

Lecture 10: Market basket Analysis

March 12, 2019

Market Basket - Introduction

Looking for rules to predict if something X is bought, what else is likely to be bought

Market Basket - Introduction



Beer and Nappies

There was no good statistical basis for this link, but the story has become well known, one of the first to 'go viral'

Market Basket - Introduction

Each transaction is a set of items, a basket, called here an *itemset*
This allows companies to understand why people make certain purchases

Market Basket - Applications

Insight can be gained about the products they sell

- ▶ Which sell quickly or slowly?
- ▶ Which are bought together?
- ▶ Identify possible missed opportunities

This helps companies to decide on:

- ▶ How to layout a shop?
- ▶ Which products to promote?

E. g. if one specific product (e.g. "Earl Grey Redbush Tea") is only rarely bought, but when it is bought that same customer spends lots of money on other products, is it worth keeping it just for that person?

Market Basket - Applications

Other applications include:

- ▶ communication (set of phone calls)
- ▶ banks (each account is a transaction)
- ▶ Medical Treatment (a patient is a transaction with a set of diseases!)

The maths and algorithms are very similar for all.

Market Basket - Definitions

Definitions:

- ▶ $I = i_1, i_2, \dots, i_n$ is a set of all items
- ▶ Transaction t_i is a set of items such that $t_i \subseteq I$ (basket)
- ▶ Transaction database D contains all transactions t_1, \dots, t_d
- ▶ An **Association Rule** is where $X \implies Y$, i.e. X implies Y
- ▶ An **itemset** is a set of items. If it has k items, it is a k - itemset
- ▶ **Support** s of an itemset X is the percentage of transactions in D that contain X
- ▶ **Support** of **association rule** $X \implies Y$ is the support of the itemset $\{X, Y\}$
- ▶ **Confidence** of the rule $X \implies Y$ is the ratio between the transactions that contain both X and Y and the number of transactions that have X in D

Market Basket - Problem

Problem: Find association rules

Given:

- ▶ a set I of items
- ▶ database D of transactions
- ▶ minimum support s
- ▶ minimum confidence c

Find: Association rules $X \implies Y$ with a minimum support s and minimum confidence c

Market Basket - Problem

Solution

- ▶ Find all itemsets that have minimum support
- ▶ Generate rules using frequent itemsets

9 / 26

Market Basket - Association Rule Mining

Using this transaction

database D

Find most frequent *itemsets*

Transaction Itemsets

t_1 A, B, C

t_2 A, C

t_3 A, C, D

t_4 A, E

t_5 D, E

itemsets frequency support

{A} 4 0.8

{B} 1 0.2

{C} 3 0.6

{D} 2 0.4

{E} 2 0.4

{A, B} 1 0.2

{A, C} 3 0.6

{A, D} 1 0.2

{A, E} 1 0.2

{B, C} 1 0.2

{D, E} 1 0.2

{A, B, C} 1 0.2

{A, C, D} 1 0.2

$$\text{support} = \frac{\text{freq}(\text{item})}{n}$$

Where n = number of transactions

10 / 26

Market Basket - Association Rule Mining

With minimum support 0.4:

itemsets frequency support

{A} 4 0.8

{B} 1 0.2

{C} 3 0.6

{D} 2 0.4

{E} 2 0.4

{A, B} 1 0.2

{A, C} 3 0.6

{A, D} 1 0.2

{A, E} 1 0.2

{B, C} 1 0.2

{D, E} 1 0.2

{A, B, C} 1 0.2

{A, C, D} 1 0.2

itemsets frequency support

{A} 4 0.8

{C} 3 0.6

{D} 2 0.4

{E} 2 0.4

{A, C} 3 0.6

So the only rules we can examine are $A \Rightarrow C$ or $C \Rightarrow A$

assn rules support confidence

$A \Rightarrow C$ 0.6 0.75

$C \Rightarrow A$ 0.6 1.00

11 / 26

Market Basket - A Priori Algorithm

The Apriori Algorithm

We know:

- ▶ Any subset of a *frequent itemset* is also frequent
- ▶ Any superset of an infrequent itemset is also infrequent

Let:

- ▶ L_k = set of frequent k – *itemsets* (have minimum support)
- ▶ C_k = set of candidate k – *itemsets* (potentially frequent)

12 / 26

Market Basket - A Priori Algorithm

Algorithm 1: A Priori Algorithm

Data: D transaction database, $minSupport$

$L_1 = \{\text{frequent items}\};$

$k = 1;$

while $L_k \neq \emptyset$ **do**

$C_{k+1} = \text{all possible candidates from } L_k;$

for each transaction t **in** D **do**

if candidate in C_{k+1} is in t **then**

 increment count for candidate;

end

end

$L_{k+1} = \text{candidates in } C_{k+1} \text{ with } minSupport;$

$k = k + 1;$

end

13 / 26

Market Basket - A Priori Algorithm

Algorithm 2: A Priori Algorithm - Generating Candidates

Data: L_{i-1}

$C_i = \{\};$

for each itemset J **in** L_{i-1} **do**

for each itemset K **in** L_{i-1} **such that** $K \neq J$ **do**

if $i - 2$ elements in J and K are equal **then**

if all subsets of $\{K \cup J\}$ are in L_{i-1} **then**

$C_i = C_i \cup \{K \cup J\};$

end

end

end

end

return $C_i;$

14 / 26

Market Basket - A Priori Algorithm

$minSupport = 0.5$

Database D :

Transaction	Basket	itemset	support
t_1	A, C, D	{A}	0.5
t_2	B, C, E	{B}	0.75
t_3	A, B, C, E	{C}	0.75
t_4	B, E	{D}	0.25
		{E}	0.75

$k = 1,$

Go through D :

So $L_1 = \{A, B, C, E\}$

$\therefore C_2 =$

itemset	support
{A, B}	0.25
{A, C}	0.5
{A, E}	0.25
{B, C}	0.5
{B, E}	0.75
{C, E}	0.5

So $L_2 = \{ \{A, C\}, \{B, C\}, \{B, E\}, \{C, E\} \}$

15 / 26

Market Basket - A Priori Algorithm

$k = 3$

$L_2 = \{ \{A, C\}, \{B, C\}, \{B, E\}, \{C, E\} \}$

Generating Candidates:

$\{A, C\}, \{B, C\}$ are both in L_2 , giving $\{A, B, C\}$

Not all subsets of $\{A, B, C\}$ are in L_2

$\{A, C\}, \{C, E\}$ are both in L_2 giving $\{A, C, E\}$

Not all subsets of $\{A, C, E\}$ are in L_2

$\{B, C\}, \{B, E\}$ are both in L_2 giving $\{B, C, E\}$

All subsets of $\{B, C, E\}$ are in L_2 so:

Go through D :

itemset	support
$\{B, C, E\}$	0.5

16 / 26

Market Basket - Generating Rules

Consider 3-itemset {B, C, E}

Use all permutations of rules from these three items

$\{B, C\} \Rightarrow E$

$\{B, E\} \Rightarrow C$

$\{C, E\} \Rightarrow B$

$E \Rightarrow \{B, C\}$

$C \Rightarrow \{B, E\}$

$B \Rightarrow \{C, E\}$

Market Basket - A Priori Algorithm

Algorithm 3: A Priori Algorithm - Generating Candidates

Data: L_{i-1}

$C_i = \{\};$

for each frequent itemset I **do**

for each subset C of I **do**

if $\text{support}(I)/\text{support}(I - C) \geq \text{minConf}$ **then**

 output rule $(I - C) \Rightarrow C;$

 with confidence = $\text{support}(I) / \text{support}(I - C);$

 and support = $\text{support}(I);$

end

end

end

Market Basket - A Priori Algorithm

Advantages of A Priori Algorithm:

- ▶ Uses large itemset property
- ▶ Can be Parallelised
- ▶ Easy to implement

Disadvantages

- ▶ Assumes D transaction database is in memory
- ▶ Requires many database scans

Market Basket - Improvements

Confidence of a rule is the ratio between transactions with $X \cup Y$ to the number of transactions with X

$$\text{conf}(X \Rightarrow Y) = \frac{\frac{n\text{Trans}(X \cup Y)}{|D|}}{\frac{n\text{Trans}(X)}{|D|}} = \frac{p(X \wedge Y)}{p(X)} = p(Y|X)$$

If Y is independent of X : $p(Y) = p(Y - X)$

This means if you have a high probability of $p(Y)$ we have a rule with high confidence that associates independent itemsets
e.g. if $p(\text{"bread"}) = 0.8$, and "bread" is independent from "sausages", then the rule "bread" \Rightarrow "sausages" will have confidence 0.8

Market Basket - Improvements

Alternative measures:

lift measure indicates departure from independence of X and Y
the **lift** of $X \implies Y$ is:

$$\text{lift}(X \implies Y) = \frac{\text{conf}(X \implies Y)}{p(Y)} = \frac{\frac{p(X \wedge Y)}{p(X)}}{p(Y)} = \frac{p(X \wedge Y)}{p(X)p(Y)}$$

Unfortunately, lift is *symmetric*, the same for $X \implies Y$ as
 $Y \implies X$

Market Basket - Improvements

Conviction indicates that X and Y are not independent, and
takes in to account the direction of implication
The conviction of $X \implies Y$ is:

$$\text{conv}(X \implies Y) = \frac{p(X)p(\neg Y)}{p(X \wedge \neg Y)}$$

Market Basket - Linked Concepts

"Baskets" = **documents**

"items" = **words** in those documents

If we can find words that appear together more often than others,
these are **linked concepts**

	word1	word2	word3	word4
doc1	1	0	1	1
doc2	0	0	1	1
doc3	0	1	1	0

$\therefore \text{word3} \implies \text{word4}$

As when *word4* occurs, there is a large probability that *word3* will
also occur

Market Basket - Linked Concepts

Detecting Plagiarism

"Baskets" = **sentences**

"items" = **documents** containing those sentences

Items that appear together could mean that a student has copied
work from another document, plagiarism!

	doc1	doc2	doc3	doc4
sent1	1	0	1	1
sent2	0	0	1	1
sent3	0	1	1	0

Here..

$\therefore \text{doc4} \implies \text{doc3}$

If there is a sentence occurring in document 4, there is a high
probability of it occurring in document 3, so if *doc3* is your
coursework, you may be in trouble!

Market Basket - Linked Concepts

Web pages

"Baskets" = **web pages**

"items" = **linked pages**

Pairs of pages with many common references may be about the same topic

"Baskets" = **web pages**, p_1

"items" = **pages that link to** p_1

Pages with many of the same links may be mirrors or about the same topic

Market Basket - Summary

Association rules form a very applied data mining approach

Many uses:

- ▶ Commercial
- ▶ Text analysis
- ▶ Medicine

Association rules are derived from frequent itemsets

The A Priori algorithm:

- ▶ searches level wide
- ▶ uses frequent item property

The resulting rules can be measured many ways, including

Confidence, Lift, Conviction