УДК 004

Исследование систем генерации ассоциативных правил

Потылицын Андрей Олегович
Приамурский государственный университет имени Шолом-Алейхема
Студент

Аннотация
В данной работе будут рассмотрены программы с целью решения ассоциативных правил и приведен пример их построения при помощи программы RStudio.
Ключевые слова: ассоциативные правила, RStudio.

Research of systems of generation of associative rules

Potylitsyn Andrey Olegovich
Sholom-Aleichem Priamursky State University
Student

Abstract
In this paper, we will consider programs for solving associative rules and give an example of their construction using the RStudio program.
Keywords: association rule, RStudio.

В настоящее время заинтересованность к методам «обнаружения знаний» в базах сведений постоянно повышается. Современные базы сведений обладают весьма крупными размерами, достигают гигабайт и терабайт, и имеют тенденцию к последующему повышению, что обусловило потребность разработки результативных масштабируемых алгоритмов, позволяющих решать проблемы в короткие сроки. Одним из самых популярных способов выявления знаний стали алгоритмы ассоциативных правил нахождения различных видов логики.

Ассоциации - обнаружение закономерностей между связанными событиями. Образцом данного стандарта является правило, указывающее, что событие X следует за событием Y. Подобные правила называются ассоциативными.

О построении ассоциативных правил писали многие авторы: А.В. Бондаренко и А.С. Гудков в статье «Интерактивный анализ ассоциативных правил в базах данных» Рассмотрели способы анализа правил в базах данных. Предложили способ интерактивного просмотра правил на сводной таблице, где правила отображаются по выбранным измерениям исходной таблицы. Разработали алгоритм обработки запросов для данного способа просмотра, основанный на хранении данных в виде префиксного дерева и выполнении запросов с помощью перестроек префиксного

дерева[1]. А.В. Бакулев и М.А. Бакулева в статье «Построение ассоциативных правил на основе дифференцирования графовой модели анализируемой выборки» Рассмотрели методы интеллектуального анализа сведений, основанные на алгоритмах построения ассоциативных правил. Предложили подход к построению ассоциативных правил на основе построения и дифференцирования модельного графа анализируемой выборки[2]. В.А. Биллиг, О.В. Иванова и Н.А. Царегородцев в статье «Построение ассоциативных правил в задаче медицинской диагностики» рассмотрели новый эффективный алгоритм построения ассоциативных правил AprioriScale. Алгоритм применили к решению конкретной задачи диагностики в медицине. Построили реализация этого алгоритма на языке программирования C#[3]. С.В. Белим, Т.Б. Смирнова и А.Н. Мироненко в статье «Применение метода построения ассоциативных правил к анализу деятельности общественных организаций» предложили математический метод анализа показателей деятельности общественных организаций, основанный на построении ассоциативных правил[4]. И.А. Олянич в статье «Сравнение алгоритмов построения ассоциативных правил на основе набора сведений покупательских транзакций» рассмотрел алгоритмы построения ассоциативных правил Apriori и Eclat, с помощью которых произвел анализ набора сведений, содержащего в себе информацию о продуктовых получениях пользователей крупнейшего ритейлера в США Walmart[5]. О.Е. Диев и В.И. Мунерман в статье «Анализ решения задачи вывода ассоциативных правил в технологии in-database» рассмотрели задачу вывода ассоциативных правил, возвозможность распараллеливания данного процесса в технологии in-database средствами одной из наиболее эффективных СУБД PostgreSQL[6]. Так же данной проблемой интересовались и зарубежные авторы [7-9].

Цель исследования - сгенерировать ассоциативные правила с помощью выбранной компьютерной программе.

RStudio – среда разработки программ на языке R с графическим интерфейсом.

Рассмотрим пример рыночной корзины, что чаще всего приобретают покупатели после того, как купят один товар.

С целью того, чтобы воспользоваться средой разработки программы, нужно установить библиотеку (пакет arules):

```
install.packages('arules')
```

Затем выбираем пакет (рис. 1).

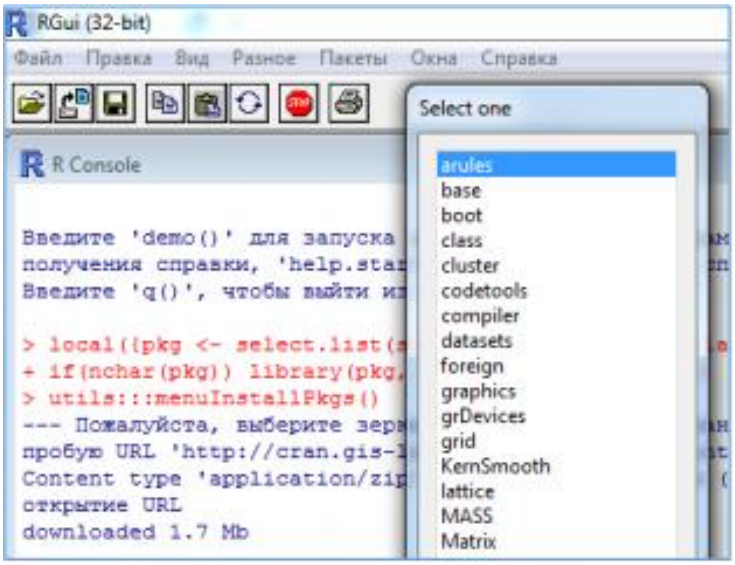

Рисунок 1 – подбор пакета

Обнаружим документ Groceries.rda, затем сделаем в пакете R следующие шаги:

Подключим требуемые библиотеки:

```
library(arules)
library(datasets)
```

Обнаружим данные Groceries : `data(Groceries)`

Получим общие сведения о сведений: `Groceriessummary(Groceries)` (рис. 2)

```
> summary(Groceries)
transactions as itemMatrix in sparse format with
 9835 rows (elements/itemsets/transactions) and
 169 columns (items) and a density of 0.026

most frequent items:
      whole milk other vegetables       rolls/buns             soda           yogurt          (Other)
            2513             1903             1809             1715             1372            34055

element (itemset/transaction) length distribution:
sizes
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16   17   18   19   20   21   22   23   24   26   27   28   29   32
2159 1643 1299 1005  855  645  545  438  350  246  182  117   78   77   55   46   29   14   14    9   11    4    6    1    1    1    1    3    1

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      1       2       3       4       6      32
```

Рисунок 2 - Сводные свойства сведений с файла Groceries.rda

Затем осуществим следующее: выстроим частотную диаграмму с целью двадцати продуктов, пользующихся максимальным спросом

```
itemFrequencyPlot(Groceries, topN=20, type="absolute")
```

Получим следующую картинку (Рис. 3):

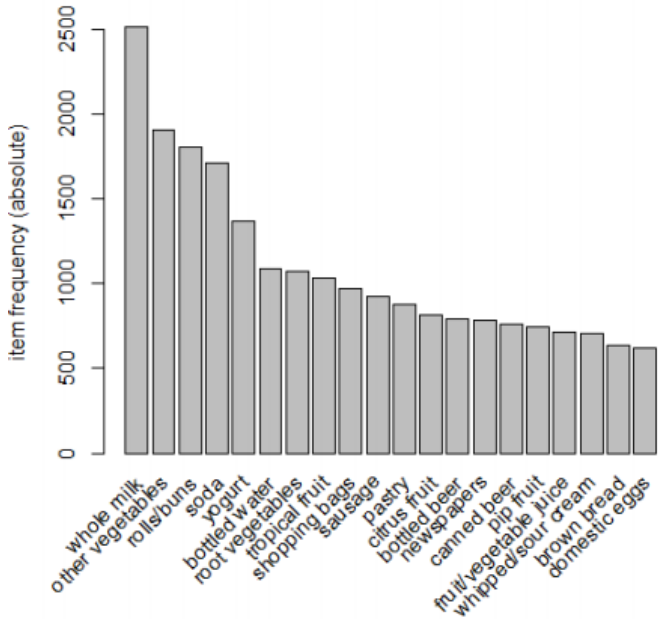

Рисунок 3 - Частотная диаграмма с целью двадцати продуктов, пользующихся максимальным спросом (согласно данным из файла Groceries.rda)

Видно, то что в главную двадцатку входят данные продукты: цельное молоко, овощи (не корнеплоды), булочки/плюшки, газированная вода, иогурт, бутилированная (негазированная) вода, корнеплоды, тропические плоды, сумки с целью покупок, колбаса, хлебобулочные изделия, цитрусовые, пиво в бутылках, печатные издания, баночное пиво, плоды-костянки, плодовый сок, взбитые сливки и сметана, чёрный хлеб, домашние яйца.

В функции `itemFrequencyPlot` параметр `type` отвечает за тип диаграммы: `absolute` – с целью заключения абсолютных значений частот; `relative` – с целью относительных.

При создании диаграммы возможно выводить не фиксированное число столбиков (товаров), а только те из них, условная частота появления которых в чеках никак не ниже установленного значения. С целью этого пользуется параметр `support`. К примеру, `support=0.1`.

С целью визуализации частот встречаемости товаров в чеках удобно применять таким образом именуемую «разреженную матрицу» `(sparsematrix)`.

Выведем сведения начальных двухсот чеков: `image(Groceries[1:200],axes="TRUE")`

Итог визуализации разреженной матрицы с целью первых двести чеков показан на Рис. 4. Имеет смысл отбирать чеки с целью визуализации сведений случайным образом. Выведем сведения двести случайным способом выбранных чеков: `image(Groceries, 200, axes="TRUE")`

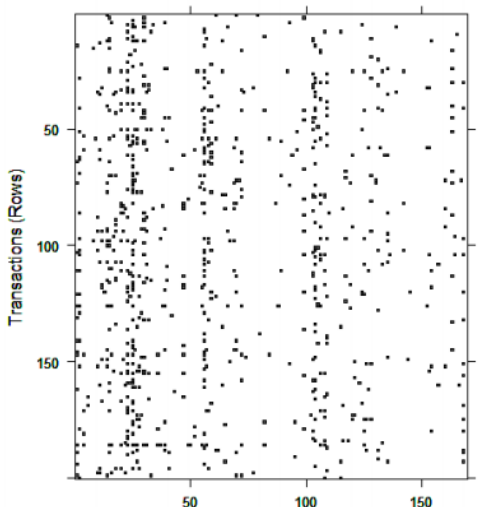

Рисунок 4 - Итог визуализации «разреженной матрицы» с целью первых двухста чеков

Далее «Закажем» пакету отбор ассоциативных правил. Будем испоьзовать Априорный алгоритм. Учитывать будем только те продукты, какие попадаются с частотой не ниже одного процента так же эти ассоциативные правила, у которых `confidence` не ниже пятидесяти процента.

```
myrules = apriori(data=Groceries,
parameter=list(support=0.001,confidence=0.9,
minlen=1))
```

Распечатаем начальные пять ассоциативных правил: `inspect(rules[1:5])`

Получим такой итог (Рис. 5):

```
> inspect(rules[1:5])
  lhs                     rhs                support confidence      lift
1 {liquor,
   red/blush wine}     => {bottled beer} 0.001931876  0.9047619 11.235269
2 {curd,
   cereals}            => {whole milk}   0.001016777  0.9090909  3.557863
3 {yogurt,
   cereals}            => {whole milk}   0.001728521  0.8095238  3.168192
4 {butter,
   jam}                => {whole milk}   0.001016777  0.8333333  3.261374
5 {soups,
   bottled beer}       => {whole milk}   0.001118454  0.9166667  3.587512
```

Рисунок 5 - Начальные пять ассоциативных правил (с целью сведений из файла Groceries.rda)

Тут `lhs` (англ.: left-handside) – левая доля (ассоциативного правила), `rhs` (англ.: right-handside) -правая доля (ассоциативного правила)

Видим, то что – со значительной вероятностью (90%) получение спиртного (ликёра или красного вина) тянет получение бутылочного пива; с вероятностью в 91% получеие творога так же хлопьев влечёт получение цельного молока; с вероятностью в восемьдесят три процента получение масла и джема влечёт получение цельного молока и т.д.

«Закажем» пакету R сводные сведения о комплекте полученных ассоциативных правил:

```
summary(myrules)
```

Получим итог, показанный на Рис. 6.

```
>   summary(myrules)
set of 129 rules

rule length distribution (lhs + rhs):sizes
 3   4   5   6
10 57 56   6
```

Рисунок 6 – Сводные сведения о комплекте ассоциативных правил, полученном Априорным методом

Подобным образом, согласно установленным условиям `(support=0.001,confidence=0.9, minlen=1)` приобретено всего сто двадцать девять правил. Длина этих правил (т.е. общее количество продуктов в двух составляющих правила) колеблется от трех до шести, а именно: три правила имеют длину десять, четыре – длину пятьдесят семь и т.д.

Перед заключением списка ассоциативных правил имеет смысл сперва просортировать их согласно тому или иному показателю: Отсортируем все правила в порядке убывания лифта:

```
myrules=sort(myrules, by="lift")
inspect(myrules[1:5])
```

Показатель lift демонстрирует, во сколько раз получение набора X повышает вероятность приобретения набора Y (по отношению к априорной вероятности приобретения набора Y).

Формула с целью расчета: $lift\ (X \rightarrow Y) = \frac{confidence\ (X \rightarrow Y)}{support\ (Y)}$

Полученные итоги показаны на Рис. 7. Сравним их с итогами, приведёнными на Рис. 5.

```
> inspect(myrules[1:5])
  lhs                     rhs                  support confidence      lift
1 {liquor,
   red/blush wine}     => {bottled beer}     0.001931876  0.9047619 11.235269
2 {citrus fruit,
   other vegetables,
   soda,
   fruit/vegetable juice} => {root vegetables} 0.001016777  0.9090909  8.340400
3 {tropical fruit,
   other vegetables,
   whole milk,
   yogurt,
   oil}                => {root vegetables} 0.001016777  0.9090909  8.340400
4 {root vegetables,
   butter,
   cream cheese }      => {yogurt}          0.001016777  0.9090909  6.516698
5 {tropical fruit,
   whole milk,
   butter,
```

Рисунок 7 - Первоначальные пять ассоциативных правил (с целью сведений из файла Groceries.rda) после сортировки по параметру lift

Как видим, на первом месте опять находится правило: {ликёр, красное вино} -- > {бутылочное пиво}, в то время как Затем следуют правила с большим значением параметра `lift`, чем в неотсортированном списке.

В некоторых случаях бывает нужно подобрать из всех полученных ассоциативных правил те, которые содержат определённый продукт. С целью этого используется функция subset:

Найдём все ассоциативные правила, где есть молоко: `milkrules<-subset(myrules, items %in% "wholemilk")`

Отсортируем список правил (рис. 8), включающих молоко, согласно убыванию лифта и распечатаем первоначальные пять правил: `inspect(sort(milkrules, by="lift")[1:5])`

```
  lhs                     rhs                    support     confidence     lift
1 {tropical fruit,
   other vegetables,
   whole milk,
   yogurt,
   oil}                => {root vegetables}  0.001016777  0.9090909 8.340400
2 {tropical fruit,
   whole milk,
   butter,
   sliced cheese}      => {yogurt}           0.001016777  0.9090909 6.516698
3 {tropical fruit,
   grapes,
   whole milk,
   yogurt}             => {other vegetables} 0.001016777  1.0000000 5.168156
4 {ham,
   tropical fruit,
   pip fruit,
   whole milk}         => {other vegetables} 0.001118454  1.0000000 5.168156
5 {whole milk,
   rolls/buns,
   soda,
   newspapers}         => {other vegetables} 0.001016777  1.0000000 5.168156
```

Рисунок 8 - Первоначальные пять ассоциативных правил из числа правил, содержащих «wholemilk», отсортированные согласно lift

Запишем отысканное множество ассоциативных правил в файл: `write(dairy_rules, file = "dairy_rules.txt", sep = ",", quote = TRUE, row.names = FALSE)`

Файл `dairy_rules.txt` будет записан в рабочий директорий пакета R. В качестве выходного файла возможно также указать файл с расширением `csv`. Такой файл возможно будет открыть в приложении EXCEL.

Подведя итоги, можно сделать следующие выводы о рассматриваемых инструментариях:

1. Выбор инструментария зависит он навыка полученных знаний в средах программирования, пакетах разнообразных алгоритмов, таких как: Apriori, DataMining и т.д. ;

2. С целью того, чтобы воспользоваться каким – либо видом построения ассоциативных правил нужно учитывать тот факт, что большинство программных инструментариев составлено на иностранном языке, поэтому нужно очень внимательно изучать данную систему;

3. Каждый инструментарий имеет ряд плюсов и минусов, поэтому определенно сказать, какой лучше – невозвозможно. Это сугубо личный выбор каждого пользователя.

Сами по себе ассоциативные правила имеют очень большую практическую значимость.

В большинстве случаев ассоциативные правила применяются в маркетинге, но так же им возможно воспользоваться и в учебном процессе.

Допустим, выбрать предмет, который наиболее популярен среди студентов, посмотреть, почему он так привлекает внимание и затем применить этот критерий к своему предмету.

Возможность доступа к разнообразным источникам информации дает возможность беспрепятственно обучиться генерации ассоциативных правил и применять их на практике.

Крупные данные – это будущее. Теперь допустимо точно сказать – это будущее уже наступило, у нас есть не только инфраструктура и готовое ПО, но и возможность скомбинировать продукты, предложенные рынком, чтобы получить бизнес-итог, реальную пользу с целью дела.

Библиографический список

1. Бондаренко А.В., Гудков А.С. Интерактивный анализ ассоциативных правил в базах данных // Вестник компьютерных и информационных технологий. 2006. № 10 (28). С. 42-45.
2. Бакулев А.В., Бакулева М.А. Построение ассоциативных правил на основе дифференцирования графовой модели анализируемой выборки // Вестник Рязанского государственного радиотехнического университета. 2013. № 46-2. С. 86-88.
3. Биллиг В.А., Иванова О.В., Царегородцев Н.А. Построение ассоциативных правил в задаче медицинской диагностики // Программные продукты и системы. 2016. № 2. С. 146-157.
4. Белим С.В., Смирнова Т.Б., Мироненко А.Н. Применение метода построения ассоциативных правил к анализу деятельности общественных организаций // Математические структуры и моделирование. 2017. № 2 (42). С. 49-58.
5. Олянич И.А. Сравнение алгоритмов построения ассоциативных правил на основе набора сведений покупательских транзакций // Известия Самарского научного центра Российской академии наук. 2018. Т. 20. № 6-2. С. 379-382.
6. Диев О.Е., Мунерман В.И. Анализ решения задачи вывода ассоциативных правил в технологии in-database // Системы компьютерной математики и их приложения. 2018. № 19. С. 134-139.
7. Piatetsky-Shapiro G. Data mining and knowledge discovery – 1996 to 2005: overcoming the hype and moving from “university” to “business” and “analytics”. Data Mining and Knowledge Discovery Journ., 2007, pp. 99–105.
8. Agrawal R. and Srikant R. Fast algorithms for mining association rules in large databases. Proc. 20th Intern. Conf. VLDB, 1994, pp. 487– 499.
9. Christian Borglet’s web pages. URL: http://www.borgelt.net/apriori.html Ihaka R., Gentleman R. R: A language for data analysis and graphics. Journ. of Computational and Graphical Statistics, 1996, vol. 5, no. 3, pp. 299–314.