Market Basket Analysis Using the Apriori Algorithm

Andrex Ibiza, MBA

2024-04-22

Market Basket Analysis in R Using the Apriori Algorithm

The dataset contains the following columns:

- `InvoiceNo` : The invoice number. If this code starts with letter 'C', it indicates a cancellation.
- `StockCode` : Product (item) code.
- `Description` : Product (item) description.
- `Quantity` : The quantities of each product (item) per transaction.
- `InvoiceDate` : The date and time when each transaction occurred.
- `UnitPrice` : Product price per unit.
- `CustomerID` : Customer number.
- `Country` : Country name.

```
# Load packages
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.3.2
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

```
## Warning: package 'tidyr' was built under R version 4.3.3
```

```
## Warning: package 'readr' was built under R version 4.3.3
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
## Warning: package 'stringr' was built under R version 4.3.3
```

```
## Warning: package 'lubridate' was built under R version 4.3.3
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.4     ✓ readr     2.1.5
## ✓ forcats   1.0.0     ✓ stringr   1.5.1
## ✓ ggplot2   3.5.0     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.1
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(arules)
```

```
## Warning: package 'arules' was built under R version 4.3.3
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 4.3.3
```

```
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
##
## Attaching package: 'arules'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following objects are masked from 'package:base':
##
##     abbreviate, write
```

```
# Read and preprocess data
data <- read.csv("online_retail.csv", stringsAsFactors = FALSE)
data <- data[!grepl("^C", data$InvoiceNo), ]  # Remove cancellations
data <- data[!is.na(data$Description), ]  # Remove missing descriptions
data <- data[data$Quantity > 0, ]  # Only positive quantities

# Aggregate items into transactions
transactions <- data %>%
  group_by(InvoiceNo) %>%
  summarise(Items = paste(unique(Description), collapse = ",")) %>%
  ungroup()

# Convert to transactions class
trans_list <- strsplit(as.character(transactions$Items), ",")
trans <- as(trans_list, "transactions")
```

```
## Warning in asMethod(object): removing duplicated items in transactions
```

```
# Run apriori algorithm
frequent_itemsets <- apriori(trans,
                             parameter = list(supp = 0.05,  # Adjusted support threshold
                                              conf = 0.1,   # Adjusted confidence threshold
                                              target = "frequent itemsets"))
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         NA    0.1    1 none FALSE            TRUE       5    0.05      1
##  maxlen            target  ext
##      10 frequent itemsets TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 1036
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[4125 item(s), 20728 transaction(s)] done [0.10s].
## sorting and recoding items ... [29 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 done [0.00s].
## sorting transactions ... done [0.00s].
## writing ... [29 set(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# Inspect the frequent itemsets
inspect(head(sort(frequent_itemsets, by="support"), 10))
```

```
##      items                                support    count
## [1]  {WHITE HANGING HEART T-LIGHT HOLDER} 0.10903126 2260
## [2]  {JUMBO BAG RED RETROSPOT}            0.10092628 2092
## [3]  {REGENCY CAKESTAND 3 TIER}           0.09595716 1989
## [4]  {PARTY BUNTING}                      0.08133925 1686
## [5]  {LUNCH BAG RED RETROSPOT}            0.07545349 1564
## [6]  {ASSORTED COLOUR BIRD ORNAMENT}      0.07019491 1455
## [7]  {SET OF 3 CAKE TINS PANTRY DESIGN }  0.06681783 1385
## [8]  {PACK OF 72 RETROSPOT CAKE CASES}    0.06368198 1320
## [9]  {LUNCH BAG  BLACK SKULL.}            0.06141451 1273
## [10] {NATURAL SLATE HEART CHALKBOARD }    0.06025666 1249
```

```
# Calculate association rules using the apriori algorithm
rules <- apriori(trans,
                 parameter = list(supp = 0.025,  # Support threshold
                                  conf = 0.1),  # Confidence threshold
                 appearance = NULL,
                 control = NULL,
                 target = "rules")
```

```
## Apriori
##
## Parameter specification:
##  confidence minval smax arem  aval originalSupport maxtime support minlen
##         0.1    0.1    1 none FALSE            TRUE       5   0.025      1
##  maxlen target  ext
##      10  rules TRUE
##
## Algorithmic control:
##  filter tree heap memopt load sort verbose
##     0.1 TRUE TRUE  FALSE TRUE    2    TRUE
##
## Absolute minimum support count: 518
##
## set item appearances ...[0 item(s)] done [0.00s].
## set transactions ...[4125 item(s), 20728 transaction(s)] done [0.10s].
## sorting and recoding items ... [177 item(s)] done [0.00s].
## creating transaction tree ... done [0.00s].
## checking subsets of size 1 2 3 done [0.00s].
## writing ... [59 rule(s)] done [0.00s].
## creating S4 object  ... done [0.00s].
```

```
# Inspect the top 10 association rules sorted by confidence
inspect(head(sort(rules, by="confidence"), 10))
```

```
##      lhs                                    rhs                                    support confidence   coverage      li
ft count
## [1]  {PINK REGENCY TEACUP AND SAUCER,
##       ROSES REGENCY TEACUP AND SAUCER }   => {GREEN REGENCY TEACUP AND SAUCER}   0.02614821  0.9048414 0.02889811 18.4783
77   542
## [2]  {GREEN REGENCY TEACUP AND SAUCER,
##       PINK REGENCY TEACUP AND SAUCER}     => {ROSES REGENCY TEACUP AND SAUCER }  0.02614821  0.8562401 0.03053840 16.6492
92   542
## [3]  {PINK REGENCY TEACUP AND SAUCER}     => {GREEN REGENCY TEACUP AND SAUCER}   0.03053840  0.8263708 0.03695484 16.8758
75   633
## [4]  {PINK REGENCY TEACUP AND SAUCER}     => {ROSES REGENCY TEACUP AND SAUCER }  0.02889811  0.7819843 0.03695484 15.2054
14   599
## [5]  {GREEN REGENCY TEACUP AND SAUCER}    => {ROSES REGENCY TEACUP AND SAUCER }  0.03705133  0.7566502 0.04896758 14.7128
01   768
## [6]  {ROSES REGENCY TEACUP AND SAUCER }   => {GREEN REGENCY TEACUP AND SAUCER}   0.03705133  0.7204503 0.05142802 14.7128
01   768
## [7]  {GARDENERS KNEELING PAD CUP OF TEA } => {GARDENERS KNEELING PAD KEEP CALM } 0.02634118  0.7203166 0.03656889 16.3534
75   546
## [8]  {GREEN REGENCY TEACUP AND SAUCER,
##       ROSES REGENCY TEACUP AND SAUCER }   => {PINK REGENCY TEACUP AND SAUCER}    0.02614821  0.7057292 0.03705133 19.0970
68   542
## [9]  {CHARLOTTE BAG PINK POLKADOT}        => {RED RETROSPOT CHARLOTTE BAG}       0.02518333  0.7025572 0.03584523 14.0837
58   522
## [10] {JUMBO BAG PINK POLKADOT}            => {JUMBO BAG RED RETROSPOT}           0.03980124  0.6773399 0.05876110  6.7112
34   825
```

```
library(arulesViz)
```

```
## Warning: package 'arulesViz' was built under R version 4.3.3
```

```
inspectDT(rules)
```

Show 10 entries Search:

	LHS	RHS	support	confidence	coverage	lift	count
	All	All	All	All	All	All	All
[1]	{}	{WHITE HANGING HEART T-LIGHT HOLDER}	0.109	0.109	1.000	1.000	2,260.000
[2]	{}	{JUMBO BAG RED RETROSPOT}	0.101	0.101	1.000	1.000	2,092.000
[3]	{WOODEN FRAME ANTIQUE WHITE }	{WOODEN PICTURE FRAME WHITE FINISH}	0.026	0.555	0.047	10.460	539.000
[4]	{WOODEN PICTURE FRAME WHITE FINISH}	{WOODEN FRAME ANTIQUE WHITE }	0.026	0.490	0.053	10.460	539.000
[5]	{JUMBO BAG STRAWBERRY}	{JUMBO BAG RED RETROSPOT}	0.026	0.651	0.040	6.449	537.000
[6]	{JUMBO BAG RED RETROSPOT}	{JUMBO BAG STRAWBERRY}	0.026	0.257	0.101	6.449	537.000
[7]	{PINK REGENCY TEACUP AND SAUCER}	{ROSES REGENCY TEACUP AND SAUCER }	0.029	0.782	0.037	15.205	599.000
[8]	{ROSES REGENCY TEACUP AND SAUCER }	{PINK REGENCY TEACUP AND SAUCER}	0.029	0.562	0.051	15.205	599.000
[9]	{PINK REGENCY TEACUP AND SAUCER}	{GREEN REGENCY TEACUP AND SAUCER}	0.031	0.826	0.037	16.876	633.000

	LHS	RHS	support	confidence	coverage	lift	count
[10]	{GREEN REGENCY TEACUP AND SAUCER}	{PINK REGENCY TEACUP AND SAUCER}	0.031	0.624	0.049	16.876	633.000

Showing 1 to 10 of 59 entries Previous 1 2 3 4 5 6 Next

```
# This visualization is INTERACTIVE. Please explore!
library(plotly)
```

```
## Warning: package 'plotly' was built under R version 4.3.3
```

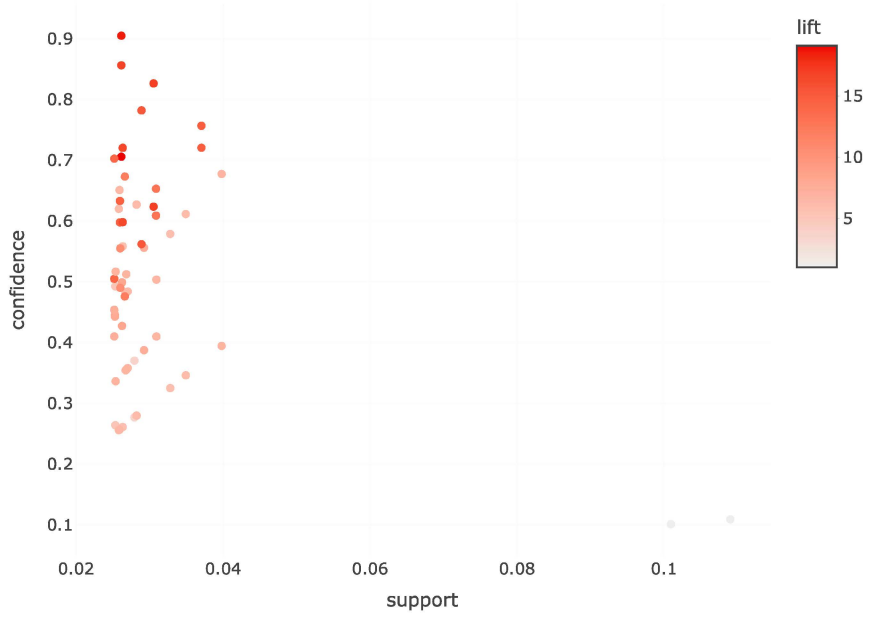
```
##
## Attaching package: 'plotly'
```

```
## The following object is masked from 'package:ggplot2':
##
##     last_plot
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following object is masked from 'package:graphics':
##
##     layout
```

```
plot(rules, engine = "plotly", )
```



Key Findings

The market basket analysis, conducted using the Apriori algorithm on online retail data, revealed interesting insights into customer purchasing behavior. Several iter
LIGHT HOLDER" and "JUMBO BAG RED RETROSPOT," as well as baking
related products such as the "REGENCY CAKESTAND 3 TIER" and "PACK OF 72 RETROSPOT CAKE CASES."

Association rules further highlighted relationships between items. Strong associations were found among teacup and saucer sets, with purchases of one color frequ