

# COMS 4721: Machine Learning for Data Science

## Lecture 23, 4/20/2017

Prof. John Paisley

Department of Electrical Engineering  
& Data Science Institute  
Columbia University

# ASSOCIATION ANALYSIS

# SETUP

Many businesses have massive amounts of customer behavior data.

- ▶ Amazon has your order and viewing history
- ▶ A grocery store knows objects purchased in each transaction
- ▶ Other retailers have similar data

Using this data, we may want to find sub-groups of products that tend to co-occur in purchasing or viewing behavior.

- ▶ 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
- ▶ This also applies to other problems, such as polling analysis

We'll discuss a method for doing this based on strategic counting.

# MARKET BASKET ANALYSIS

*Association analysis* mines these patterns through *empirical* probabilities.

For example consider the following “market baskets” of five customers.

<i>TID</i>	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-occurrence 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:

- ▶  $p$  different objects indexed by  $\{1, \dots, p\}$
- ▶ A collection of subsets of these objects  $X_n \subset \{1, \dots, p\}$ . Think of  $X_n$  as the index of items purchased by customer  $n = 1, \dots, N$ .

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

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

*Example:*  $\mathcal{K} = \{\text{peanut\_butter}, \text{ jelly}, \text{ bread}\}$

**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:*  $\{\text{peanut\_butter}, \text{ jelly}\} \Rightarrow \{\text{bread}\}$

# PROCESSING THE BASKET

<i>TID</i>	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  $p = 6$  as in this case, we can easily check every possible combination. However, real problems might have  $N \approx 10^8$  and  $p \approx 10^4$ .

# SOME COMBINATORICS

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

**Q:** How many different subsets  $\mathcal{K} \subseteq \{1, \dots, p\}$  are there?

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

**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  $p$  items is  $\binom{p}{k} = \frac{p!}{k!(p-k)!}$ . For example, if  $p = 10^4$  and  $k = 5$ , then  $\binom{p}{k} \approx 10^{18}$ .

**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 count?

- ▶ Again, let  $\mathcal{K} \subset \{1, \dots, p\}$  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

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

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

- ▶ 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  $\mathcal{K} \subset \{1, \dots, p\}$  that have probability greater than a predefined threshold  $t$ .

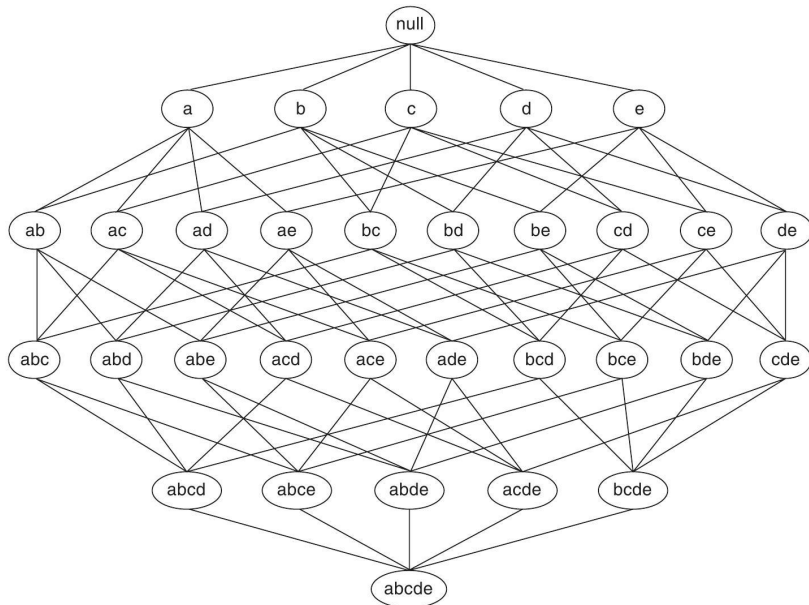
- ▶ Such a  $\mathcal{K}$  will contain items that appear in at least  $N \cdot t$  of the  $N$  baskets.
- ▶ A small fraction of such  $\mathcal{K}$  should exist out of the  $2^p$  possibilities.

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

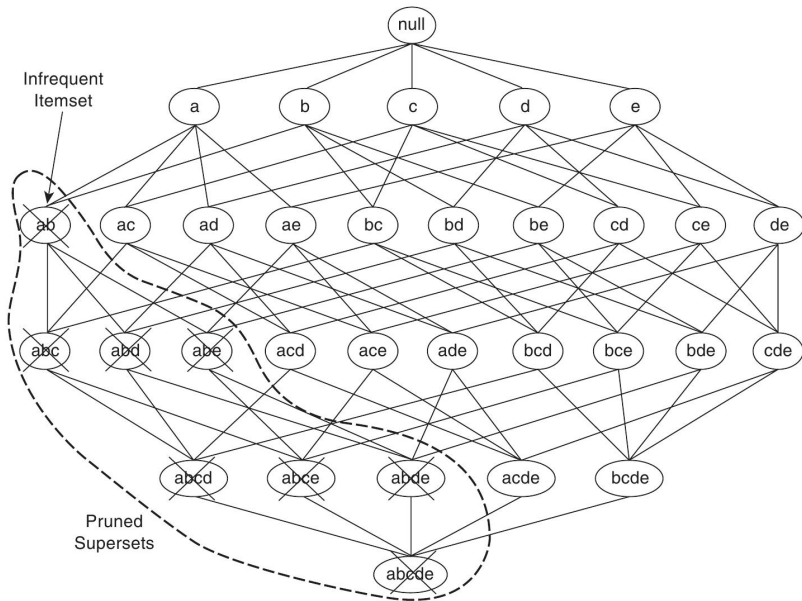
1. If the set  $\mathcal{K}$  is not big enough, then  $\mathcal{K}' = \mathcal{K} \cup A$  with  $A \subset \{1, \dots, p\}$  is not big enough. In other words:  $P(\mathcal{K}) < t$  implies  $P(\mathcal{K}') < t$

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  $\mathcal{K} \subset \mathcal{K}'$ .

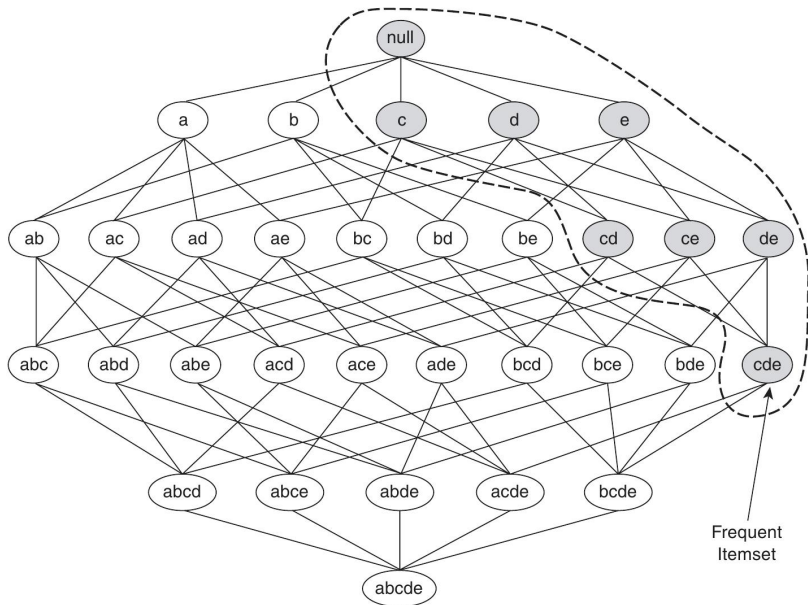
Mathematically:  $P(\mathcal{K}') = P(\mathcal{K} \cup A) = P(A|\mathcal{K})P(\mathcal{K}) \leq P(\mathcal{K}) < t$

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
- 

It should be clear that as  $k$  increases, we can hope that the number of sets that survive decrease. At a certain  $k < p$ , no sets will survive and we're done.



# MORE CONSIDERATIONS

1. 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 have  $P(\mathcal{K}) > t$  will 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).
2. As written, this can lead to duplicate sets for checking, e.g.,  $\{a, b\} \cup \{c\}$  and  $\{a, c\} \cup \{b\}$ . Indexing methods can ensure we create  $\{a, b, c\}$  once.
3. For each proposed  $\mathcal{K}$ , should we iterate through each basket for checking? There are tricks to make this faster that takes structure into account.

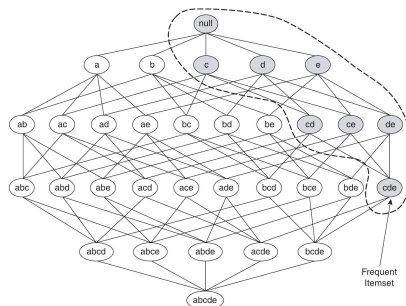
# FINDING ASSOCIATION RULES

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

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  $p = 50$  items

For example:

- ▶ ordinal (2 items): Pick the item based on value being  $\leq$  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

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

$$\left[ \begin{array}{lcl} \text{language in home} & = & \textit{English} \\ \text{householder status} & = & \textit{own} \\ \text{occupation} & = & \{\textit{professional/managerial}\} \end{array} \right] \\ \Downarrow \\ \text{income} \geq \$40,000$$

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

$$\left[ \begin{array}{lcl} \text{language in home} & = & \textit{English} \\ \text{income} & < & \$40,000 \\ \text{marital status} & = & \textit{not married} \\ \text{number of children} & = & 0 \end{array} \right] \\ \Downarrow \\ \text{education} \notin \{\textit{college graduate, graduate study}\}$$