Association rules

Morten Berg Jensen

Department of Economics and Business Economics

April 25, 2024

Outline

Outcome

This lecture will help you to understand

- Advantages of market basket analysis and key concepts hereof
- Association analysis and association rules
- Algorithms (frequent itemset generation and rule generation) and interpretation from market basket analysis

Market basket transactions

- Many business enterprises accumulate large quantities of data from their day-to-day operations
- E.g. huge amounts of customer purchase data are collected daily at the checkout counters of grocery stores – such data is commonly known as **market basket transactions**
- Retailers analyze market basket transactions to learn about the purchasing behavior of their customers
- Such information can be used to support a variety of business-related tasks like marketing promotions, inventory management, and customer relationship management

Data sources

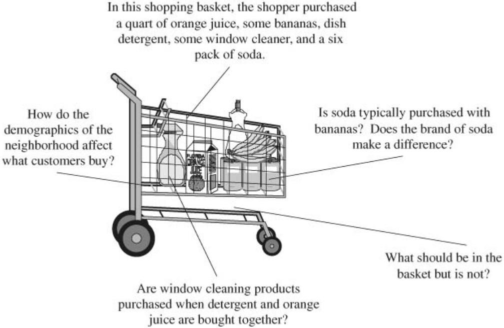
- The proliferation of this type of analysis is first and foremost driven by the increased availability of relevant data
- Data is typically obtained automatically – point-of-sales data from supermarkets, recordings of which movies/series Netflix customers see, recordings of which songs/artists Spotify customers listen to, ...
- In many situations, such as the two latter examples, we typically have additional information about the customer (demographics, seniority, ...)

Market basket analysis

- **Market basket analysis** focuses at purchase coincidence
- That is, whether two products are being purchased together, and whether the purchase of one product predicts the purchase of another
- This can, of course, be extended to more than two products
- Furthermore, this kind of analysis has been applied to an enlarged definition of the word product – services, census data, questionnaire data, Web data, medical records...

Market basket analysis (cont'd)

- In general, market basket analysis can be used to address question like these



source: Berry & Linoff, 2004: Data Mining Techniques for Marketing, Sales and Customer Relationship Management

Why market basket analysis

- Market basket analysis uses point-of-sale data (customers, orders/transactions, items/SKUs) to
  - **Identify and understand customers:** who are they and why do they make certain purchases – segmentation based on buying patterns
  - **Gain insight about products:** products purchased together, products which might benefit from promotion
  - **Take action:** pricing, cross-selling/cross-marketing, catalogue design, customized e-mails with add-on sales, store layout, stocking shelves
- Combining all of this with a customer loyalty card it becomes even more valuable

The fundamental assumption

- Joint occurrence of two (or more) products in most baskets imply that these products are complements in purchase and therefore a purchase of one will lead to a purchase of the other

Association analysis and association rules

- Association analysis can be useful for discovering interesting relationships hidden in large data sets
- The uncovered relationships can be represented in the form of association rules and/or sets of frequent items
- Association rules can be automatically generated from point-of-sale transaction data

	Milk, Bread	Milk -> Bread	Bread -> Milk
Examples	Milk, Tuna	Milk -> Tuna	Tuna -> Milk
	Vodka, Caviar	Vodka -> Caviar	Caviar -> Vodka

- Association rules represent patterns in the data without a real target variable
- They are a good example of undirected, exploratory data mining – in ML referred to as unsupervised learning

Two methodological themes

- There are two key issues that need to be addressed when applying association analysis to market basket data
  - First, some of the discovered patterns are potentially spurious because they may happen simply by chance
  - Second, discovering patterns from a large transaction data set can be computationally expensive
- These two themes guide the methods of association rule mining, and the remainder of this lecture
- We shall see that efficient algorithms have been developed along with recommendations for evaluating discovered patterns

Toy example

- Based on these transactions

TID	Items
1	{Bread, Milk}
2	{Bread, Diapers, Beer, Eggs}
3	{Milk, Diapers, Beer, Coke}
4	{Bread, Milk, Diapers, Beer}
5	{Bread, Milk, Diapers, Coke}

  the following association rule may be extracted

$$\{\text{Diapers}\} \rightarrow \{\text{Beer}\}$$

- {Beer} is referred to as a **consequent** whereas {Diapers} is an **antecedent**

Toy example (cont'd)

- Market basket data for association analysis are represented in a binary format

TID	Bread	Milk	Diapers	Beer	Eggs	Coke
1	1	1	0	0	0	0
2	1	0	1	1	1	0
3	0	1	1	1	0	1
4	1	1	1	1	0	0
5	1	1	1	0	0	1

- Each row corresponds to a transaction and each column corresponds to an item
- Obviously, this ignores important aspects of the data such as the quantity of items sold or the price paid to purchase them

Itemset

- We let $\mathcal{I} = \{i_1, i_2, \ldots, i_d\}$ be the set of all items in our market basket data
- The set of all transactions is denoted $\mathcal{T} = \{t_1, t_2, \ldots, t_N\}$ where each transaction, t_j, contains a subset of items from $\mathcal{I}$
- A collection of zero or more items, X, is termed an **itemset**
- If an itemset has k items it is called a $k-$itemset – the itemset associated with TID $= 1$ is a 2-itemset
- The transaction width is defined as the number of items in a transaction
- A transaction, t_j is said to contain an itemset X if X is a subset of t_j – the transaction t_2 contains {Bread, Diapers} but not {Bread, Milk}

Support count and support

- For an itemset we define its **support count**, which refers to the number of transactions that contains the itemset

$$\sigma(X) = |\{t_i | X \subseteq t_i, t_i \in \mathcal{T}\}|$$

  with $|\cdot|$ denoting the number of elements
- For the itemset {Milk, Diapers, Beer} the support count equals two
- For the itemset {Milk, Diapers} the support count equals three
- The **support** of an itemset is the fraction of transactions that contains the itemset

$$s(X) = \sigma(X)/N$$

Association rule

- For disjoint itemsets X and Y an **association rule** is an implication expression of the form $X \rightarrow Y$
- Notice that implication means co-occurrence, not causality!
- We are interested in finding association rules that will predict the occurrence of an item based on the occurrence of other items in a transaction

Support (of an association rule)

- The strength of an association rule can be measured in terms of its **support**, s

$$s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N} = s(X \cup Y)$$

- For the association rule $\{\text{Milk, Diapers}\} \rightarrow \{\text{Beer}\}$ the support equals $2/5 = 0.4$
- It is an estimate of the probability of observing both item sets in a randomly selected transaction, $\mathbf{P}(X \cup Y)$
- An association rule with very low support may occur by chance
- A rule with low support may not be of interest from a business perspective as it involves items that are rarely bought together
- As a consequence, support is often used to eliminate uninteresting rules via a minimum support threshold

Confidence

- The strength of an association rule can also be measured via its **confidence**, c

$$c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} = \frac{s(X \rightarrow Y)}{s(X)} = \frac{s(X \cup Y)}{s(X)}$$

- For the association rule $\{\text{Milk, Diapers}\} \rightarrow \{\text{Beer}\}$ the confidence equals $2/3 = 0.67$
- It is an estimate of the probability of Y conditional of X, $\mathbf{P}(Y|X)$
- Confidence thus measures the reliability of the inference made by a rule – the higher the confidence the more likely it is for Y to be present in transactions that contain X

Confidence (cont'd)

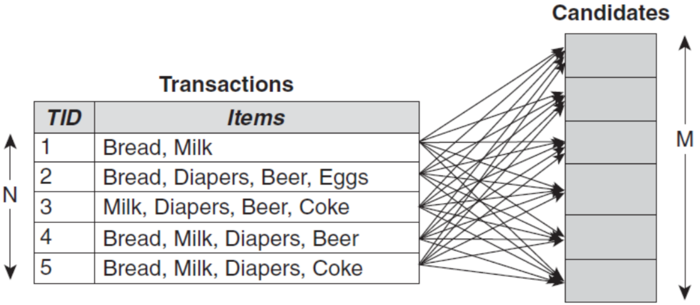
- There are some drawbacks associated with the confidence measure

	Coffee	$\overline{\text{Coffee}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	65	15	80
	80	20	100

- Association rule is Tea $\rightarrow$ Coffee
- Confidence $= \mathbf{P}(\text{Coffee}|\text{Tea}) = 0.15/0.20 = 0.75$ – high confidence
- But $\mathbf{P}(\text{Coffee}) = 0.80$
- And $\mathbf{P}(\text{Coffee}|\overline{\text{Tea}}) = 0.65/0.80 = 0.8125$ – higher confidence

Problem formulation and algorithm

- The problem that we face can now be expressed explicitly: *Given the set of transactions, $\mathcal{T}$ find all rules having support $\geq$ minsup and confidence $\geq$ minconf where minsup and minconf are thresholds determined by the investigator*



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Problem formulation and algorithm (cont'd)

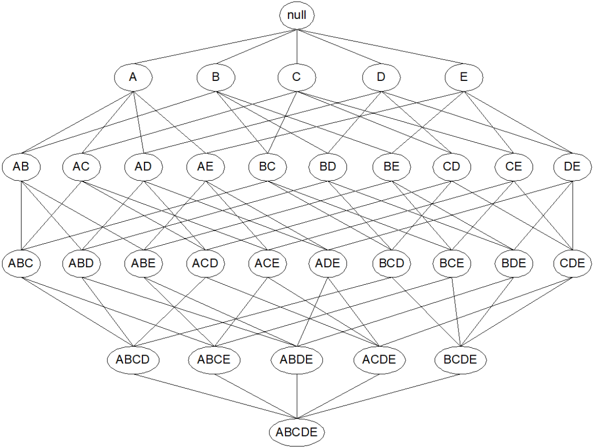
- From Agrawal et al. 1993
  - Find all rules that have "Diet Coke" as consequent. These rules may help plan what the store should do to boost the sale of Diet Coke. LHS -> Diet coke
  - Find all rules that have "bagels" in the antecedent. These rules may help determine what products may be impacted if the store discontinues selling bagels. Bagels -> RHS
  - Find all rules that have "sausage" in the antecedent and "mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in order to make it highly likely that mustard will also be sold. Sausage, ... -> Mustard
  - Find all the rules relating items located on shelves A and B in the store. These rules may help shelf planning by determining if the sale of items on shelf A is related to the sale of items on shelf B.

Problem formulation and algorithm (cont'd)

- A naive approach to this problem is brute-force – calculate all possible rules and their associated support and confidence
- The sheer number of rules renders this approach computationally infeasible
- The solution is to decompose the problem into two subtasks
  1. Frequent itemset generation – find the itemsets that satisfy the *minsup* threshold
  2. Rule generation – extract all high-confidence rules from 1.

Frequent itemset generation

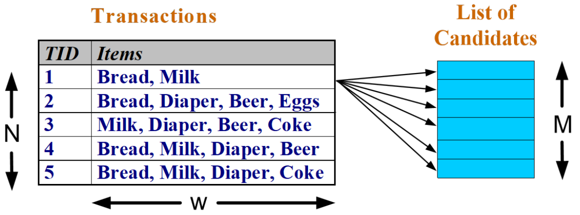
- A lattice can be used to enumerate the list of all possible item sets, M, which equals 2^k for k items – i.e. M grows exponentially



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Frequent itemset generation (cont'd)

- The brute-force approach determines the support count for all candidate itemsets in the lattice



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

- Complexity $\sim$ O(NMw) which is extremely expensive - w is the maximum transaction width

Frequent itemset generation (cont'd)

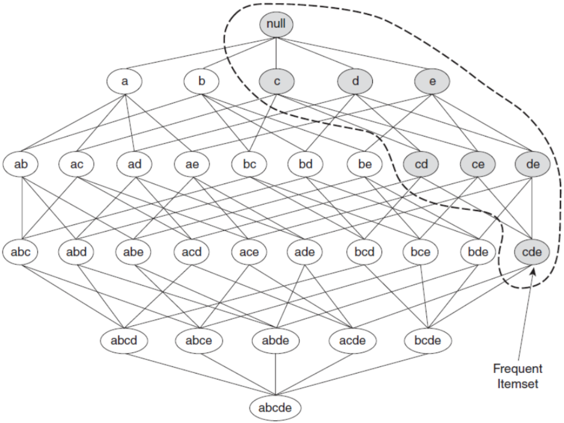
- I.e. ask how many rows have a 1 in column Beer, how many rows have 1's in columns Beer and Bread, how many rows have 1's in columns Beer, Bread, and Milk ... and do this for all M combinations (or at least them involving at most w items)
- A reduction can be accomplished if we either reduce the number of candidate itemsets, M, reduce the number of comparisons, or reduce the number of transactions, N

Frequent itemset generation (cont'd)

- The **Apriori** principle utilizes the support measure to reduce the number of candidate item sets
- This is done by noticing that if an itemset is frequent then all of its subsets must also be frequent
- Conversely, if an itemset is infrequent then all of its supersets must also be infrequent
- Trimming the exponentially growing search space based on the support measure is called support-based pruning

Frequent itemset generation (cont'd)

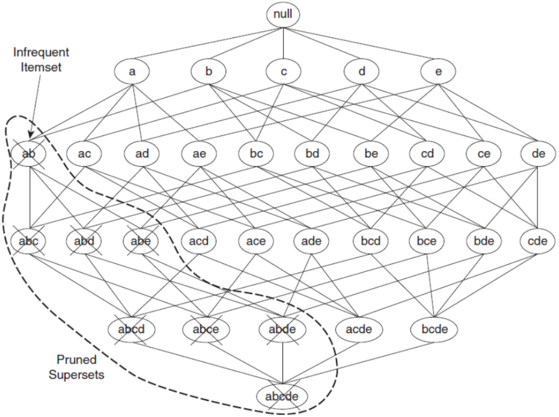
- Frequent subsets due to the apriori principle



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Frequent itemset generation (cont'd)

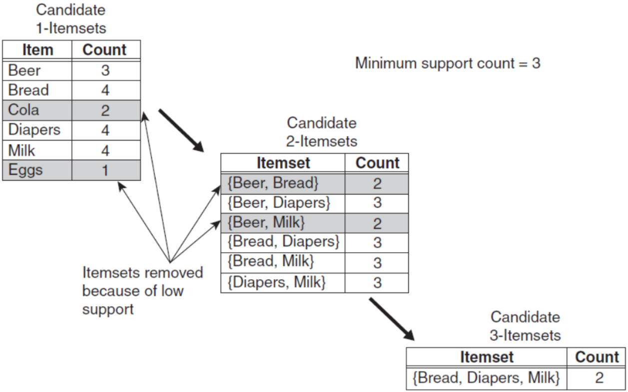
- Pruned supersets due to the apriori principle



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Frequent itemset generation (cont'd)

- Using the apriori algorithm to generate frequent itemsets



Candidate 1-Itemsets

Item	Count
Beer	3
Bread	4
Cola	2
Diapers	4
Milk	4
Eggs	1

Candidate 2-Itemsets

Itemset	Count
{Beer, Bread}	2
{Beer, Diapers}	3
{Beer, Milk}	2
{Bread, Diapers}	3
{Bread, Milk}	3
{Diapers, Milk}	3

Candidate 3-Itemsets

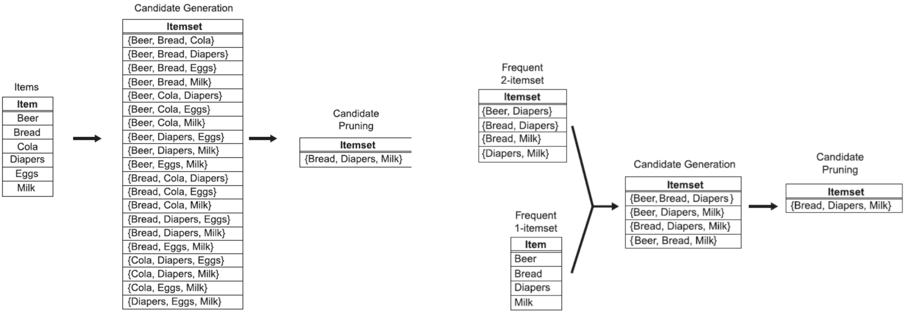
Itemset	Count
{Bread, Diapers, Milk}	2

source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Candidate generation and pruning

- Candidate generation
  - Brute-force – generate all possible k-itemsets
  - $F_{k-1} \times F_1$ – extend each frequent (k-1)-itemset with a frequent itemset that is not part of the (k-1) itemset
  - $F_{k-1} \times F_1 +$ lexicographic – extend each frequent (k-1)-itemset with a frequent itemset that is lexigographically larger than the elements of the (k-1) itemset
  - $F_{k-1} \times F_{k-1}$ – extend each frequent (k-1)-itemset with another frequent (k-1)-itemset if their first k-2 items are identical
- Pruning
  - To prune a candidate k-itemset, X, look at X-$\{i_j\}$, $\forall j = 1, \ldots, k$
  - If any of them are infrequent, then X is pruned

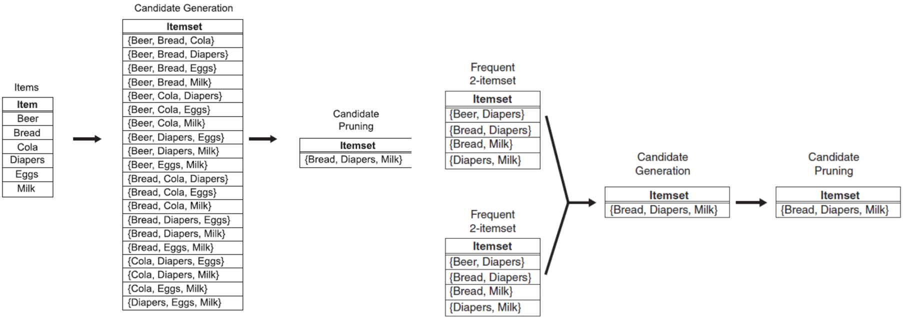
Comparing brute-force with $F_{k-1} \times F_1$ candidate generation



Brute-force Apriori

source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Comparing brute-force with $F_{k-1} \times F_{k-1}$ candidate generation



Brute-force

Apriori

source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Setting an appropriate support threshold

- If the minimum support threshold is set too high, one could miss itemsets involving interesting but rarely purchased items
  - Newly launched products
  - Highly priced products
  - Products with long replacement cycles
- If the minimum support threshold is set too low, market basket analysis becomes computationally expensive and the number of itemsets will be very large

Rule generation

- From a frequent itemset, Y, an association rule may be extracted by partitioning Y into X and $Y - X$ such that $X \to Y - X$ satisfies the confidence threshold
- Notice that as Y is frequent so is $Y - X$

Rule generation (cont'd)

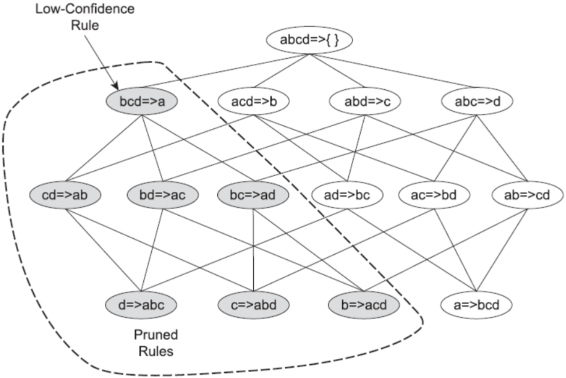
- The number of possible association rules R in an itemset grows exponentially with the size d of the itemset

$$R = \sum_{k=1}^{d} \binom{d}{k} \sum_{i=1}^{d-k} \binom{d-k}{i} = 3^d - 2^{d+1} + 1$$

d	R
1	0
2	2
3	12
4	50
5	180
6	602
7	1932
8	6050

Rule generation (cont'd)

- An idea similar to support-based pruning for itemsets can be established for association rules



source: Tan, Steinbach, Karpatne and Kumar, 2020: Introduction to Data Mining

Assessment of association rules

- The **lift** for the association rule $X \to Y$ is defined as

$$\text{lift}(X \to Y) = \frac{c(X \to Y)}{s(Y)} = \frac{s(X \cup Y)}{s(X)s(Y)}$$

- If this ratio is larger than 1 we have an upward lift – knowing that X has happened increases the probability that Y occurs
- Lift is the factor by which prediction improves when we apply the rule, compared to what we would be able to predict if we did not apply the rule

Assessment of association rules (cont'd)

- Calculating the lift

	Coffee	$\overline{\text{Coffee}}$	
Tea	15	5	20
$\overline{\text{Tea}}$	65	15	80
	80	20	100

- $\mathbf{P}(\text{Coffee}|\text{Tea})/\mathbf{P}(\text{Coffee}) = (0.15/0.20)/0.8 = 0.9375$
- $\mathbf{P}(\text{Coffee}|\overline{\text{Tea}})/\mathbf{P}(\text{Coffee}) = (0.65/0.80)/0.8 = 1.0156$

Example – Groceries

- From Chapman and Feit 2015
- In this example we will investigate the possibility of recommending grocery items to customers
- We have information from 9,835 transactions comprising 169 unique items
- Approximately half of the transactions involve one, two, or three items, the largest transaction involves 32 items
- 'The most frequently bought item is "whole milk" followed by "other vegetables"
- The data is provided as a "transactions" class
- We extract association rules with support a above 0.01 and with a confidence above 0.3 – this will result in a modest number of rules and involve a suitable number of items

Example – Groceries (cont'd)

- We see that the rules found involve 88 items (out of the 169)
- A total of 125 rules were found
- If we filter by requiring that the rules should have a lift above 3 we see that for rule 1
  - If a transaction contains {beef} then it is also relatively more likely to contain {root vegetables}
  - The combination appears in 1.7 % of the transactions – support = 0.017
  - The combination is more than 3 times more likely to occur together than would be expected from the individual rates of incidence
  - The unconditional probability for {root vegetables} equals 0.109 whereas the conditional probability equals 0.331
- A store might exploit this by creating a display for root vegetables near the beef counter or put a coupon for beef in the root vegetable area