Introduction to association rule learning in R with the arules package

Alexander C. Mueller, PhD

August 9, 2018

Who is the speaker?

30 second resume:

- an ancient metro Saint Louis townie
- grew up in University City
- University City High School 2003
- B.A. Washington University 2007 (econ and math)
- Ph.D. University of Michigan 2013 (math)
- a few years of data science
- founded data privacy company Capnion

Agenda

Today, we will talk about...

- Examples and Lore
- What is association rule learning?
- Example point of sale-data
- Association rule differentiators
- Rule metrics
- Data wrangling and input format
- Computing and visualizing

Example: E-Commerce Recommendations

Frequently bought together

Total price: $69.12

Add all three to Cart

Add all three to List

- ☑ **This item:** Giordano Bruno and the Hermetic Tradition by Frances A. Yates Paperback $28.50
- ☑ The Art of Memory by Frances Yates Paperback $25.62
- ☑ De Umbris Idearum: On the Shadows of Ideas by Giordano Bruno Paperback $15.00

Beer and Diapers

?

Not many Wikipedia pages relevant to programming in R have a Lore section.

What is association rule learning?

An rule $X \Rightarrow Y$ suggests that any transaction containing all the items in a set X is "likely" to contain all the items in a set Y.

A good prototype transaction is the shopping cart full of goods a consumer purchasers in one visit to a store. A speculative example rule might be $\{\text{cereal}\} \Rightarrow \{\text{milk}\}$.

We'll use the apriori algorithm to find the association rules in some point-of-sale data relative to a floor on what is "likely" enough. We're essentially **mining for coincidences.** The point of the algorithm is that it helps us handle the difficulties of a thick dataset - our point-of-sale database describes purchases of many different items.

Example Point-of-Sale Data

We use an online E-commerce dataset from the UCI Machine Learning Repository.

https://archive.ics.uci.edu/ml/datasets/online+retail

It's entries look like this...

	InvoiceNo	Description	CustomerID
1	536365	WHITE HANGING HEART T-LIGHT HOLDER	17850
2	536365	WHITE METAL LANTERN	17850
3	536365	CREAM CUPID HEARTS COAT HANGER	17850
4	536365	KNITTED UNION FLAG HOT WATER BOTTLE	17850
5	536365	RED WOOLLY HOTTIE WHITE HEART.	17850

Differentiator: Thick Datasets

Point-of-sale data is often very thick - each data point (transaction) has many properties (possible items). A natural idea is to model an item on a feature made from 0s and 1s depending on if any given item is included.

This turns out to be tough because...

- There are too many variables.
- Some items may appear only in a small number of transactions (the rows are mostly 0).

Association rules study the relationship of each item to the others in an efficient way.

Differentiator: Categorical Variables

Association rules are a tool for working with categorical data, as input and output.

Continuous variables (age, amounts of money, etc.) must be discretized for use with association rules...

- The approach to discretization can affect the outcome.
- The arules library contains standard tools for discretizing.

The categories can be both inputs and output themselves, i.e. $\{\text{beer} \Rightarrow \text{diapers}\}$ is just as good as $\{\text{diapers} \Rightarrow \text{beer}\}$.

Association Rule Tutorials

There are a number of helpful tutorials out there, including on blogs like...

- Michael Hahsler
- R-bloggers
- Data Science Plus

Will discuss later some important facts about how to format the input that these blogs mostly ignore.

The key algorithm in the arules library, apriori, is applied to compute all the sufficiently unusual association rules in a dataset.

Rule Metrics 1

There are many possible rules that we could exmamine. We need to quantify rule quality and set some minimum floors for how "good" a rule should be.

One obvious, dataset-wide property any given item has is how often it occurs in transactions. For a set of transactions T and items X, we'll define the support of X to be

$$\text{supp}(X) = \frac{|\{t \in T : X \subseteq t\}|}{|T|}$$

and use these numbers in different forms to study our rules. The support of X is **the proportion of transactions that contain all the elements of X.**

Rule Metrics 2

Confidence ("prediction reliability"):

$$\mathrm{conf}(X \Rightarrow Y) = \frac{\mathrm{supp}(X \cup Y)}{\mathrm{supp}(X)}$$

Lift ("statistical anomalousness"):

$$\mathrm{lift}(X \Rightarrow Y) = \frac{\mathrm{supp}(X \cap Y)}{\mathrm{supp}(X) \times \mathrm{supp}(Y)}$$

If bread and eggs occur in every single transaction, then

$$\{\mathrm{bread}\} \Rightarrow \{\mathrm{eggs}\}$$

will have maximum confidence but poor lift.

Rule Metrics 3

$$T = \{\{\text{cereal}, \text{milk}\}, \{\text{eggs}, \text{milk}\}, \{\text{watermelon}\}\}$$

$$\text{supp}(\{\text{cereal}\}) = \frac{1}{3}$$

$$\text{supp}(\{\text{milk}\}) = \frac{2}{3}$$

$$\text{supp}(\{\text{watermelon}\}) = \frac{1}{3}$$

$$\text{conf}(\{\text{cereal}\} \Rightarrow \{\text{milk}\}) = \frac{1}{1}$$

$$\text{lift}(\{\text{cereal}\} \Rightarrow \{\text{milk}\}) = \frac{1/3}{1/3 \times 2/3} = \frac{3}{2}$$

Input Format vs. Tabular Data

The most basic input to the apriori algorithm function is a **list of atomic vectors**, each item in the list a vector representing a transaction and each entry in the vector a string describing an item. The vectors should no repeated entries.

```
transactions = list(
  c('milk','cereal'),
  c('milk','eggs'),
  c('watermelon')
)
```

An Issue Sometimes Hidden 1

Data

I'm using the AdultUCI dataset that comes bundled with the `arules` package.

```
> data("Groceries")
```

Write dataframe to a csv file using write.csv()

```
write.csv(df_itemList,"ItemList.csv", qoute = FALSE, row.names = TRUE)
```

Using the read.transactions() functions, we can read the file ItemList.csv and convert it to a transaction format

```
txn = read.transactions(file="ItemList.csv", rm.duplicates= TRUE, format="ba
```

An Issue Sometimes Hidden 2

A number of tutorials skipped data wrangling entirely by using prepared data.

In others, important wrangling is hidden across three steps

- Creating a column of item strings collapse with commas
- Exporting to a text file
- Re-Importing using a specialized arules function

Data wrangling for association rules poses some unique challenges worth noting.

Computing Rules

```
#somewhat nonstandard use of aggregate
formatted <- aggregate(
  df[c('Description')],
  by=df[c('InvoiceNo')],
  unique
)
# formatted$Description will be a list of
# atomic vectors with no repeated elements

#compute rules
rules <- apriori(
  formatted$Description,
  parameter = list (
    supp = 0.005,
    conf = 0.9,
    maxlen=3)
)
```

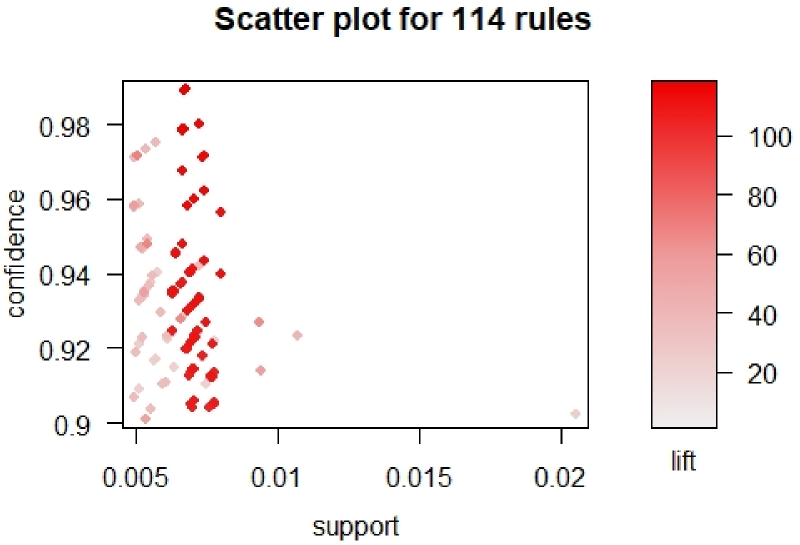
Computing Rules

```
> inspect(head(sort(rules, by = "support"),15))
      lhs                                     rhs                                 support confidence      lift count
[1]  {PINK REGENCY TEACUP AND SAUCER,
      ROSES REGENCY TEACUP AND SAUCER }   => {GREEN REGENCY TEACUP AND SAUCER}    0.020509037  0.9025974  23.71067   278
[2]  {WOODEN HEART CHRISTMAS SCANDINAVIAN,
      WOODEN TREE CHRISTMAS SCANDINAVIAN} => {WOODEN STAR CHRISTMAS SCANDINAVIAN} 0.010697160  0.9235669  39.61693   145
[3]  {REGENCY TEA PLATE GREEN ,
      REGENCY TEA PLATE PINK}             => {REGENCY TEA PLATE ROSES }           0.009369236  0.9136691  53.61378   127
[4]  {REGENCY TEA PLATE PINK,
      REGENCY TEA PLATE ROSES }           => {REGENCY TEA PLATE GREEN }           0.009369236  0.9270073  65.10665   127
[5]  {HERB MARKER THYME}                  => {HERB MARKER ROSEMARY}               0.008041313  0.9561404 111.72830   109
[6]  {HERB MARKER ROSEMARY}               => {HERB MARKER THYME}                  0.008041313  0.9396552 111.72830   109
[7]  {HERB MARKER ROSEMARY}               => {HERB MARKER BASIL}                  0.007819993  0.9137931 104.97005   106
[8]  {GREEN REGENCY TEACUP AND SAUCER,
      REGENCY TEA PLATE ROSES }           => {ROSES REGENCY TEACUP AND SAUCER }   0.007819993  0.9217391  21.99679   106
[9]  {HERB MARKER PARSLEY}                => {HERB MARKER ROSEMARY}               0.007746219  0.9210526 107.62818   105
[10] {HERB MARKER ROSEMARY}               => {HERB MARKER PARSLEY}                0.007746219  0.9051724 107.62818   105
[11] {HERB MARKER MINT}                   => {HERB MARKER BASIL}                  0.007746219  0.9051724 103.97976   105
[12] {HERB MARKER MINT}                   => {HERB MARKER ROSEMARY}               0.007746219  0.9051724 105.77252   105
[13] {HERB MARKER ROSEMARY}               => {HERB MARKER MINT}                   0.007746219  0.9051724 105.77252   105
[14] {HERB MARKER THYME}                  => {HERB MARKER PARSLEY}                0.007672446  0.9122807 108.47338   104
[15] {HERB MARKER PARSLEY}                => {HERB MARKER THYME}                  0.007672446  0.9122807 108.47338   104
```

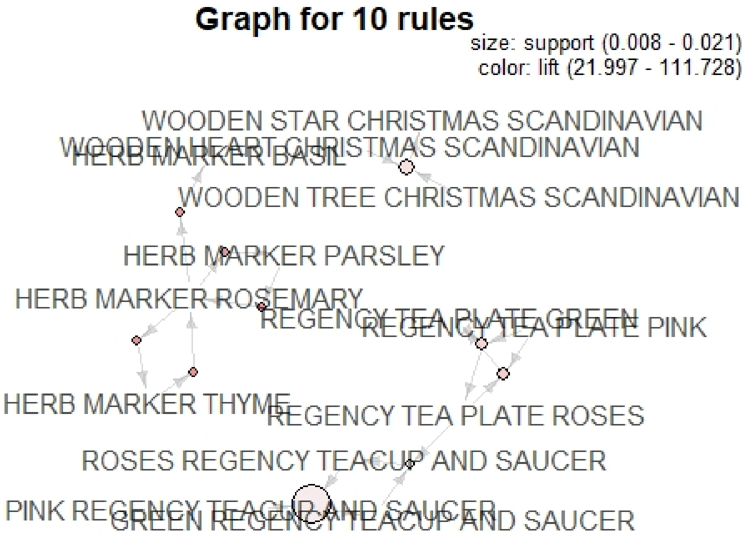
Rule Basics

```
> # restrictive parameters => few rules
> length(rules)
[1] 114
> #rules have their own class
> class(rules)
[1] "rules"
attr(,"package")
[1] "arules"
> # the apriori function actually coerced our data
> # to a a type called a transaction
> class(as(formatted$Description,'transactions'))
[1] "transactions"
attr(,"package")
[1] "arules"
```

vizRules Scatter



vizRules Web



vizRules Code

```
#visualizations
library(arulesViz)

#the birds eye view
plot(rules)

#filter down to the high support rules
inspect(head(sort(rules, by = "support"),10))

#make a web graph of the high support rules
plot(
  head(sort(rules, by = "support"), 10),
  method="graph",
  control=list(type="items")
)
```

A Surprising Rule

Inevitably, a lot of the rules are pretty obvious. One always hopes, however, that there will be a few which are not...

$$\{\text{JAM MAKING SET WITH JARS}, \text{SUKI SHOULDER BAG}\}$$

$$\Rightarrow \{\text{DOTCOM POSTAGE}\}$$

Unfortunately, making a computer judge what is interesting or not and to which humans is a harder problem.

vizRules Big Clusters