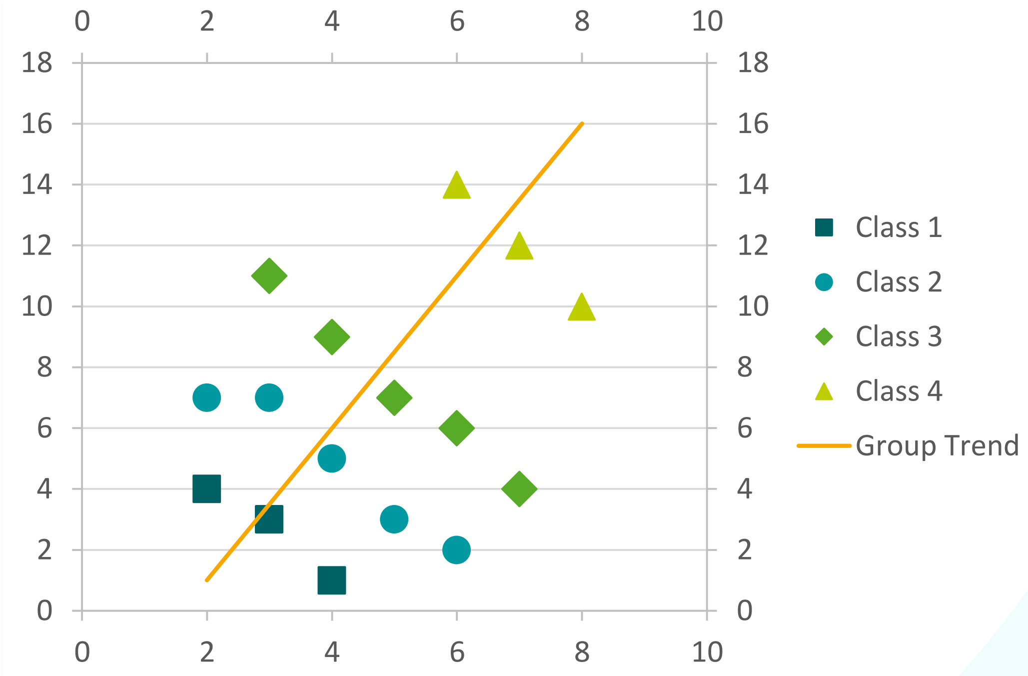
1. Introduction
2. Generating Association Rules
3. Applications
4. Evaluation
5. **Simpson's Paradox**



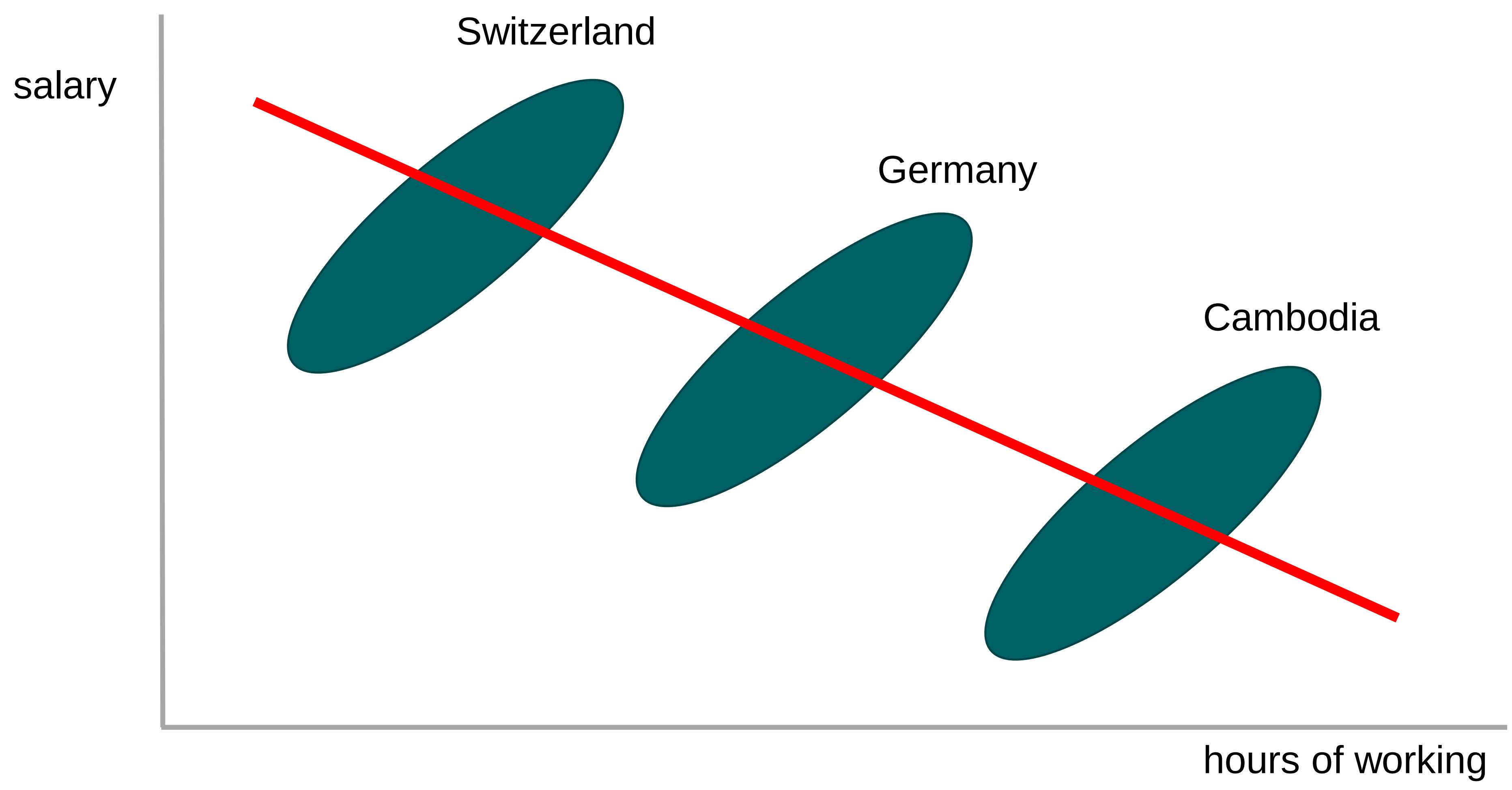
Simpson’s Paradox

A trend appears in several different groups of data but disappears or reverses when these groups are combined.

- Edward Simpson in 1951 (earlier variants by Udny Yule and Karl Pearson)
- Nice example of ‘How to lie with statistics?’
- The paradox is often encountered in social-science and medical-science



Simpson’s Paradox When Using Regression



Simpson's Paradox in Association Rules

Consider the association rule $\mathcal{A} \Rightarrow \mathcal{B}$ and any feature which splits the instances (location, age …)

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$a + p$	$(b - a) + (q - p)$	$b + q$
$\mathcal{A}$ is not included	$c + r$	$(d - c) + (s - r)$	$d + s$
	$a + c + p + r$	$(b + d + q + s) - (a + c + p + r)$	$b + d + q + s$

Two classes – blue and orange
(e.g., old and young)

Simpson's Paradox in Association Rules

Consider the association rule $\mathcal{A} \Rightarrow \mathcal{B}$ and any feature which splits the instances (location, age …)

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$a + p$	$(b - a) + (q - p)$	$b + q$
$\mathcal{A}$ is not included	$c + r$	$(d - c) + (s - r)$	$d + s$
	$a + c + p + r$	$(b + d + q + s) - (a + c + p + r)$	$b + d + q + s$

$$\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{a+p}{b+q} \qquad \mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{\frac{a+p}{b+d+q+s}}{\frac{b+q}{b+d+q+s} \cdot \frac{a+c+p+r}{b+d+q+s}}$$

Simpson's Paradox in Association Rules

Consider the association rule $\mathcal{A} \Rightarrow \mathcal{B}$ and any feature which splits the instances (location, age …)

$\mathcal{A} \Rightarrow \mathcal{B}$	$\mathcal{B}$ is included	$\mathcal{B}$ is not included	
$\mathcal{A}$ is included	$a + p$	$(b - a) + (q - p)$	$b + q$
$\mathcal{A}$ is not included	$c + r$	$(d - c) + (s - r)$	$d + s$
	$a + c + p + r$	$(b + d + q + s) - (a + c + p + r)$	$b + d + q + s$

$$\mathrm{conf}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{a+p}{b+q} \qquad \mathrm{lift}(\mathcal{A} \Rightarrow \mathcal{B}) = \frac{(a+p)\cdot(b+d+q+s)}{(b+q)\cdot(a+c+p+r)}$$

Simpson's Paradox - Example

Two classes: old and young

smoke $\Rightarrow$ cancer	has cancer	doesn't have cancer	
smokes	$1 + 66$	$2 + 34$	$3 + 100$
doesn't smoke	$34 + 2$	$66 + 1$	$100 + 3$
	$35 + 68$	$68 + 35$	$103 + 103$

humans $\mathrm{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{67}{103} = 0.65 > \mathrm{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{36}{103} = 0.35$

old $\mathrm{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{1}{3} = 0.333 < \mathrm{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{34}{100} = 0.34$

young $\mathrm{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{66}{100} = 0.66 < \mathrm{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{2}{3} = 0.666$

Simpson's Paradox - Example

Two classes: old and young

smoke $\Rightarrow$ cancer	has cancer	doesn't have cancer	
smokes	$1+66$	$2+34$	$3+100$
doesn't smoke	$34+2$	$66+1$	$100+3$
	$35+68$	$68+35$	$103+103$

humans $\text{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{67}{103} = 0.65 > \text{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{36}{103} = 0.35$

old $\text{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{1}{3} = 0.333 < \text{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{34}{100} = 0.34$

young $\text{conf}(\text{smoke} \Rightarrow \text{cancer}) = \frac{66}{100} = 0.66 < \text{conf}(\text{not smoke} \Rightarrow \text{cancer}) = \frac{2}{3} = 0.666$

Smoking is healthy for old and young people, but not for all humans!


[1]

Simpson's Paradox - Example

Two classes: old and young

smoke $\Rightarrow$ cancer	has cancer	doesn't have cancer	
smokes	$1 + 66$	$2 + 34$	$3 + 100$
doesn't smoke	$34 + 2$	$66 + 1$	$100 + 3$
	$35 + 68$	$68 + 35$	$103 + 103$

- The presence of smoking has a strong positive effect on the occurrence of cancer in the overall set (supports the rule)
- However, the effect cannot be seen in the subsets!

Simpson's Paradox - Example

Two classes: old and young

smoke $\Rightarrow$ cancer	has cancer	doesn't have cancer	
smokes	$1 + 66$	$2 + 34$	$3 + 100$
doesn't smoke	$34 + 2$	$66 + 1$	$100 + 3$
	$35 + 68$	$68 + 35$	$103 + 103$

humans $\quad \text{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{67}{206}}{\frac{103}{206} \cdot \frac{103}{206}} = \frac{67 \cdot 206}{103 \cdot 103} = 1.301$

old $\quad \text{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{1}{103}}{\frac{3}{103} \cdot \frac{35}{103}} = \frac{1 \cdot 103}{3 \cdot 35} = 0.9809$

young $\quad \text{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{66}{103}}{\frac{100}{103} \cdot \frac{68}{103}} = \frac{66 \cdot 103}{100 \cdot 68} = 0.9997$

Simpson's Paradox - Example

Two classes: old and young

$\text{smoke} \Rightarrow \text{cancer}$	has cancer	doesn't have cancer	
smokes	$1 + 66$	$2 + 34$	$3 + 100$
doesn't smoke	$34 + 2$	$66 + 1$	$100 + 3$
	$35 + 68$	$68 + 35$	$103 + 103$

humans $\mathrm{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{67}{206}}{\frac{103}{206} \cdot \frac{103}{206}} = \frac{67 \cdot 206}{103 \cdot 103} = 1.301$

old $\mathrm{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{1}{103}}{\frac{3}{103} \cdot \frac{35}{103}} = \frac{1 \cdot 103}{3 \cdot 35} = 0.9809$

young $\mathrm{lift}(\text{smoke} \Rightarrow \text{cancer}) = \frac{\frac{66}{103}}{\frac{100}{103} \cdot \frac{68}{103}} = \frac{66 \cdot 103}{100 \cdot 68} = 0.9997$

Positively correlated

Negatively correlated

Simpson’s Paradox – Another Example

	Computer Science		Mathematics		ALL	
	get degree	drop out	get degree	drop out	get degree	drop out
female	80 (80%)	20 (20%)	400 (40%)	600 (60%)	480 (44%)	620 (56%)
male	700 (70%)	300 (30%)	30 (30%)	70 (70%)	730 (66%)	370 (34%)

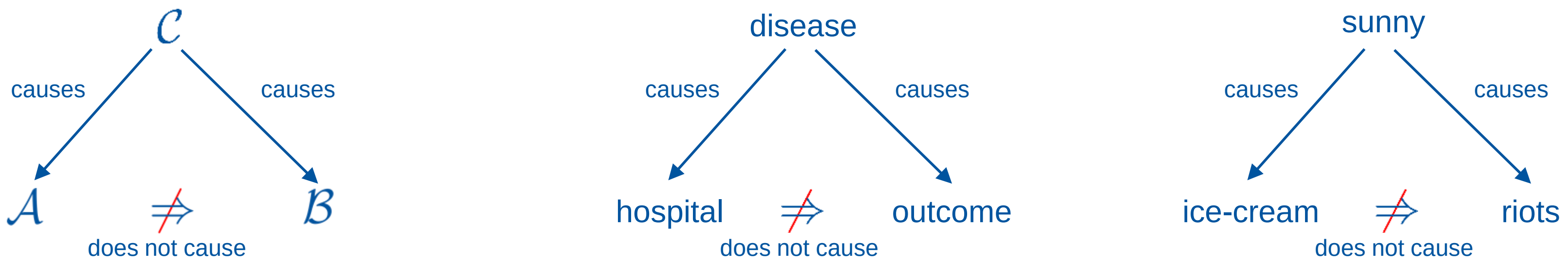
1100 females and 1100 males, 1100 CS students and 1100 math students

Simpson’s Paradox – Other Examples

- The hospital in the city of Stolberg has an overall better performance (e.g., lower mortality rate) than the hospital in Aachen. However, for any specific disease, Aachen performs better. This paradox is due to different distributions of diseases (patients with more serious diseases tend to end up in Aachen and not Stolberg).
- Males have higher wages on average, but in any given profession, females earn more on average. This paradox is explained by males going for higher-paid professions.
- Low birth-weight paradox: low birth-weight children born to smoking mothers have a lower infant mortality rate than low-birth-weight children of non-smokers. Smoking is harmful and contributes to low birth weight and higher mortality than normal birth weight. However, other causes of low birth weight are generally more harmful than smoking.

Confounding

- Simpsons paradox is related to confounding, i.e., another (possibly hidden) feature that influences two other features
- A confounding feature $\mathcal{C}$ (also called "lurking variable") may influence both $\mathcal{A}$ and $\mathcal{B}$, and therefore "blur" $\mathcal{A} \Rightarrow \mathcal{B}$

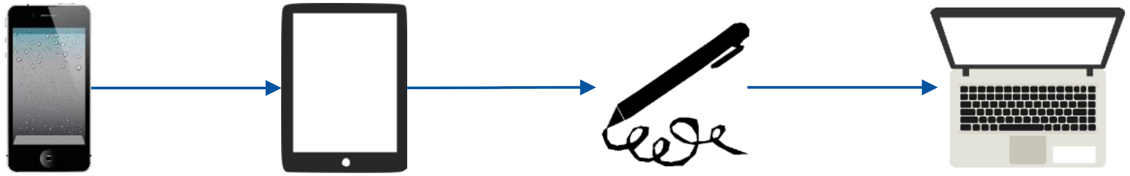


Summary

- Association rules can be discovered starting from frequent items sets $\mathcal{A} \Rightarrow \mathcal{B}$
- Any dataset with instances and feature values can be turned into a multiset of itemsets and used for association rule mining (not just "pure itemsets")
- Support, confidence, and lift can be used to prune and sort association rules
- Rules should be interpreted carefully (Simpson's paradox and confounders)

Sequence Mining

1. **Temporal Data**
2. Measuring Support
3. Apriori-All Algorithm
4. Extensions and Conclusion



Temporal Data – Discrete Timestamped Events

Time-stamp	f_1	f_2	f_3	f_4	…	f_D
t_1						
t_2						
t_3						
t_4						
t_5						
…						

Every instance happened at a specific time

Temporal Data – Discrete Timestamped Events

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Temporal Data – Discrete Timestamped Events

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Event Data

- Timestamp (typically not equal intervals)
- Case ID (maps events to cases)
- Activity (identifies the event type)
- Other features are optional (resource, location, cost, duration, …)

Temporal Data – Discrete Timestamped Events

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Event Data

- Timestamp (typically not equal intervals)
- Case ID (maps events to cases)
- Activity (identifies the event type)
- Other features are optional (resource, location, cost, duration, …)

Case 1: $\langle a, b, \ldots \rangle$

Case 2: $\langle a, \ldots \rangle$

Case 3: $\langle a, b, \ldots \rangle$

Temporal Data – Discrete Timestamped Events

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Event Data

- Timestamp (typically not equal intervals)
- Case ID (maps events to cases)
- Activity (identifies the event type)
- Other features are optional (resource, location, cost, duration, …)

Case 1: $\langle a, b, \ldots \rangle$
Case 2: $\langle a, \ldots \rangle$
Case 3: $\langle a, b, \ldots \rangle$

We can abstract from timestamps and optional features to obtain sequences of activities

Temporal Data – Discrete Timestamped Events

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Event Data

- Timestamp (typically not equal intervals)
- Case ID (maps events to cases)
- Activity (identifies the event type)
- Other features are optional (resource, location, cost, duration, …)

Case 1: $\langle a, b, \ldots \rangle$

Case 2: $\langle a, \ldots \rangle$

Case 3: $\langle a, b, \ldots \rangle$

$\Rightarrow [\langle a, b, \ldots \rangle^2, \langle a, \ldots \rangle]$

Event Data – Example 1

Case ID	Activity name	Timestamp	Other features	
Patient ID	**Activity**	**Time**	**Doctor**	**Age**
5611	Blood Test	12:25	Dr. Scott	45
3645	X-Ray	14:34	Dr. House	67
5611	Surgery	15:01	Dr. Scott	45
7891	Blood Test	15:03	Dr. House	24
3645	Radiation Therapy	17:25	Dr. Jenna	81
…	…	…	…	…

$5611 : \langle \text{Blood Test}, \text{Surgery}, \ldots \rangle$

$3645 : \langle \text{X-Ray}, \text{Radiation Therapy}, \ldots \rangle$

$7891 : \langle \text{Blood Test}, \ldots \rangle$

Event Data – Example 2

Case ID	Activity name	Timestamp	Other features		
Order Number	**Activity**	**Time**	**Username**	**Product**	**Quantity**
11152	Register Order	15.12.22 12:25	Carrie192	Iphone 14	1
52690	Ship Order	15.12.22 12:45	Johnny1	Earpods	2
11152	Check Stock	15.12.22 13:01	Carrie192	Iphone 14	1
44891	Handle Payment	30.12.22 18:01	Obelisk	USB-C Charger	3
61238	Cancel Order	11.01.23 17:25	Apex_512	MacBook Air	1
…	…	…	…	…	…

$$11152 : \langle \text{Register Order}, \text{Check Stock}, \text{Cancel Order}, \ldots \rangle$$

$$52690 : \langle \text{Ship Order}, \ldots \rangle$$

$$44891 : \langle \text{Handle Payment}, \ldots \rangle$$

Note: 'Username' could also be our Case ID, changing the meaning of data!

Event Data – Example 2

Case ID	Activity name	Timestamp	Other features		
Order Number	**Activity**	**Time**	**Username**	**Product**	**Quantity**
11152	Register Order	15.12.22 12:25	Carrie192	Iphone 14	1
52690	Ship Order	15.12.22 12:45	Johnny1	Earpods	2
11152	Check Stock	15.12.22 13:01	Carrie192	Iphone 14	1
44891	Handle Payment	30.12.22 18:01	Obelisk	USB-C Charger	3
61238	Cancel Order	11.01.23 17:25	Apex_512	MacBook Air	1
…	…	…	…	…	…

$11152 : \langle \text{Register Order}, \text{Check Stock}, \text{Cancel Order}, \ldots \rangle$

$52690 : \langle \text{Ship Order}, \ldots \rangle$

$88721 : \langle \text{Register Order}, \text{Check Stock}, \text{Cancel Order}, \ldots \rangle$

Note: the same sequence can occur multiple times for different cases (multiset of sequences)

Event data – Basis for Process Mining

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

Process Mining

- Processes generate event data
- Every process execution is a case

Common Tasks

- Discover the process
- Validate the process
- Improve the process

Temporal Data – Discrete Timestamped Events

Generalized sequential data

Time-stamp	Case ID	Item
t_1	3	a
t_2	1	a
t_3	1	b
t_4	2	a
t_5	3	b
…	…	…

Item Identifier

Sequential Data

- Timestamp (typically not equal intervals)
- Case ID (maps events to cases)
- Item (identifies the item type)

Relation to event data:
item could be an activity

Temporal Data – Discrete Timestamped Events

Generalized sequential data

Timestamp	Customer ID	Purchased Item
22-07-12	1172	Razor
22-07-12	8121	Shampoo
22-07-12	1172	Shaving Cream
22-08-13	3434	Shampoo
22-09-01	1172	Shaving Cream
…	…	…

$$1172 : \langle \text{Razor}, \text{Shaving Cream}, \text{Shaving Cream} \rangle$$
$$8121 : \langle \text{Shampoo} \rangle$$
$$3434 : \langle \text{Shampoo} \rangle$$
$$\dots$$
$$\Rightarrow [\langle \text{Razor}, \text{Shaving Cream}, \text{Shaving Cream} \rangle, \langle \text{Shampoo} \rangle^2, \dots]$$

Temporal Data – Discrete Timestamped Events

Generalized sequential data

Timestamp	Customer ID	Purchased Item
22-07-12	1172	Razor
22-07-12	8121	Shampoo
22-07-12	1172	Shaving Cream
22-08-13	3434	Shampoo
22-09-01	1172	Shaving Cream
…	…	…

$$1172 : \langle \text{Razor}, \text{Shaving Cream}, \text{Shaving Cream} \rangle$$
$$8121 : \langle \text{Shampoo} \rangle$$
$$3434 : \langle \text{Shampoo} \rangle$$
$$\dots$$
$$\Rightarrow [\langle \text{Razor}, \text{Shaving Cream}, \text{Shaving Cream} \rangle, \langle \text{Shampoo} \rangle^2, \dots]$$

⇔

Timestamp	Customer ID	Purchased Itemset
22-07-12	1172	Razor, Shaving Cream
22-07-12	8121	Shampoo
22-08-13	3434	Shampoo
22-09-01	1172	Shaving Cream
…	…	…

$$1172 : \langle \{\text{Razor}, \text{Shaving Cream}\}, \{\text{Shaving Cream}\} \rangle$$
$$8121 : \langle \{\text{Shampoo}\} \rangle$$
$$3434 : \langle \{\text{Shampoo}\} \rangle$$
$$\dots$$
$$\Rightarrow [\langle \{\text{Razor}, \text{Shaving Cream}\}, \{\text{Shaving Cream}\} \rangle, \langle \{\text{Shampoo}\} \rangle^2, \dots]$$

Temporal Data – Discrete Timestamped Events

Generalized sequential data

Sequential Pattern Mining

- Input: a multiset of nonempty sequences of itemsets
- Main analysis question: identify frequent subsequences (recurring patterns)
- Relation to event data:
  an itemset can be interpreted as activity,
  an activity can be an itemset of size 1

Timestamp	Customer ID	Purchased Itemset
22-07-12	1172	Razor, Shaving Cream
22-07-12	8121	Shampoo
22-08-13	3434	Soap
22-09-01	1172	Shaving Cream
…	…	…

$$1172 : \langle\{\text{Razor}, \text{Shaving Cream}\}, \{\text{Shaving Cream}\}\rangle$$

$$8121 : \langle\{\text{Shampoo}\}\rangle$$

$$3434 : \langle\{\text{Shampoo}\}\rangle$$

$$\ldots$$

$$\Rightarrow [\langle\{\text{Razor}, \text{Shaving Cream}\}, \{\text{Shaving Cream}\}\rangle, \langle\{\text{Shampoo}\}\rangle^2, \ldots]$$

Sequential Pattern Mining

- Uses a specific type of (event) data as input: multiset of sequences of itemsets
- A sequence is a nonempty sequence of itemsets
- Two notations for sequence data:

Itemset (Activity)

  – Formal:

$$\mathcal{X} = [\langle\{a\},\{b\},\{c,d\},\{e\}\rangle, \langle\{a\},\{b\},\{c,d\},\{e\}\rangle, \langle\{a\},\{b,c\},\{c,d,e\},\{f\}\rangle]$$

  – Informal (short notation):

$$\mathcal{X} = [ab(cd)e, ab(cd)e, a(cd)e, a(bc)(cde)f]$$

A sequence of itemsets (activities):
e.g., $\{c,d,e\}$ happened **after** $\{b,c\}$

- Formally $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ for a set of items $\mathcal{I}$

  ($\mathbb{M}$ is the multiset and $\mathbb{P}$ the powerset operator)

Sequential Pattern Mining – Input Example

Customer ID	Purchased Items	Time
1	A	15.12.22 12:25
1	A, B	15.12.22 12:45
2	B	15.12.22 13:01
3	C	30.12.22 18:01
3	A, C, D	11.01.23 17:25
4	B	31.12.22 17:32
…	…	…

⇒

Customer ID	Customer Sequence
1	$\langle\{A\},\{A,B\}\rangle$
2	$\langle\{B\}\rangle$
3	$\langle\{C\},\{A,C,D\}\rangle$
4	$\langle\{B\}\rangle$
…	…

Input is a **multiset of sequences of itemsets**

Sequential Pattern Mining – Input

Input $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$

- Formal:

$$[\langle\{A\},\{A,B\}\rangle,\langle\{B\}\rangle,\langle\{C\},\{A,C,D\}\rangle,\langle\{B\}\rangle]$$
$$=[\langle\{A\},\{A,B\}\rangle,\langle\{B\}\rangle^2,\langle\{C\},\{A,C,D\}\rangle]$$

- Informal:

$$[A(AB),B,C(ACD),B]$$
$$=[A(AB),B^2,C(ACD)]$$

Customer ID	Customer Sequence
1	$\langle\{A\},\{A,B\}\rangle$
2	$\langle\{B\}\rangle$
3	$\langle\{C\},\{A,C,D\}\rangle$
4	$\langle\{B\}\rangle$
…	…

Input is a **multiset of sequences of itemsets**

NESPRESSO

Willkommen Wil van der Aalst | IHR WARENKORB (0)

Kaffee | Nespresso & You | Maschinen | Accessories | Geschenke | Our Choices | Nachhaltigkeit | Storefinder | Service | FAQ | Professional

$$\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$$

MEIN KONTO

Willkommen Wil van der Aalst

Mitglied seit 21-09-2018

Kundennummer: 3868857

- Meine Bestellungen
- Meine Adressen
- Meine persönlichen Daten
- Meine Maschinen
- Benachrichtigungen & Erinnerungen
- Marketingpräferenzen
- Express Checkout
- Mein Kaffee Abo

Meine Bestellungen

BESTELLDATUM	STATUS	QUELLE	Liefermethode	BESTELLNUMMER	BETRAG
01/11/2018	Geliefert	Internet	Standardlieferung - Lieferung am nächsten Werktag	25250378	97,80 €
21/09/2018	Geliefert	Internet	Standardlieferung innerhalb von 2 Werktagen	24533528	78,60 €

Mehr Bestellungen anzeigen >

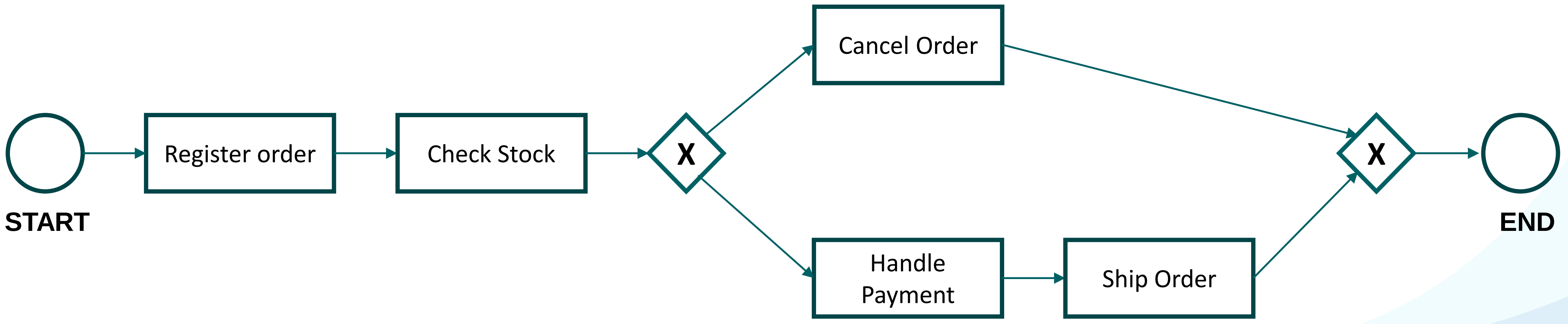
Bestellung - 01/11/2018

Wieder bestellen

Kapseln (250)	Stückpreis		Menge	Gesamt
Ristretto	0,38 €	x	30	11,40 €
Roma	0,38 €	x	80	30,40 €
Vivalto Lungo	0,40 €	x	50	20,00 €
Linizio Lungo	0,40 €	x	80	32,00 €
Ristretto Decaffeinato	0,40 €	x	10	4,00 €

Temporal Data – Analysis Techniques

- This lecture – Sequential Pattern Mining
- Next lectures – Time Series and Process Mining:
  - Analyze and predict time series data
  - Discover, validate and improve processes



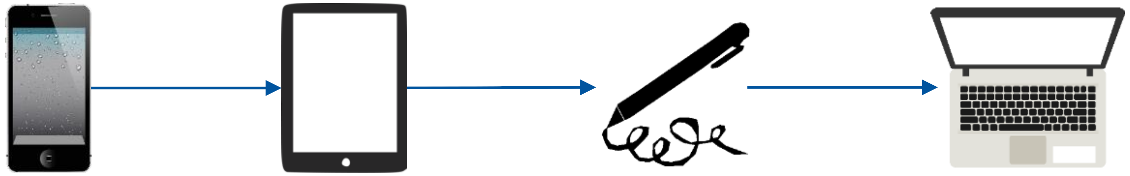
Sequence Mining

1. Temporal Data

2. **Measuring Support**

3. Apriori-All Algorithm

4. Extensions and Conclusion



Goal – Find Frequent Sequential Patterns

Customer ID	Customer Sequence
1	$\langle\{11\},\{25\}\rangle$
2	$\langle\{31\}\rangle$
3	$\langle\{12\},\{11\}\rangle$
…	…

Sequential Patterns with Support > Threshold (Min_Sup)
$\langle\{31\}\rangle$
$\langle\{12\},\{11\}\rangle$
…

- Given a dataset $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ find all frequent sequential patterns
- Sequential pattern $\mathcal{P}$ is a sequence of itemsets, i.e., $\mathcal{P} \in (\mathbb{P}(\mathcal{I}))^*$
- Support of a sequential pattern is the fraction of sequences in $\mathcal{X}$ that contain the pattern $\mathcal{P}$

Containment

- Let $\mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n \rangle \in (\mathbb{P}(\mathcal{I}))^*$ and $\mathcal{B} = \langle \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_m \rangle \in (\mathbb{P}(\mathcal{I}))^*$ be two itemset sequences
- $\mathcal{A}$ is contained in $\mathcal{B}$ if there exist integers $1 \leq i_1 < i_2 < \cdots < i_n \leq m$ such that

$$\mathcal{A}_1 \subseteq \mathcal{B}_{i_1}, \mathcal{A}_2 \subseteq \mathcal{B}_{i_2}, \ldots, \mathcal{A}_n \subseteq \mathcal{B}_{i_n}$$

Containment

- Let $\mathcal{A} = \langle \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n \rangle \in (\mathbb{P}(\mathcal{I}))^*$ and $\mathcal{B} = \langle \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_m \rangle \in (\mathbb{P}(\mathcal{I}))^*$ be two itemset sequences
- $\mathcal{A}$ is contained in $\mathcal{B}$ if there exist integers $1 \leq i_1 < i_2 < \cdots < i_n \leq m$ such that

$$\mathcal{A}_1 \subseteq \mathcal{B}_{i_1}, \mathcal{A}_2 \subseteq \mathcal{B}_{i_2}, \ldots, \mathcal{A}_n \subseteq \mathcal{B}_{i_n}$$

$$\langle \mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4 \rangle$$

$\mathcal{A}_1 \subseteq \mathcal{B}_1$ $\quad \mathcal{A}_2 \subseteq \mathcal{B}_4$ $\quad \mathcal{A}_3 \subseteq \mathcal{B}_7$ $\quad \mathcal{A}_4 \subseteq \mathcal{B}_8$

$$\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5, \mathcal{B}_6, \mathcal{B}_7, \mathcal{B}_8, \mathcal{B}_9 \rangle$$

$$\boxed{1 \leq 1 < 4 < 7 < 8 \leq 9}$$

Containment

- Notation: $\mathcal{A} \sqsubseteq \mathcal{B}$ if $\mathcal{A}$ is contained in $\mathcal{B}$
- If $\mathcal{A} \sqsubseteq \mathcal{B}$, then $\mathcal{A}$ is a subsequence of $\mathcal{B}$ and $\mathcal{B}$ is a supersequence of $\mathcal{A}$

$$\mathcal{A}_1 \subseteq \mathcal{B}_{i_1}, \mathcal{A}_2 \subseteq \mathcal{B}_{i_2}, \ldots, \mathcal{A}_n \subseteq \mathcal{B}_{i_n}$$

subsequence $\langle \mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_4 \rangle$

$$\mathcal{A} \sqsubseteq \mathcal{B}$$

$\mathcal{A}_1 \subseteq \mathcal{B}_2$ $\mathcal{A}_2 \subseteq \mathcal{B}_4$ $\mathcal{A}_3 \subseteq \mathcal{B}_7$ $\mathcal{A}_4 \subseteq \mathcal{B}_8$

supersequence $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4, \mathcal{B}_5, \mathcal{B}_6, \mathcal{B}_7, \mathcal{B}_8, \mathcal{B}_9 \rangle$

Containment - Examples

Formal notation:

- $\langle\{\mathrm{a}\},\{\mathrm{a},\mathrm{b}\},\{\mathrm{b},\mathrm{c}\},\{\mathrm{c}\}\rangle \sqsubseteq \langle\{\mathrm{a}\},\{\mathrm{a}\},\{\mathrm{a},\mathrm{b},\mathrm{c}\},\{\mathrm{b},\mathrm{c}\},\{\mathrm{b},\mathrm{c}\},\{\mathrm{a},\mathrm{c}\}\rangle$

aa(abc)(bc)(bc)(ac)

a(ab)(bc)c

multiple mappings possible

Informal notation:

- a(ab)(bc)c ⊑ aa(abc)(bc)(bc)(ac)

$\langle \boldsymbol{a_1}, \boldsymbol{a_2}, \ldots, \boldsymbol{a_n} \rangle \sqsubseteq \langle \boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_m} \rangle$ if and only if there exist integers $1 \le \boldsymbol{i_1} < \boldsymbol{i_2} < \cdots < \boldsymbol{i_n} \le m$ such that $\boldsymbol{a_1} \subseteq \boldsymbol{b_{i_1}}, \boldsymbol{a_2} \subseteq \boldsymbol{b_{i_2}}, \ldots, \boldsymbol{a_n} \subseteq \boldsymbol{b_{i_n}}$

Containment - Examples

Formal notation:

- $\langle\{a\},\{a,b\},\{b,c\},\{c\}\rangle \sqsubseteq \langle\{a\},\{a\},\{a,b,c\},\{b,c\},\{b,c\},\{a,c\}\rangle$
- $\langle\{a\},\{a,b\},\{b,c\},\{c\}\rangle \not\sqsubseteq \langle\{a\},\{a,b,c\},\{b,d\},\{b,e\},\{a,c\}\rangle$

a(abc)(bd)(be)(ac)

?

a(ab)(bc)c

Informal notation:

- a(ab)(bc)c $\sqsubseteq$ aa(abc)(bc)(bc)(ac)
- a(ab)(bc)c $\not\sqsubseteq$ a(abc)(bd)(be)(ac)

!

$\langle \boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_n}\rangle \sqsubseteq \langle \boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_m}\rangle$ if and only if there exist integers $1 \leq \boldsymbol{i_1} < \boldsymbol{i_2} < \cdots < \boldsymbol{i_n} \leq m$ such that $\boldsymbol{a_1} \subseteq \boldsymbol{b_{i_1}}, \boldsymbol{a_2} \subseteq \boldsymbol{b_{i_2}}, \dots, \boldsymbol{a_n} \subseteq \boldsymbol{b_{i_n}}$

Containment – Practice Questions

- (ab)(bc) $\sqsubseteq$ (bc)(ab) **?**
- ab $\sqsubseteq$ a(ac)(bc)c **?**
- aa(ab)(bc) $\sqsubseteq$ (ab)(ace)(bce)(ab) **?**
- (abc)ef $\sqsubseteq$ (ab)(bc)(ef)f **?**
- (abc)ef $\sqsubseteq$ (ab)(bc)(abcd)(ef)f **?**

$\langle \boldsymbol{a_1}, \boldsymbol{a_2}, \ldots, \boldsymbol{a_n} \rangle \sqsubseteq \langle \boldsymbol{b_1}, \boldsymbol{b_2}, \ldots, \boldsymbol{b_m} \rangle$ if and only if there exist integers $1 \le \boldsymbol{i_1} < \boldsymbol{i_2} < \cdots < \boldsymbol{i_n} \le m$ such that $\boldsymbol{a_1} \subseteq \boldsymbol{b_{i_1}}, \boldsymbol{a_2} \subseteq \boldsymbol{b_{i_2}}, \ldots, \boldsymbol{a_n} \subseteq \boldsymbol{b_{i_n}}$

Containment – Practice Answers

- (ab)(bc) $\not\sqsubseteq$ (bc)(ab) (incompatible order)
- ab $\sqsubseteq$ a(ac)(bc)c
- aa(ab)(bc) $\not\sqsubseteq$ (ab)(ace)(bce)(ab) ((ab) cannot be mapped without also handling aa or (bc), etc.)
- (abc)ef $\not\sqsubseteq$ (ab)(bc)(ef)f (no match for (abc))
- (abc)ef $\sqsubseteq$ (ab)(bc)(abcd)(ef)f

$\langle \boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_n} \rangle \sqsubseteq \langle \boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_m} \rangle$ if and only if there exist integers $1 \leq \boldsymbol{i_1} < \boldsymbol{i_2} < \cdots < \boldsymbol{i_n} \leq m$ such that $\boldsymbol{a_1} \subseteq \boldsymbol{b_{i_1}}, \boldsymbol{a_2} \subseteq \boldsymbol{b_{i_2}}, \dots, \boldsymbol{a_n} \subseteq \boldsymbol{b_{i_n}}$

Support

- The support of a sequential pattern $\mathcal{P}$ is the fraction of sequences in $\mathcal{X}$ that contain $\mathcal{P}$
- $\text{support}(\mathcal{P}) = \frac{|[\mathcal{S} \in \mathcal{X} | \mathcal{P} \sqsubseteq \mathcal{S}]|}{|\mathcal{X}|}$
- Minimum support threshold ***min_sup*** defines which sequences are frequent

Support

- The support of a sequential pattern $\mathcal{P}$ is the fraction of sequences in $\mathcal{X}$ that contain $\mathcal{P}$
- $\text{support}(\mathcal{P}) = \frac{|[\mathcal{S} \in \mathcal{X} | \mathcal{P} \sqsubseteq \mathcal{S}]|}{|\mathcal{X}|}$
- Minimum support threshold ***min_sup*** defines which sequences are frequent
- Support count is the number of sequences in $\mathcal{X}$ that contain $\mathcal{P}$
- $\text{support\_count}(\mathcal{P}) = |[\mathcal{S} \in \mathcal{X} \mid \mathcal{P} \sqsubseteq \mathcal{S}]|$

Support – Practice Questions

- $\mathcal{X} = [abcd^2, (abcd), (ab)(cd)^3, (ab)(bc)(cd)]$
- What is the $\text{support\_count}(\mathcal{P})$ for
  - $\mathcal{P}$ = a
  - $\mathcal{P}$ = ab
  - $\mathcal{P}$ = (ab)
  - $\mathcal{P}$ = (ab)c
  - $\mathcal{P}$ = (ab)(bd)
  - $\mathcal{P}$ = ab(cd)

$$
\begin{aligned}
= [&\langle\{a\},\{b\},\{c\},\{d\}\rangle^2,\\
&\langle\{a,b,c,d\}\rangle,\\
&\langle\{a,b\}\{c,d\}\rangle^3,\\
&\langle\{a,b\},\{b,c\},\{c,d\}\rangle]
\end{aligned}
$$

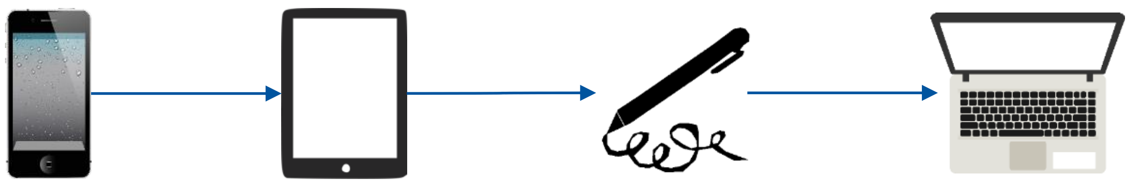
$$
\begin{aligned}
= [&\langle\{a\},\{b\},\{c\},\{d\}\rangle,\\
&\langle\{a,b,c,d\}\rangle,\\
&\langle\{a,b\}\{c,d\}\rangle,\\
&\langle\{a,b\},\{b,c\},\{c,d\}\rangle,\\
&\langle\{a\},\{b\},\{c\},\{d\}\rangle,\\
&\langle\{a,b\}\{c,d\}\rangle,\\
&\langle\{a,b\}\{c,d\}\rangle]
\end{aligned}
$$

Support – Practice Questions

- $\mathcal{X} = [abcd^2, (abcd), (ab)(cd)^3, (ab)(bc)(cd)]$
- What is the $\text{support\_count}(\mathcal{P})$ for
  - $\mathcal{P}$ = a — 7 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]
  - $\mathcal{P}$ = ab — 3 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]
  - $\mathcal{P}$ = (ab) — 5 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]
  - $\mathcal{P}$ = (ab)c — 4 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]
  - $\mathcal{P}$ = (ab)(bd) — 0 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]
  - $\mathcal{P}$ = ab(cd) — 1 : [abcd^2,(abcd),(ab)(cd)^3,(ab)(bc)(cd)]

Sequence Mining

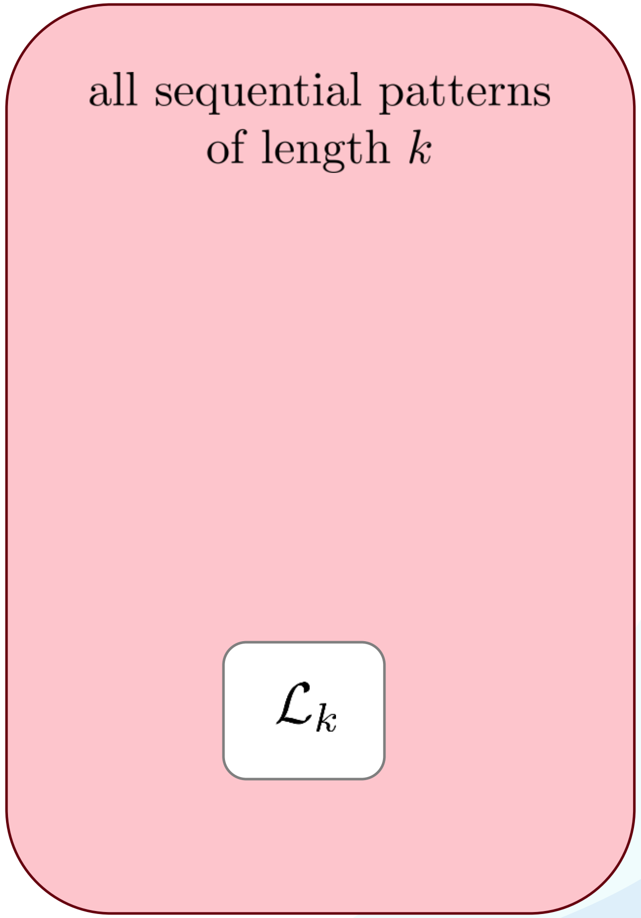
1. Temporal Data
2. Measuring Support
3. **Apriori-All Algorithm**
4. Extensions and Conclusion



Brute Force Approach

Goal: find all frequent sequential patterns

- Let $\boldsymbol{k}$ be the length of the longest sequence in $\mathcal{X}$ and $\boldsymbol{q}$ the size of the largest itemset
- Generate all sequential patterns of length $\leq \boldsymbol{k}$ with itemsets of size $\leq \boldsymbol{q}$ (this numer is finite)
- Compute the support of each candidate pattern
- Return all that have a support higher than $\boldsymbol{min\_sup}$
- Obviously, this is very expensive!



Smarter Approach Based on Apriori

- First described in Rakesh Agrawal, Ramakrishnan Srikant: Mining Sequential Patterns
- Similar to Apriori for frequent itemsets – avoid testing hopeless candidates
- If $\mathcal{A} \sqsubseteq \mathcal{B}$ ($\mathcal{A}$ is contained in $\mathcal{B}$), then $\mathcal{B}$ cannot be frequent if $\mathcal{A}$ is not frequent
  - $\text{support}\,(\mathcal{A}) \geq \text{support}\,(\mathcal{B})$ if $\mathcal{A} \sqsubseteq \mathcal{B}$
  - if $\mathcal{A} \sqsubseteq \mathcal{B}$ and $\text{support}\,(\mathcal{A}) < \mathbf{min\_sup}$ then $\text{support}\,(\mathcal{B}) < \mathbf{min\_sup}$

Step 1 – Determine All Litemsets

- $\mathcal{L} = \{\mathcal{A} \subseteq \mathcal{I} \mid \text{support}(\langle\mathcal{A}\rangle) \geq \mathbf{min\_sup}\}$ are all itemsets that appear in a sufficient number of sequences
- These itemsets are called **l**itemsets ($\mathcal{L}$ is the set of all **l**itemsets)

Step 1 – Determine All Litemsets

- $\mathcal{L} = \{\mathcal{A} \subseteq \mathcal{I} \mid \mathrm{support}(\langle \mathcal{A} \rangle) \geq \mathbf{min\_sup}\}$ are all itemsets that appear in a sufficient number of sequences
- These itemsets are called litemsets ($\mathcal{L}$ is the set of all litemsets)
- Consider $\mathcal{X}$ = [abcd,(abcd),(ab)(cd),(ab)(bc)(cd)] and $\mathbf{min\_sup} = 0.7$.

  The following itemsets are frequent:

  a (support = 4/4), b (support = 4/4), c (support = 4/4), d (support = 4/4), (ab) (support = 3/4), (cd) (support = 3/4)
- To determine all litemsets, we can use a variant of the original Apriori algorithm

  (the only difference is that support is now counted per sequence of transactions and not per transaction)

Step 2 – Transform the Dataset

- We only need to consider the **l**itemsets $\mathcal{L}$
  - There cannot be any frequent patterns that involve other itemsets
  - Frequent sequence patterns must be of the form $\mathcal{L}^*$!

- The set $\mathcal{L}_1 = \{\langle \mathcal{I} \rangle \mid \mathcal{I} \in \mathcal{L}\}$ is the set of all frequent sequence patterns of length 1
- $\mathcal{L}_k \subseteq \mathcal{L}^*$ is the set of all frequent sequence patterns of length exactly $\boldsymbol{k}$

  (to be 'grown' from shorter sequence patterns)

Step 2 – Transform the Dataset

- Transform $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ into $\mathcal{X}_T \in \mathbb{M}((\mathbb{P}(\mathcal{L}))^*)$

  → itemsets are mapped onto all **l**itemsets they contain

- Each sequence is now described by a sequence of sets of **l**itemsets (extra level)

Step 2 – Transform the Dataset

- Transform $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ into $\mathcal{X}_T \in \mathbb{M}((\mathbb{P}(\mathcal{L}))^*)$
  → itemsets are mapped onto all **l**itemsets they contain
- Each sequence is now described by a sequence of sets of **l**itemsets (extra level)

Example 1: Consider $\mathcal{L} = \{\{a\},\{b\},\{c\},\{a,b\}\}$

- $\langle\{a,c\},\{a,b,c\}\rangle$ corresponds to $\langle\{\{a\},\{c\}\},\{\{a\},\{b\},\{c\},\{a,b\}\}\rangle$
  - because $\{a,c\}$ has frequent subsets $\{a\},\{c\}$,
  - and $\{a,b,c\}$ has frequent subsets $\{a\},\{b\},\{c\}, \text{and } \{a,b\}$
- $\langle\{c\},\{a,c\}\rangle$ corresponds to $\langle\{\{c\}\},\{\{a\},\{c\}\}\rangle$

Step 2 – Transform the Dataset

- Transform $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ into $\mathcal{X}_T \in \mathbb{M}((\mathbb{P}(\mathcal{L}))^*)$

  → itemsets are mapped onto all **l**itemsets they contain

- Each sequence is now described by a sequence of sets of **l**itemsets (extra level)

Example 2: Consider $\mathcal{L} = \{\{a\}, \{b\}, \{c\}, \{a, b\}\}$ and $\mathcal{X} = [\langle\{a, c\}, \{a, b, c\}\rangle, \langle\{c\}, \{a, c\}\rangle, \ldots]$

- Then $\mathcal{X}_T = [\langle\{\{a\}\{c\}\}, \{\{a\}, \{b\}, \{c\}, \{a, b\}\}\rangle, \langle\{\{c\}\}, \{\{a\}, \{c\}\}\rangle, \ldots]$

Step 2 – Transform the Dataset

This preprocessing is not essential but makes sense because the dataset is traversed many times

- Transform $\mathcal{X} \in \mathbb{M}((\mathbb{P}(\mathcal{I}))^*)$ into $\mathcal{X}_T \in \mathbb{M}((\mathbb{P}(\mathcal{L}))^*)$

  → itemsets are mapped onto all **l**itemsets they contain

- Each sequence is now described by a sequence of sets of **l**itemsets (extra level)

Example 2: Consider $\mathcal{L} = \{\{a\},\{b\},\{c\},\{a,b\}\}$ and $\mathcal{X} = [\langle\{a,c\},\{a,b,c\}\rangle,\langle\{c\},\{a,c\}\rangle, \dots]$

- Then $\mathcal{X}_T = [\langle\{\{a\}\{c\}\},\{\{a\},\{b\},\{c\},\{a,b\}\}\rangle,\langle\{\{c\}\},\{\{a\},\{c\}\}\rangle, \dots]$

Testing whether a sequence pattern is supported by a sequence in the dataset is easy now!

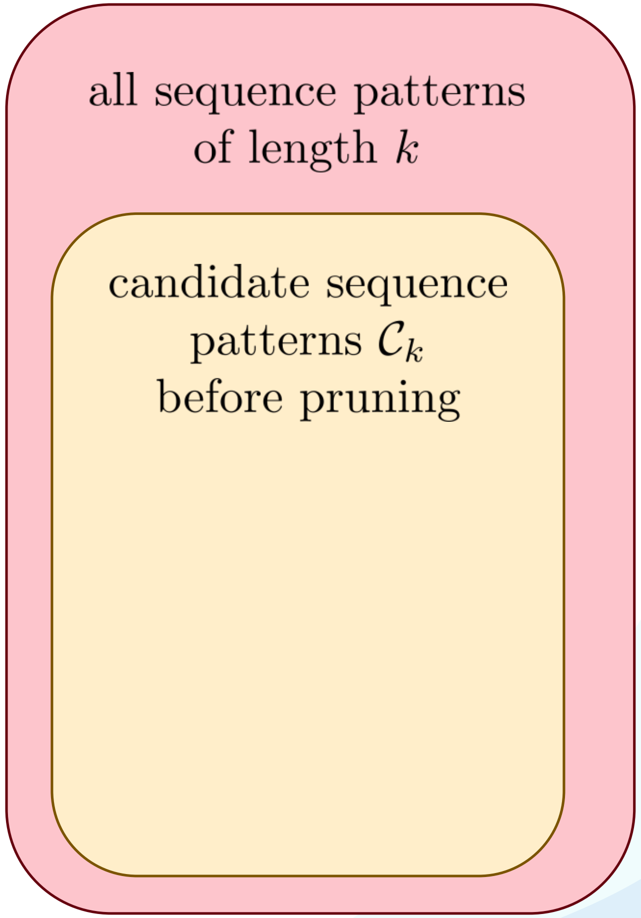
Step 3 – Generate a Set of Candidate Sequences

- Assume we have $\mathcal{L}_{k-1}$, the set of all frequent sequence patterns of length $\boldsymbol{k-1}$

  (recall that $\mathcal{L}_1 = \{\langle \mathcal{I} \rangle \mid \mathcal{I} \in \mathcal{L}\}$)

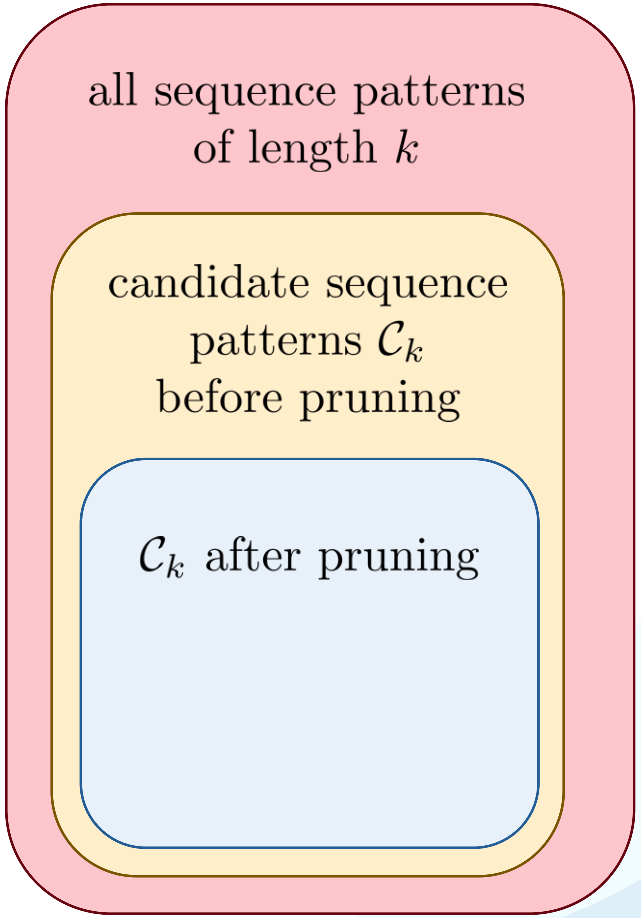
- Create the set of candidate sequences $\mathcal{C}_k$ by combining two sequences from $\mathcal{L}_{k-1}$ where the first $\boldsymbol{k-1}$ itemsets are the same

  (just like in Apriori for frequent itemsets)



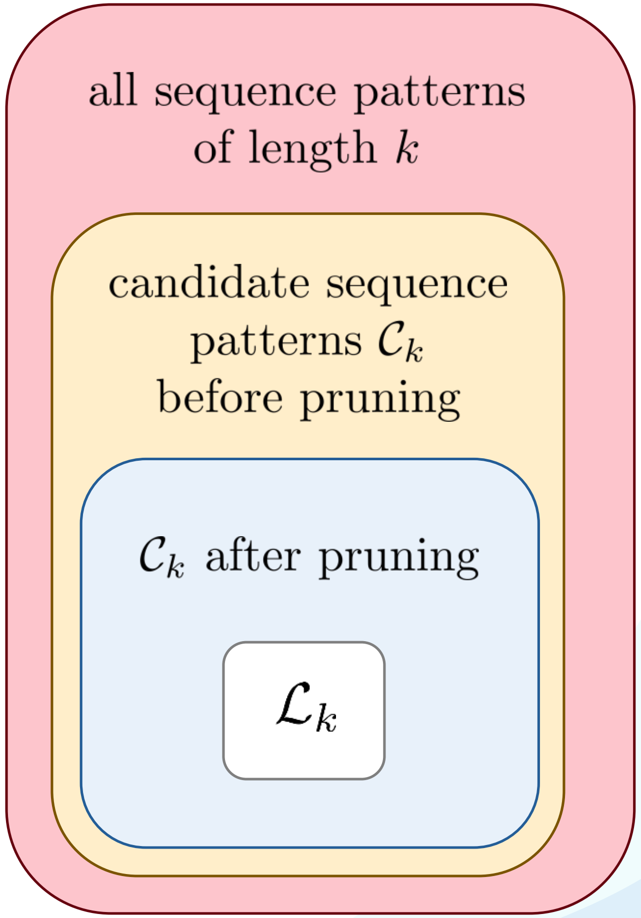
Step 4 – Prune the Set of Candidate Sequences

- For all candidate sequences $\mathcal{C} \in \mathcal{C}_k$
  - Consider all subsequences of $\mathcal{C}$ of length $\boldsymbol{k-1}$
  - If one of these subsequences is not in $\mathcal{L}_{k-1}$, then remove $\mathcal{C}$ from $\mathcal{C}_k$



Step 5 – Test All Candidate Sequences

- For each transformed sequence $\mathcal{S} \in \mathcal{X}_T$: Increment the count of $\mathcal{C} \in \mathcal{C}_k$ if $\mathcal{C}$ is contained in $\mathcal{S}$
- Remove all candidates $\mathcal{C} \in \mathcal{C}_k$ that do no meet the threshold to obtain $\mathcal{L}_k = \{\, \mathcal{C} \in \mathcal{C}_k \mid \mathrm{support}(\mathcal{C}) \geq \mathrm{min\_sup}\}$
- Increment $\boldsymbol{k}$ and go to Step 3 (Candidate Generation) until $\mathcal{L}_k = \emptyset$
- $\bigcup_k \mathcal{L}_k$ is the set of all frequent sequence patterns



Step 6 (Optional) – Remove Non-Maximal Patterns

- A sequence $\mathcal{S}$ is a maximal frequent sequence in $\mathcal{X}$
  - if $\mathcal{S}$ is frequent,
  - and there is no real supersequence $\mathcal{S}'$ that is also frequent ($\mathcal{S} \sqsubset \mathcal{S}'$)

Step 6 (Optional) – Remove Non-Maximal Patterns

- A sequence $\mathcal{S}$ is a maximal frequent sequence in $\mathcal{X}$
  - if $\mathcal{S}$ is frequent,
  - and there is no real supersequence $\mathcal{S}'$ that is also frequent ($\mathcal{S} \sqsubset \mathcal{S}'$)
- It is possible to keep only the maximal sequences
- However, support information for the subsequences will be lost (subsequences may have higher supports)


[1]

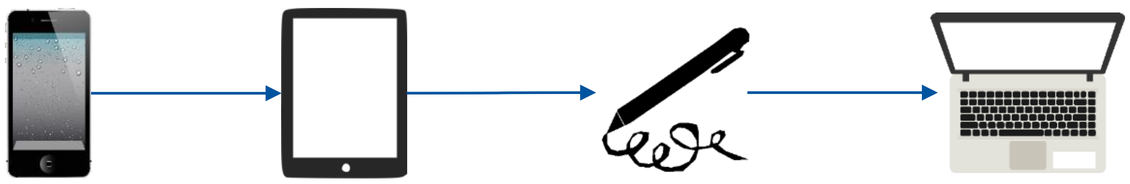
Other Sequential Pattern Mining Approaches

There are many other algorithms to find frequent sequential patterns:

- Maximal Frequent Sequences (MFS)
- Maximal Sequential Patterns using Sampling (MSPS)
- Indexed Bit Map (IBM)
- Sequential Pattern Mining with Length-decreasing Support (SLPMiner)
- WINEPI, MINEPI
- And many others…

Sequence Mining

1. Temporal Data
2. Measuring Support
3. Apriori-All Algorithm
4. **Extensions and Conclusion**



Association Rules Based on Frequent Sequences

- Frequent sequence patterns can be split in an ‘if’ and ‘then’ part (just like ‘normal’ association rules)
- $\langle\{\boldsymbol{beer}\},\{\boldsymbol{red},\boldsymbol{white}\}\rangle \Rightarrow \langle\{\boldsymbol{beer}\},\{\boldsymbol{red},\boldsymbol{white}\},\{\boldsymbol{wodka}\}\rangle$
- $\langle\{\boldsymbol{beer}\},\{\boldsymbol{beer}\}\rangle \Rightarrow \langle\{\boldsymbol{beer}\},\{\boldsymbol{beer}\},\{\boldsymbol{beer}\},\{\boldsymbol{beer}\},\{\boldsymbol{beer}\}\rangle$
- Many variants possible

How to Identify Interesting Frequent Sequences?

- For any technique that identifies patterns, it is important to filter out the less interesting ones
- One can look at things like correlations and base frequencies to decide how surprising sequences or sequence-based rules are (lift metric)
- It is also possible to add further constraints…

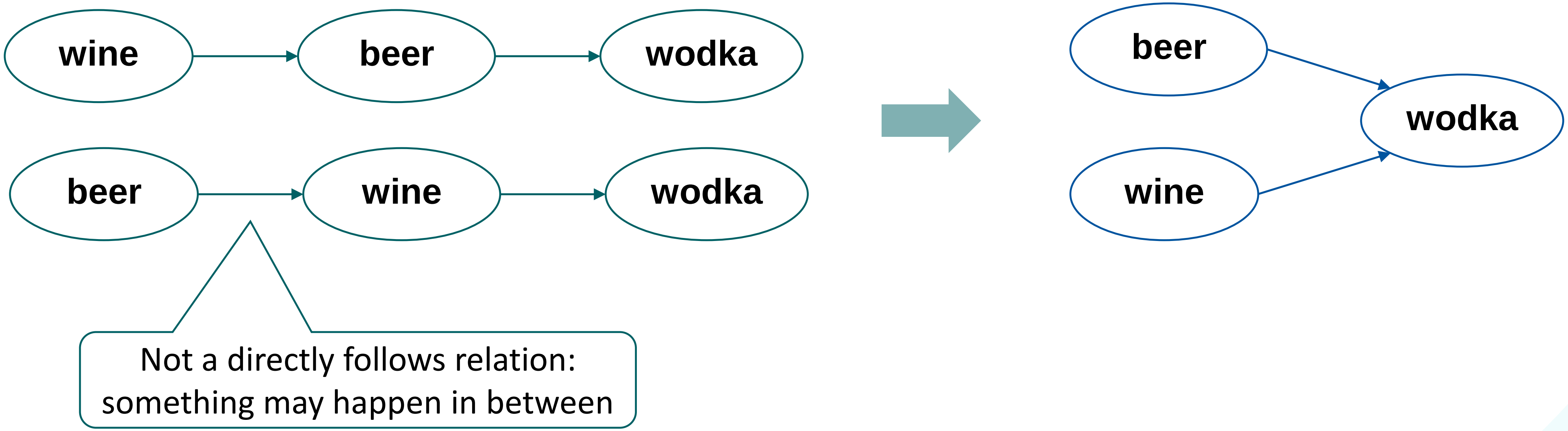
How to Identify Interesting Frequent Sequences?

Examples for further constraints:

- Item constraints: only consider sequences that include or exclude a set of items
- Length constraints: only consider patterns of a given size
- Time constraints: only consider patters that occur in a short timeframe
  (this includes gap and duration constraints)
- Regular expression constraints: only consider patterns that satisfy a regular expression or temporal constraint

Episode Mining

Extension: rather than looking for sequences we look for embedded partial orders (not subject of this course)



In future lectures – More Temporal Data!

Event data

Time-stamp	Case ID	Activity	f_1	f_2	…	f_D
t_1	3	a				
t_2	1	a				
t_3	1	b				
t_4	2	a				
t_5	3	b				
…	…	…				

Case ID is used to group events

Activity identifies the type of event

- We will see how to analyze time series
- Process mining: the analysis of event data as interplay between events and models