# Association Rules

## BabyFood $\rightarrow$ Nappies

1. Market basket analysis
2. Wider applications
3. The problems of big data
4. Evaluating rules
5. Support driven rule derivation
6. Supermarket applications

# 1. Market basket analysis

… is based upon:

- Basket data … what's in a basket
  - The term *basket* is clearly taken from supermarket baskets
  - Product data
- Transaction data
- Customer data
- Derived data/variables

# Basket/product data

- A basket contains a set of *line-items*
  - *Each line-item relates to a unique product*
  - *Products exist in categories (and have other attributes)*
- Information about basket data allows us to understand products and their inter-relationships

# Basket data matrix

- Shopping basket data in its simplest form looks like:

| basket | prod1 | prod2 | prod3 | prod4 | … |
|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 1 | … |
| 2 | 1 | 1 | 0 | 0 | … |
| 3 | 0 | 0 | 0 | 0 | … |
| 4 | 1 | 0 | 0 | 1 | … |
| … | … | … | … | … | … |

# Transaction data

- Of course each basket corresponds to a transaction
- Transaction data might include:
  - time/day/date/season/location
  - method of payment
  - gift wrapping
  - delivery address vs. billing address
  - offer responder

# Customer data

- Loyalty cards identify “the customer” and allow us to
  – feed demographic attributes into the analysis
  – link baskets/transactions to tell us who is (and who isn’t) buying what
  – derive other customer-level information, e.g.,
    - Profitability; AVGmonthly_spend; AVGday; AVGmonthly_transactions; …
- Other domains will require different methods of identifying the customer

Sometimes referred to as *association analysis*

# Market-basket analysis

- An *undirected* technique that generates/ identifies patterns in the data
  - Item sets: a *set* of *products* that commonly occur together (in a basket)
    - E.g., {BabyFood, Nappies}
  - Association rules: indicating that the presence of one (or more) product(s) suggests the presence of another
    - E.g., BabyFood → Nappies

Curly brackets { } denote sets

# Association rules

. . . employ *binary* attributes (variables)

- BabyFood → Nappies
- AVGday = Saturday &
  AVGspend > 100 → Profitable
- AVGhighmargingoods > 15 & … → …

with a single variable on the RHS. Note the use of basket/transaction/customer/derived data

# 2. Baskets in other domains

- Items purchased on a credit card
  - also facilitates time-series analysis
- Service/product subscriptions
  - E.g., magazine subscription
- Insurance claims

  X&Y&Z → Fraud

In each case there is a collection that corresponds to the concept of a basket

# Baskets in other domains

- Medical history
- E-Commerce
- Web pages visited by a visitor
  - “Click-stream analysis”
  - Can also consider page order
    - page45 → page57
    - the meaning of → needs defining

We’ll focus mostly on supermarkets

# 3. Big data

- With 20 variables there are
  - 20*19=380 binary rules (X→Y)
  - 380*18=6840 order-3 rules (X&Y→Z)
  - 116280 order-4 rules …
- With 100 variables, the 3 figures above become: 9900; 970200; 94 million
- The complexity of the problem grows exponentially

# Big data for supermarkets

- But a supermarket chain might stock over 30K differing products, with over 1M baskets/week
  - Aside: But half the sales revenue might come from the top-selling 1000 products!
- Product, transaction, customer & derived variables further increase the number of variables: some pruning will be needed

# 4. Evaluating rules

Types of rules

- Actionable rules … suggest a viable business strategy that was not previously obvious: Nappies → Beer
- Trivial rules: Toothbrush → Toothpaste
- Inexplicable rules … explanation is unclear as is any related business strategy
- How do we prune the rules generated to allow us to find those that matter?

# Mathematical evaluation

**Support** (X→ Y) **= %** baskets with X & Y =

#baskets with X&Y / #baskets

How often do the products occur together? If support is low, then only a small percentage of the baskets contain both X and Y and therefore a business strategy to get them to be purchased together might not be sensible

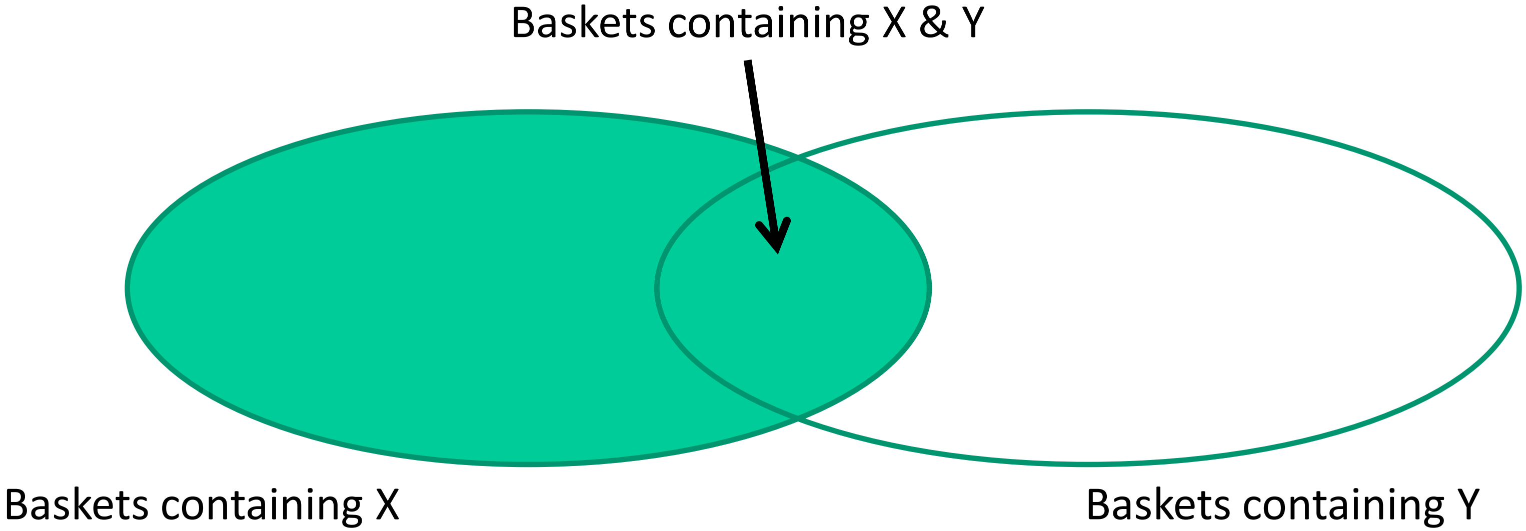
# Mathematical evaluation

**Confidence** $(X \rightarrow Y)$ = predictive accuracy of the rule

= probability that a basket that contains X will also contain Y

= #baskets with X&Y / #baskets with X

= %baskets with X&Y / %baskets with X



# Mathematical evaluation

**Lift** (X→Y) = the predictive accuracy of the rule relative to (divided by) the predictive accuracy of random guessing

The predictive accuracy of random guessing =

% baskets with Y

$$\textbf{Lift}\ (X \rightarrow Y) = \frac{\text{\% baskets containing X \& Y}}{(\text{\% baskets containing X}) \ * \ (\text{\% baskets containing Y})}$$

# Example

- Suppose we have 1000 baskets of which 400 contain Z; 300 contain X&Y; 250 contain X&Y&Z
- Support(X & Y → Z) = 250/1000 = 0.25
- Confidence (X & Y → Z) = 250/300 = 5/6 = predictive accuracy of the rule
  - Of the 300 baskets that contain X&Y – how many contain Z?
- Lift (X & Y → Z) = (5/6)/(4/10) = 50/24 . . .

  … the rule's accuracy is about twice that of random guessing

# Caviar → Vodka

- In some countries, vodka is a traditional accompaniment to caviar
- Of course caviar is expensive and therefore not commonly purchased (unlike vodka)
- Good confidence & lift, but poor support
- Not applicable to a sufficiently large segment of the population …

# Vodka → Caviar

- Good lift
  - The lift of Vodka → Caviar is the same as that of Caviar → Vodka
- … but poor confidence (& support) due to the rarity of caviar purchase
- Not applicable to a sufficiently large segment of the population …

# Apples → Milk

- High support
  - two very popular products
- High confidence
  - as a result of the fact that everyone buys Milk
- … but lift = 1
  - buying apples has nothing to do with buying milk

# Other rules

- Pepsi → Coke … lift < 1
  - When lift < 1 the negated rule can be more useful: Pepsi → not Coke
- Toothpaste → Toothbrush … a trivial rule
  - Some trivial rules may have confidence close to or equal to 1

# 5. Rule derivation

- … support driven: suppose that we insist upon a threshold *T* for rule support
- Note that rule support does not depend upon the rule structure, i.e.,

$$\text{support}(X\&Y \rightarrow Z) = \text{support}(\{X,Y,Z\})$$

- Also, a basket containing X&Y contains X, therefore $\text{support}(\{X,Y\}) \leq \text{support}(\{X\})$
- Similarly $\text{support}(\{X,Y,Z\}) \leq \text{support}(\{X,Y\})$, …

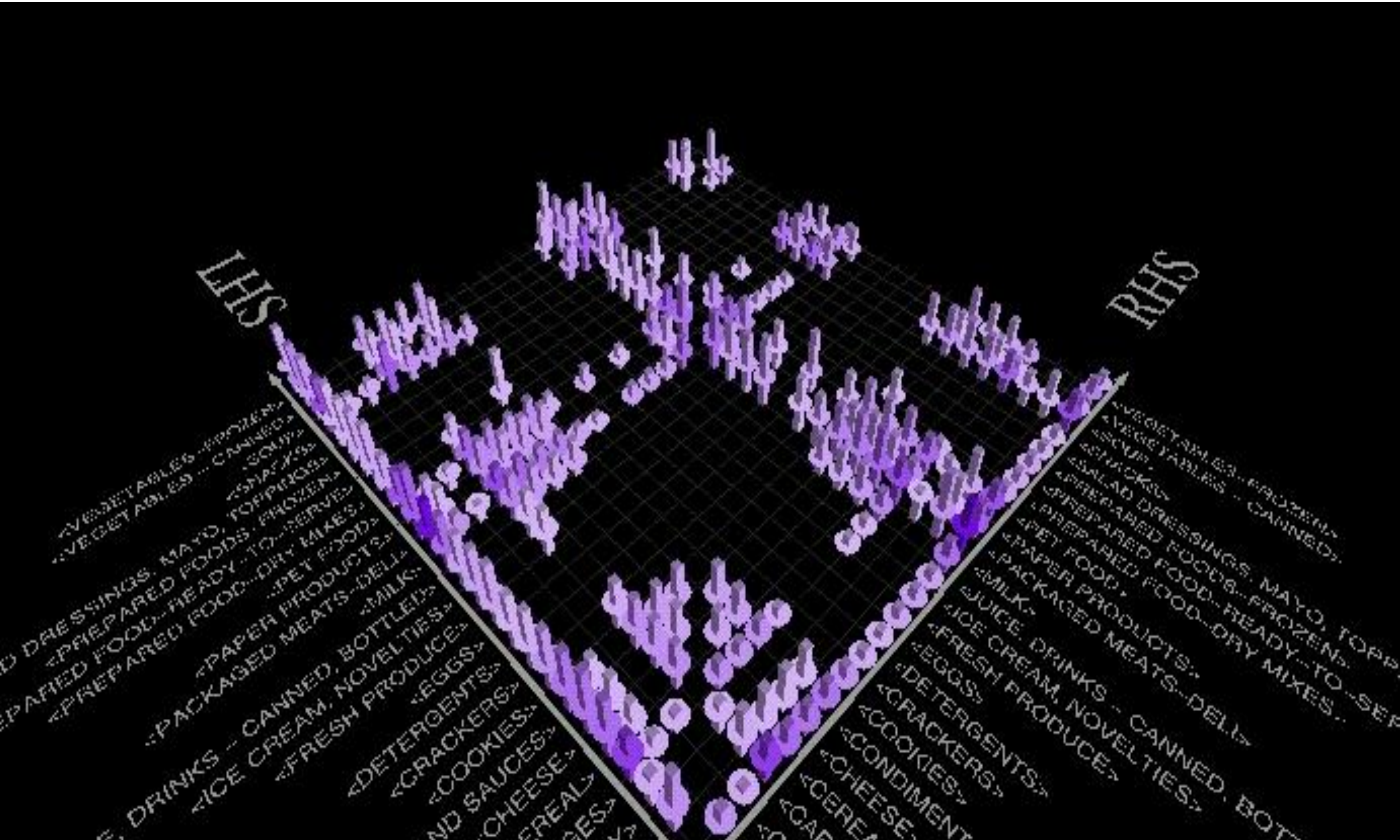
# The algorithm

- Let S1 = set of variables X where support(X) > T
  - we can now ignore variables not in S1
- Let S2 = set of pairs {X,Y} where
  - X,Y are in S1 and support({X,Y})>T
  - we can now ignore pairs not in S2
- Let S3 = set of triples {X,Y,Z} where
  - {X,Y}, {X,Z} and {Y,Z} are in S2, and
  - support({X,Y,Z})>T

# Pruning when applying the tool

- A tool will generate rules together with their support, confidence and lift
  - Only business insight can identify the really interesting rules
- We can ask the tool to
  - Generate only rules that pass given thresholds on support, confidence & lift, **and**
  - Order the rules produced by decreasing support, confidence **or** lift
  - The business goals need to be considered here …

# Association rule visualisation



# 6. Supermarket applications

- Understanding which products are sold together
  - placing things together on the shelves (merchandising)
  - making sure that related products are not discounted at the same time
  - designing product packages
- Understanding who buys what
  - Supermarkets as information brokers
  - To understand a product-group from a particular manufacturer, we might include 2 binary variables
    - P : the basket contains an item from the given product group
    - C : the basket contains a competitor brand

# Using product categories

- Product categories provide a means of reducing the number of variables, e.g.,
- Frozen food
  - Frozen Veg
    - Peas, Carrots, Chips, …
  - Frozen desserts
    - Ice cream, frozen cake, …
  - Frozen meals
    - …

# Using product categories

- How – and to what extent – base products are aggregated into categories depends upon the application
  - product type/size/brand/diet vs non-diet/…
- Some products can be aggregated whilst others – e.g., a product of specific interest – are not
- Aggregated data produces more general results
  - Can subsequently drill down into the data

# Transaction variables

- E.g., suppose that were interested in understanding whether the sale of some product(s) were time-dependent.   We might then add variables of the form
  - am/pm/evening (3 binary variables), and/or
  - mon/tues/ … /sun (7 binary variables), and/or
  - day1, day2, …, day31 (day of the month; 31 binary variables), etc, etc.

# Profitability

- We might add derived variables to represent
  - Basket profitability or
  - Customer profitability (assuming of course we have a loyalty card)
- Are we making an overall profit on discounted ranges/express lane customers?
- Product discontinuation: if a product sells very little, is it worth keeping?

# Marketing

- Discount coupons at the till
  - to customers who’ve just bought a competitor brand?
  - to customers who should want your product and have not just bought a competitor brand
- Targetted marketing
  - Coupons in the post
  - Which additional product might we get this customer to buy?
  - Which additional products do we want this customer to buy … up-selling

# Applications

- Reward for monthly spend > threshold
  - cannot simply apply uniform threshold since doing so does not modify behaviour
  - apply data mining to choose the reward!
    - Cross-selling, Up-selling