ELEN 4720: Machine Learning for Signals, Information and Data

Lecture 23

Prof. John Paisley

Department of Electrical Engineering

Columbia University

ASSOCIATION ANALYSIS

SETUP

Many businesses have massive amounts of customer purchase/behavior data.

- Amazon has your order and viewing history
- A grocery store knows objects purchased in each transaction

Using this data, we may want to find sub-groups of products that co-occur

- Retailers can use this to cross-promote products through “deals”
- Grocery stores can use this to strategically place items
- Online retailers can use this to recommend content

We’ll discuss a method for doing this based on strategic counting. We’ll also see how this generalizes to other problems like polling analysis.

Market basket analysis

Association analysis mines these patterns through *empirical* probabilities.

Classic example: Consider the following "market baskets" of five customers.

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

Using such data, we want to analyze patterns of co-occurance within it. We can use these patterns to define *association rules*. For example,

$$\{\text{diapers}\} \;\Rightarrow\; \{\text{beer}\}$$

ASSOCIATION ANALYSIS AND RULES

Imagine we have:

- M different unique objects indexed by $\{1, \dots, M\}$
- A collection of subsets $X_n \subset \{1, \dots, M\}$. Think of X_n as the index of items purchased by customer $n = 1, \dots, N$. This is the data.

Association analysis: Find subsets of objects that often appear together. For example, if $\mathcal{K} \subset \{1, \dots, M\}$ indexes objects that frequently co-occur, then

$$P(\mathcal{K}) = \frac{\#\{n \text{ such that } \mathcal{K} \subseteq X_n\}}{N} \quad \text{is large relatively speaking}$$

Example: $\mathcal{K} = \{$`peanut_butter, jelly, bread`$\}$

Association rules: Learn correlations. Let A and B be disjoint sets. Then $A \Rightarrow B$ means purchasing A increases likelihood of also purchasing B.

Example: $\{$`peanut_butter, jelly`$\} \Rightarrow \{$`bread`$\}$

PROCESSING THE BASKET

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

Figure: An example of 5 baskets.

TID	Bread	Milk	Diapers	Beer	Eggs	Cola
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

Figure: A binary representation of these 5 baskets for analysis.

PROCESSING THE BASKET

TID	Bread	Milk	Diapers	Beer	Eggs	Cola
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

We want to find subsets that occur with probability above some threshold.

For example, does {`bread, milk`} occur relatively frequently?

- Go to each of the 5 baskets and count the number that contain both.
- Divide this number by 5 to get the frequency.
- Aside: Notice that the basket might have more items in it.

When $N = 5$ and $M = 6$ as in this case, we can easily check every possible combination. However, real problems might have $N \approx 10^8$ and $M \approx 10^4$.

Some combinatorics

Some combinatorial analysis will show that brute-force search isn't possible.

Q: How many different subsets $\mathcal{K} \subseteq \{1, \ldots, M\}$ are there?

A: Each subset can be represented by a binary indicator vector of length M. The total number of possible vectors is 2^M.

Q: Nobody will have a basket with every item in it, so we shouldn't check every combination. How about if we only check up to k items?

A: The number of sets of size k picked from M items is $\binom{M}{k} = \frac{M!}{k!(M-k)!}$. For example, if $M = 10^4$ and $k = 5$, then $\binom{M}{k} \approx 10^{18}$.

Takeaway: Though the problem only requires counting, we need an algorithm that can tell us which $\mathcal{K}$ we should count and which we can ignore.

Quantities of interest

Before we find an efficient counting algorithm, what do we want to learn?

- Again, let $\mathcal{K} \subset \{1, \ldots, M\}$ and $A, B \subset \mathcal{K}$, where $A \cup B = \mathcal{K}$, $A \cap B = \emptyset$.

We're interested in the following empirically-calculated probabilities:

1. $P(\mathcal{K}) = P(A, B)$: The *prevalence* (or support) of items in set $\mathcal{K}$. We want to find which combinations co-occur often.
2. $P(B|A) = \frac{P(\mathcal{K})}{P(A)}$: The *confidence* that B appears in the basket given A is in the basket. We use this to define a *rule* $A \Rightarrow B$.
3. $L(A, B) = \frac{P(A,B)}{P(A)P(B)} = \frac{P(B|A)}{P(B)}$: The *lift* of the rule $A \Rightarrow B$. This is a measure of how much *more* confident we are in B given that we see A.

Example

For example, let

$$\mathcal{K} = \{\texttt{peanut\_butter, jelly, bread}\},$$

$$A = \{\texttt{peanut\_butter, jelly}\}, B = \{\texttt{bread}\}$$

- A *prevalence* of 0.03 means that `peanut_butter, jelly` and `bread` appeared together in 3% of baskets.
- A *confidence* of 0.82 means that when both `peanut_butter` and `jelly` were purchased, 82% of the time `bread` was also purchased.
- A *lift* of 1.95 means that it's 1.95 more probable that `bread` will be purchased given that `peanut_butter` and `jelly` were purchased.

Apriori algorithm

The goal of the **Apriori algorithm** is to quickly find all of the subsets $\mathcal{K} \subset \{1, \ldots, M\}$ that have probability greater than a predefined threshold t.

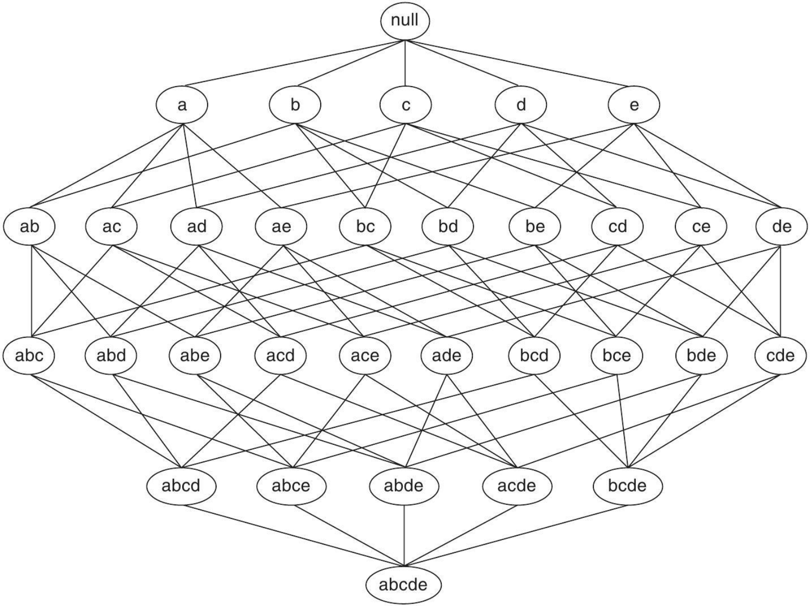
- Such a $\mathcal{K}$ will contain items that appear in at least $N \cdot t$ of the N baskets.
- A small fraction of such $\mathcal{K}$ should exist out of the 2^M possibilities.

Apriori uses properties about $P(\mathcal{K})$ to reduce the number of subsets that need to be checked to a small fraction of all 2^M sets.

- It starts with $\mathcal{K}$ containing 1 item. It then moves to 2 items, etc.
- Sets of size $k-1$ that "survive" help determine sets of size k to check.
- Important: Apriori finds *every* set $\mathcal{K}$ such that $P(\mathcal{K}) > t$.

Next slide: The structure of the problem can be organized in a lattice.

LATTICE REPRESENTATION



Frequency dependence

We can use two properties to develop an algorithm for efficiently counting.

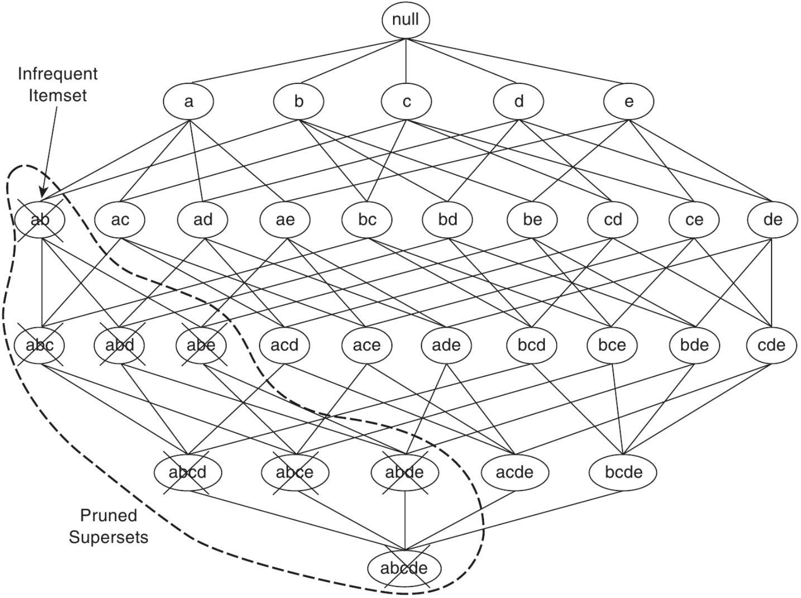
1. If the set $\mathcal{K}$ is not big enough, then $\mathcal{K}' = \mathcal{K} \cup A$ with $A \subset \{1, \ldots, M\}$ is not big enough. In other words: $P(\mathcal{K}) < t$ implies $P(\mathcal{K}') < t$

   e.g., Let $\mathcal{K} = \{a, b\}$. If these items appear together in x baskets, then the set of items $\mathcal{K}' = \{a, b, c\}$ appears in $\leq x$ baskets since $\mathcal{K} \subset \mathcal{K}'$.

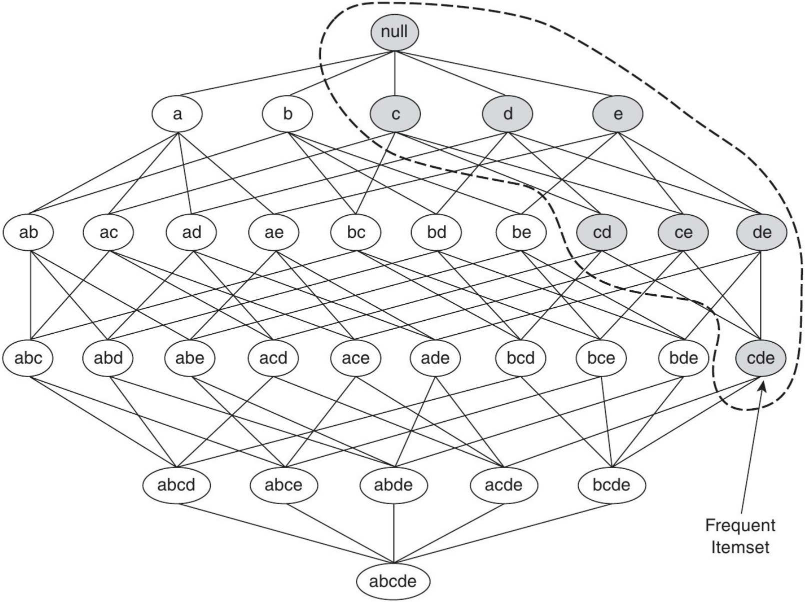
   Mathematically: $P(\mathcal{K}') = P(\mathcal{K} \cup A) = P(A|\mathcal{K})P(\mathcal{K}) \leq P(\mathcal{K}) < t$

2. By the converse, if $P(\mathcal{K}) > t$ and $A \subset \mathcal{K}$, then $P(A) > P(\mathcal{K}) > t$.

FREQUENCY DEPENDENCE: PROPERTY 1



FREQUENCY DEPENDENCE: PROPERTY 2



APRIORI ALGORITHM (ONE VERSION)

Here is a basic version of the algorithm. It can be improved in clever ways.

Apriori algorithm

Set a threshold $N \cdot t$, where $0 < t < 1$ (but relatively small).

1. $|\mathcal{K}| = 1$: Check each object and keep those that appear in $\geq N \cdot t$ baskets.
2. $|\mathcal{K}| = 2$: Check all pairs of objects that survived Step 1 and keep the sets that appear in $\geq N \cdot t$ baskets.

$\vdots$

k. $|\mathcal{K}| = k$: Using all sets of size $k - 1$ that appear in $\geq N \cdot t$ baskets,
   - Increment each set with an object surviving Step 1 not already in the set.
   - Keep all sets that appear in $\geq N \cdot t$ baskets

As k increases, we hope that the number of sets that survive decreases. At a certain $k < M$, no sets will survive and we're done.

ROUGH PROOF (OPTIONAL)

We can show that this algorithm returns *every* set $\mathcal{K}$ for which $P(\mathcal{K}) > t$.

- Imagine we know every set of size $k-1$ for which $P(\mathcal{K}) > t$.
- Then every potential set of size k that could possibly have $P(\mathcal{K}) > t$ will also be checked.

e.g. Let $k = 3$: The set $\{a, b, c\}$ appears in $> N \cdot t$ baskets. Will we check it?
Known: $\{a, b\}$ and $\{c\}$ must appear in $> N \cdot t$ baskets.
Assumption: We've found $\mathcal{K} = \{a, b\}$ as a set satisfying $P(\mathcal{K}) > t$.
Apriori algorithm: We know $P(\{c\}) > t$ and so will check $\{a, b\} \cup \{c\}$.
Induction: We have all $|\mathcal{K}| = 1$ by brute-force search (start induction).

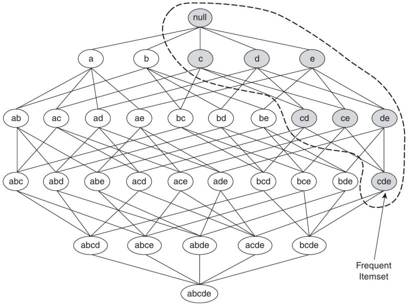
FINDING ASSOCIATION RULES

We've found all $\mathcal{K}$ such that

$$P(\mathcal{K}) > t.$$

Now we want to find association rules.

These are of the form $P(A|B) > t_2$ where we split $\mathcal{K}$ into subsets A and B.



Notice:

1. $P(A|B) = \frac{P(\mathcal{K})}{P(B)}$.
2. If $P(\mathcal{K}) > t$ and A and B partition $\mathcal{K}$, then $P(A) > t$ and $P(B) > t$.
3. Since Apriori found all $\mathcal{K}$ such that $P(\mathcal{K}) > t$, it found $P(A)$ and $P(B)$, so we can calculate $P(A|B)$ without counting again.

Example 1: Polling data

Feature	Demographic	# Values	Type
1	Sex	2	Categorical
2	Marital status	5	Categorical
3	Age	7	Ordinal
4	Education	6	Ordinal
5	Occupation	9	Categorical
6	Income	9	Ordinal
7	Years in Bay Area	5	Ordinal
8	Dual incomes	3	Categorical
9	Number in household	9	Ordinal
10	Number of children	9	Ordinal
11	Householder status	3	Categorical
12	Type of home	5	Categorical
13	Ethnic classification	8	Categorical
14	Language in home	3	Categorical

Data

$N = 6876$ questionnaires

14 questions coded into $M = 50$ items

For example:

- ordinal (2 items): Pick the item based on value being $\lessgtr$ median
- categorical: item = category

  x categories $\rightarrow$ x items

- Based on the item encoding, it's clear that no "basket" can have every item.
- We see that association analysis extends to more than consumer analysis.

EXAMPLE 1: POLLING DATA

Association rule 1: Support 13.4%, confidence 80.8%, and lift 2.13.

$$\begin{bmatrix} \text{language in home} & = & \textit{English} \\ \text{householder status} & = & \textit{own} \\ \text{occupation} & = & \{\textit{professional/managerial}\} \end{bmatrix}$$

$$\Downarrow$$

$$\text{income} \geq \textit{\$40,000}$$

Association rule 2: Support 26.5%, confidence 82.8% and lift 2.15.

$$\begin{bmatrix} \text{language in home} & = & \textit{English} \\ \text{income} & < & \textit{\$40,000} \\ \text{marital status} & = & \textit{not married} \\ \text{number of children} & = & \textit{0} \end{bmatrix}$$

$$\Downarrow$$

$$\text{education} \notin \{\textit{college graduate, graduate study}\}$$

EXAMPLE 2: VOTER POLLING IN 1984

1. Republican
2. Democrat
3. handicapped-infants = yes
4. handicapped-infants = no
5. water project cost sharing = yes
6. water project cost sharing = no
7. budget-resolution = yes
8. budget-resolution = no
9. physician fee freeze = yes
10. physician fee freeze = no
11. aid to El Salvador = yes
12. aid to El Salvador = no
13. religious groups in schools = yes
14. religious groups in schools = no
15. anti-satellite test ban = yes
16. anti-satellite test ban = no
17. aid to Nicaragua = yes
18. aid to Nicaragua = no
19. MX-missile = yes
20. MX-missile = no
21. immigration = yes
22. immigration = no
23. synfuel corporation cutback = yes
24. synfuel corporation cutback = no
25. education spending = yes
26. education spending = no
27. right-to-sue = yes
28. right-to-sue = no
29. crime = yes
30. crime = no
31. duty-free-exports = yes
32. duty-free-exports = no
33. export administration act = yes
34. export administration act = no

Association Rule	Confidence
{budget resolution = no, MX-missile=no, aid to El Salvador = yes } $\longrightarrow$ {Republican}	91.0%
{budget resolution = yes, MX-missile=yes, aid to El Salvador = no } $\longrightarrow$ {Democrat}	97.5%
{crime = yes, right-to-sue = yes, physician fee freeze = yes} $\longrightarrow$ {Republican}	93.5%
{crime = no, right-to-sue = no, physician fee freeze = no} $\longrightarrow$ {Democrat}	100%