CSE514 – Datamining
Fall 2024

Association Rules Mining

Cynthia Ma

Department of Computer Science
Washington University in St. Louis

czma@wustl.edu

Frequent pattern mining

- Searching for recurring relationships in a data set
- Frequent itemset
  - A set of items that often appear together in a data set
- Frequent sequential pattern
  - A series of items that often occur in sequence
- Frequent structured pattern
  - A structural form like a graph or tree that often appears in ordered data

Market Basket Analysis

- Analyze customer buying habits by finding associations between the different items that customers place in their “shopping basket”
  - Items frequently purchased together can be placed together to encourage combined sales
  - Items frequently purchased together can be place far apart to encourage more customer browsing
  - A sale on one item can encourage purchases of other items that are frequently purchased together

Association Rules Mining

Example rule:

$$eggs \Rightarrow milk$$
$$[support = 5\%, confidence = 50\%]$$

What does 5% support mean?

What does 50% confidence mean?

Rule Structure

For itemset = {Bread, Eggs, Milk}

there exists an association rule:

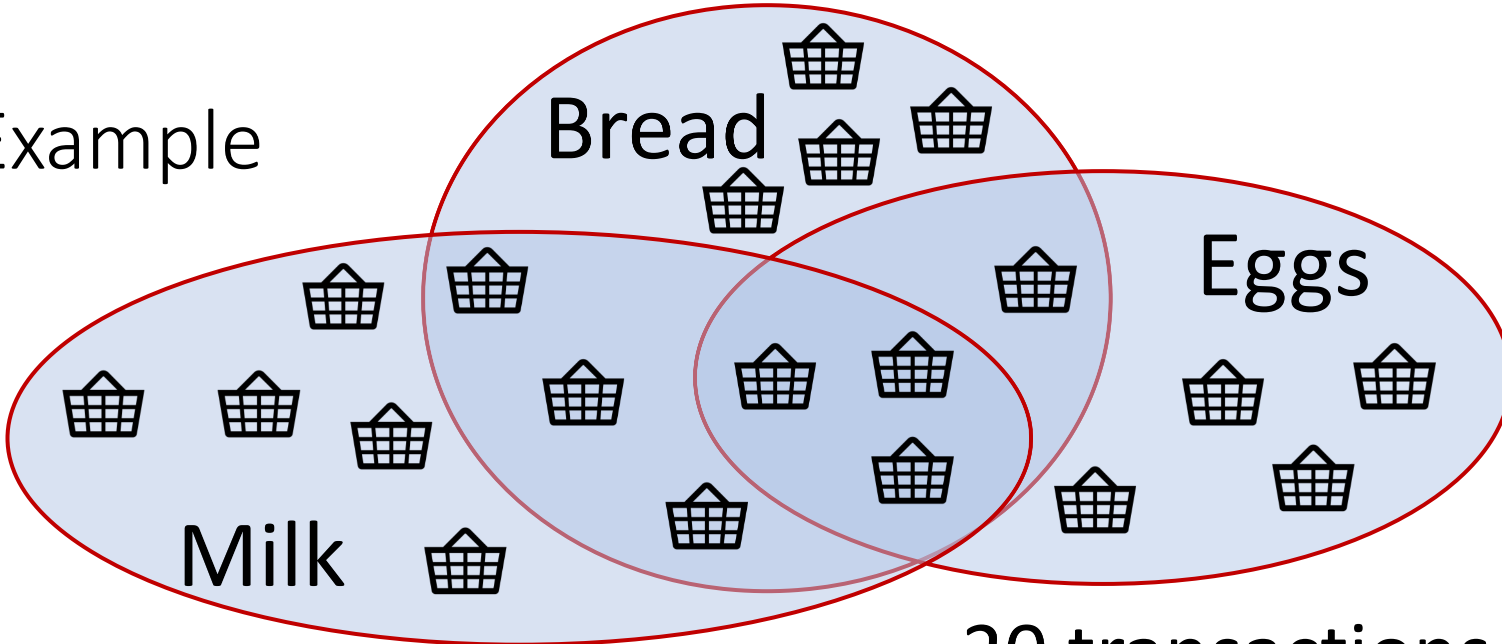
{Bread, Egg} ⇨ {Milk}

Antecedent

Consequent

The implication is co-occurrence, **not** causality

Example



20 transactions total

Itemsets are all subsets of items across transactions

Generating itemsets

The first step to Association Rule Mining is the creation of itemsets from a list of items

This can get very costly!

Ex. List of 10 items has 1000+ possible itemsets

List of 20 items has 1mil+ possible itemsets

How can we avoid having to generate and store so many itemsets?

Filter for frequency by using Support

Support

Support is itemset frequency in all transactions

$$Support(A \Rightarrow B) = Support(C) = P(C)$$
$$\text{where itemset } C = A \cup B$$

$$Support(A \Rightarrow B) = \frac{Transactions\ containing\ A \cup B}{Total\ number\ of\ transactions}$$

- Support never increases as size of itemset increases
- Support does not care about relationships within set

Minimum Support Threshold

- Itemsets can be filtered using a minimum support threshold

- Itemsets that pass the min_Support threshold are considered *frequent*

- This helps with filtering what to store, but what about generating?

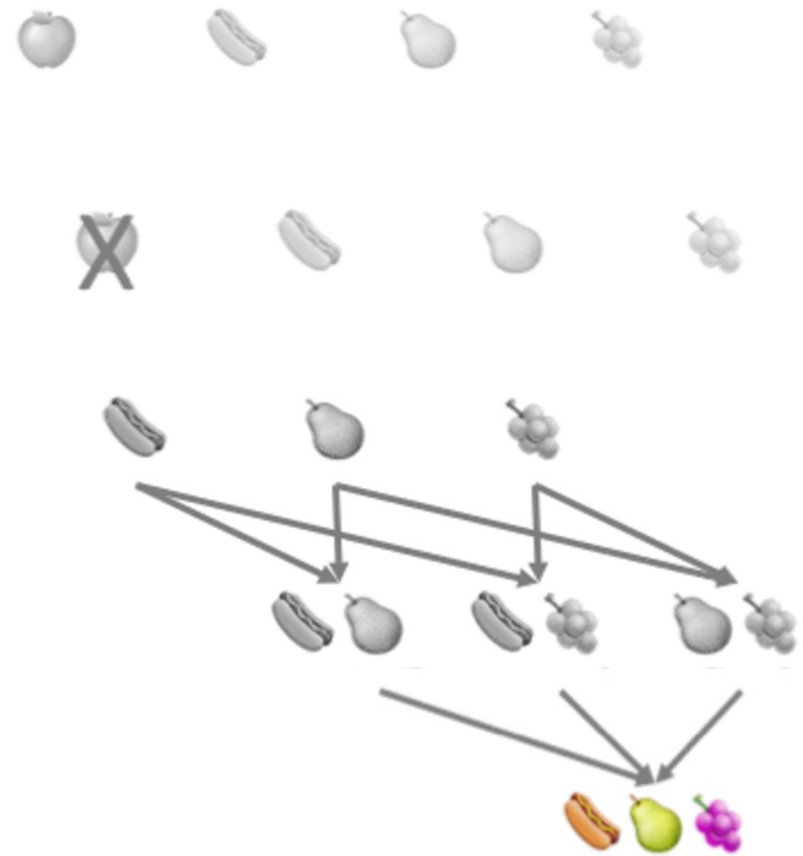
$$Support(A \Rightarrow B) = \frac{Transactions\ containing\ A \cup B}{Total\ number\ of\ transactions}$$

Apriori principle

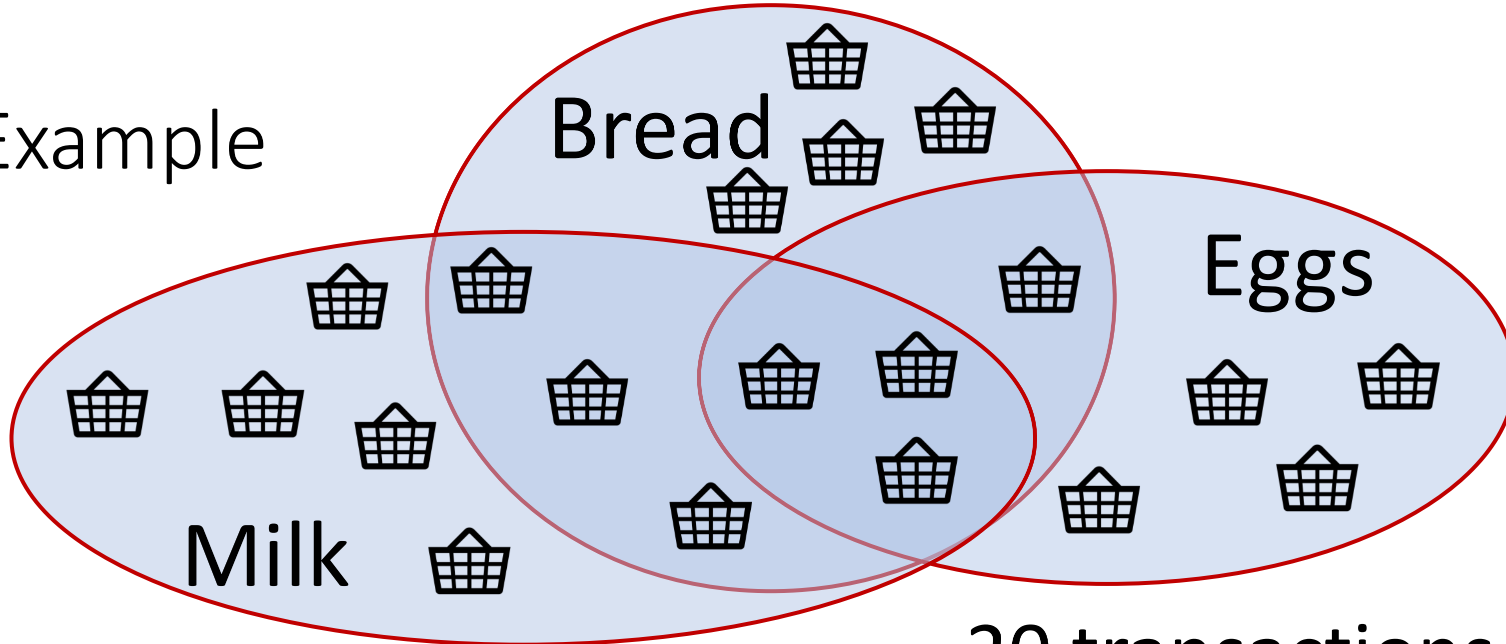
If an itemset is infrequent,
then all its supersets must also be infrequent

Apriori Algorithm

1. Start with itemsets containing single items
2. Calculate itemsets' support. Remove itemsets below minimum support
3. Generate all possible itemsets from merging current itemsets
4. Repeat steps 2 + 3 until no more itemsets to be made



Example



20 transactions total

Itemsets with Support >= 20%:

{Bread} : 11

{Milk} : 11

{Bread, Milk} : 6

{Bread, Eggs, Milk} : 3

{Eggs} : 8

{Bread, Eggs} : 4

{Eggs, Milk} : 3

Example

Rules are binary partitions of itemsets.
For Support >= 20%:

Itemsets:

{Bread}
{Milk}
{Bread, Milk}
{Eggs}
{Bread, Eggs}

Rules:

Generating Association Rules

Once itemsets are defined, candidate rules are **all binary partitions of each itemset**

This can get very costly!

Ex. Itemset of size 3 has 6 possible rules

Itemset of size 10 has 1000+ possible rules

How can we avoid having to generate and store too many rules?

Filter for strength by using Confidence

Confidence

Confidence is likeliness of consequents, given antecedents (already in cart)

$$Confidence(A \Rightarrow B) = P(B|A)$$

$$Confidence(A \Rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

- Rule directions matters!

Minimum Confidence Threshold

- Rules can be filtered using a minimum confidence threshold

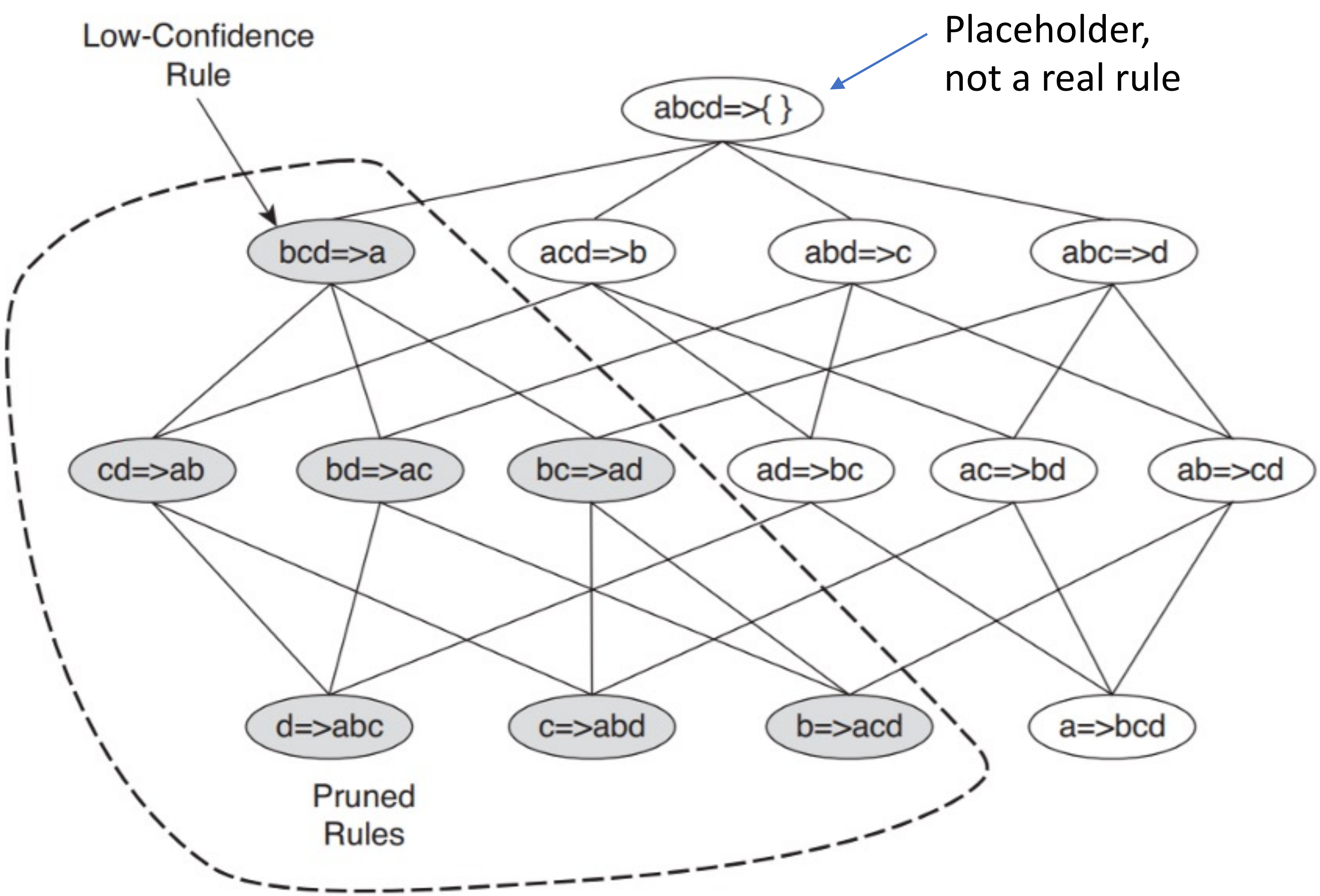
- Rules are **strong** if they pass the minimum confidence threshold using **frequent** itemsets

- This helps with filtering what to store, but what about generating?

$$Confidence(A \Rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

Apriori Principle of confidence

$$Conf(\{A, B, C\} \Rightarrow \{D\}) \geq Conf(\{A, B\} \Rightarrow \{C, D\}) \geq Conf(\{A\} \Rightarrow \{B, C, D\})$$



https://www-users.cs.umn.edu/~kumar001/dmbook/ch6.pdf

Example

$$Confidence(A \Rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

Itemsets with Support >= 20%:

{Bread} : 11 => 55%

{Milk} : 11 => 55%

{Bread, Milk} : 6 => 30%

{Eggs} : 8 => 40%

{Bread, Eggs} : 4 => 20%

Rules with Support >= 20% and Confidence >= 51%:

Example

$$Confidence(A \Rightarrow B) = \frac{Support(A \cup B)}{Support(A)}$$

Itemsets:

{Bread} : 55% {Eggs} : 40% {Milk} : 55%

{Bread, Eggs} : 20% {Bread, Milk} : 30%

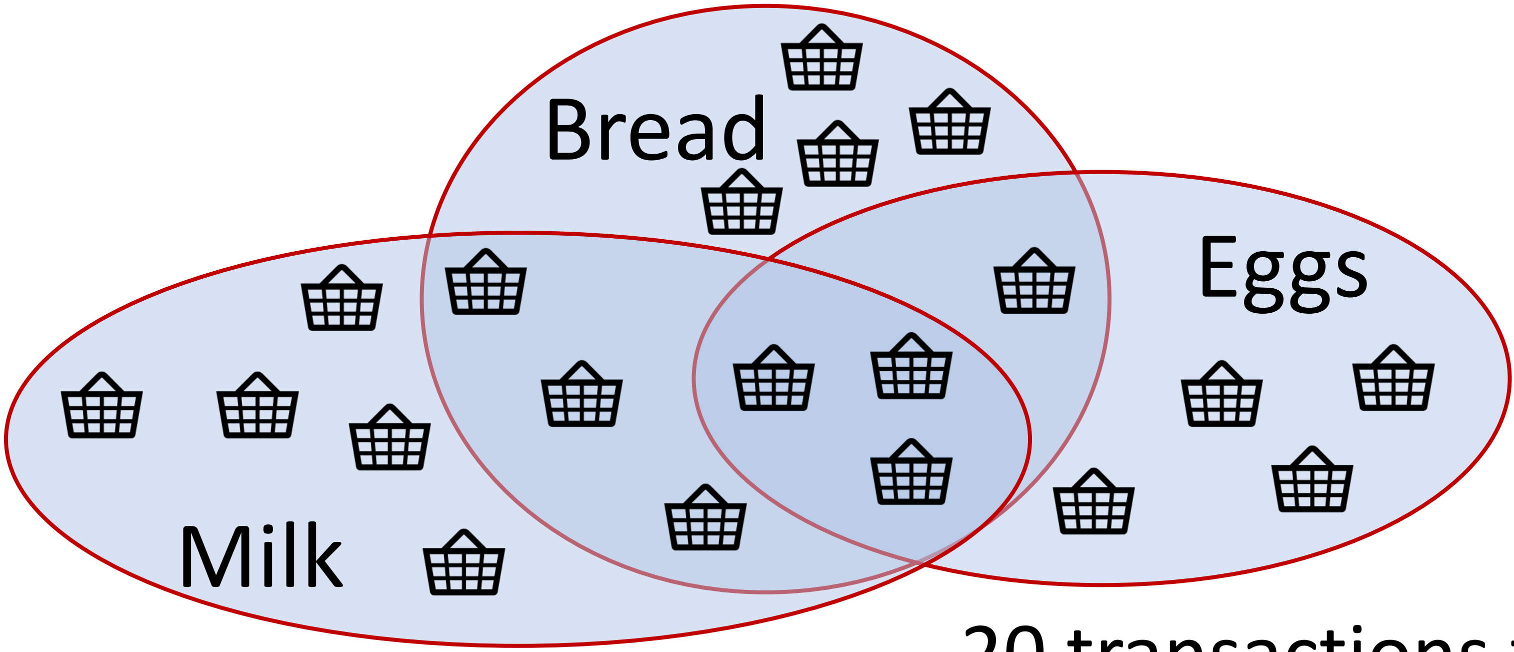
{Eggs, Milk} : 15% {Bread, Eggs, Milk} : 15%

Rules with {Bread, Eggs, Milk}, Confidence >= 51% :

Strong doesn't mean Interesting

{Milk} -> {Bread}

[Support = 30%, Confidence = 54.5%]



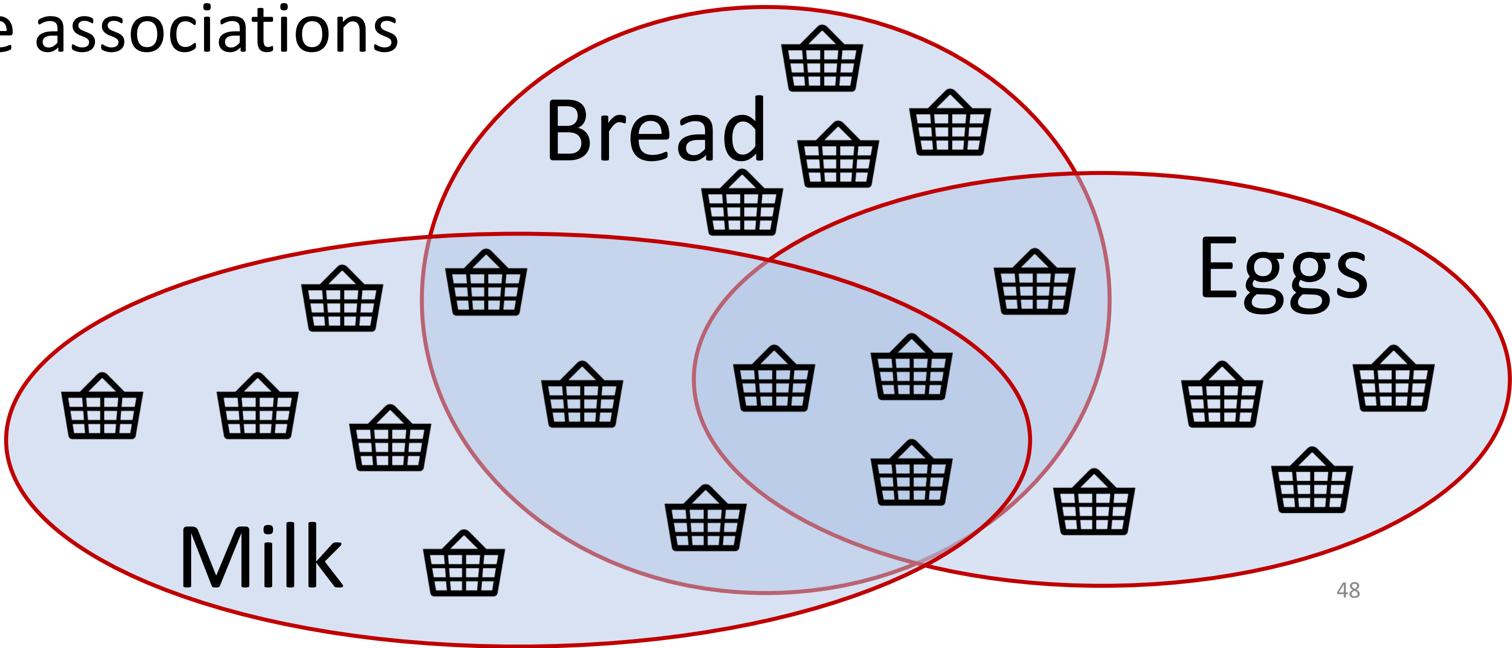
20 transactions total

Strong doesn't mean Interesting

{Milk} -> {Bread}

[Support = 30%, Confidence = 54.5%]

- Seeing milk in the shopping basket gives us a 54.5% chance of also seeing bread
- Bread is bought 55% of the time
- Negative associations are less actionable than positive associations



Lift

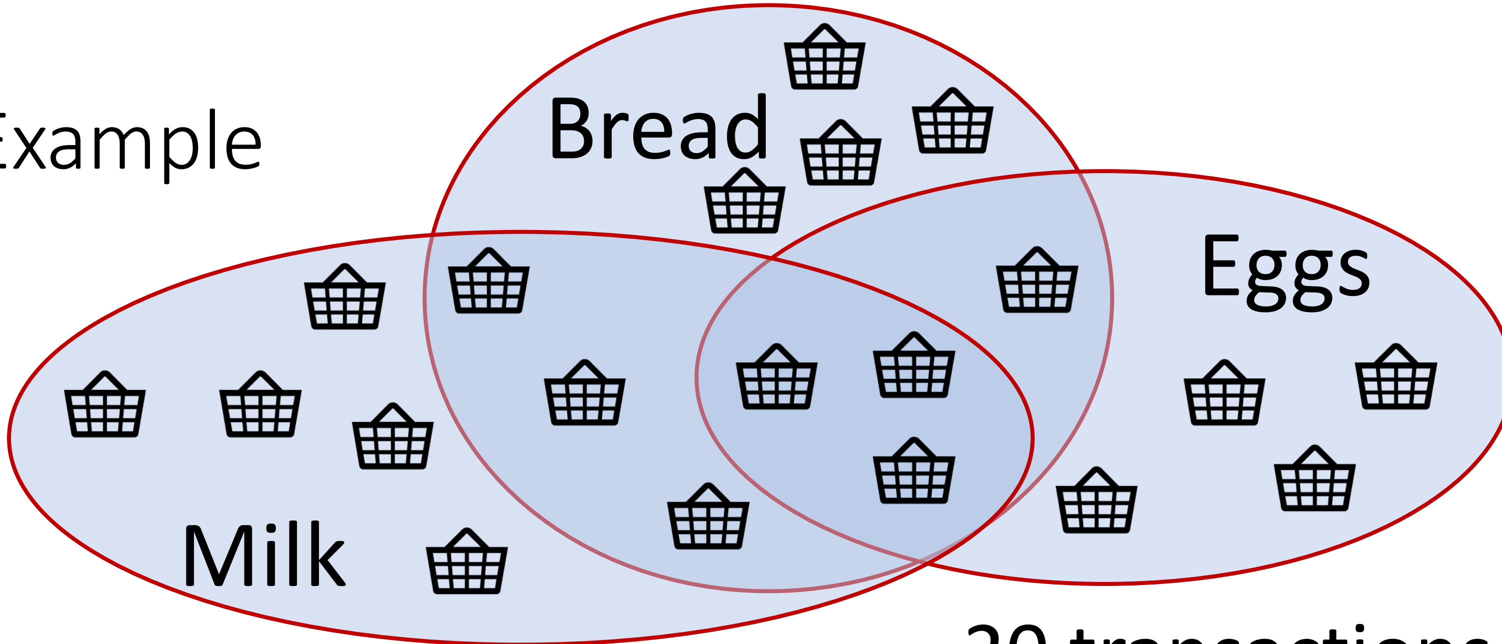
The **rise** in probability of having the consequents
i.e. Controls for *support* while calculating *confidence*

$$Lift(A \Rightarrow B) = \frac{P(A \cup B)}{P(A)P(B)}$$

$$Lift(A \Rightarrow B) = \frac{Confidence(A \Rightarrow B)}{Support(B)}$$

- Lift <1 shows that the antecedent does not increase the probability of the consequent
- The higher the lift, the more informative the rule

Example



20 transactions total

Rules with Support > 20% and Confidence >= 51%:

{Bread} -> {Milk} : Conf = 54.5%　　　Lift =

{Milk} -> {Bread} : Conf = 54.5%　　　Lift =

Neither have Lift > 1, so neither are "interesting"

Useful terms

- **Frequent**: an itemset that passes support threshold
- **Strong**: a rule on a frequent itemset that passes confidence threshold
- Closed: an itemset for which there is no super-itemset with the same support
- Closed Frequent Itemset: a frequent itemset for which there is no super-itemset with the same support
- Maximal Frequent Itemset: a frequent itemset for which there is no super-itemset that is frequent

Coming up

- Office hour today until 5pm
- Homework 10 due Nov. 19
- Programming Assignment 2 due Nov. 20
- Exam 2
  - Wednesday, November 13, Wrighton Hall 300
  - Material from Oct 1 – Nov 6