УДК 336.74:004.738.5

# ИССЛЕДОВАНИЕ АССОЦИАТИВНЫХ ПРАВИЛ В ИНТЕЛЛЕКТУАЛЬНОМ АНАЛИЗЕ ДАННЫХ

***А.В. Цехан***
*Магистрант кафедры математического и информационного обеспечения экономических систем*
*УО ГрГУ.им. Я.Купалы*

***Н.В. Марковская***
*Доцент кафедры математического и информационного обеспечения экономических систем*
*УО ГрГУ.им. Я.Купалы*

*УО Гродненский государственный университет имени Янки Купалы*
*E-mail: tsekhan96@gmail.com, n.markovskaya@grsu.by*

***А.В. Цехан***
*Окончила Гродненский государственный университет имени Янки Купалы. Магистрант кафедры математического и информационного обеспечения экономических систем УО ГрГУ им. Я. Купалы.*

***Н.В. Марковская***
*Доцент кафедры математического и информационного обеспечения экономических систем УО ГрГУ им.Я.Купалы, кандидат физико-математических наук, доцент.*

**Аннотация.** Рассмотрена задача построения моделей на основе ассоциативных правил. Проанализирован процесс поиска ассоциативных правил. Рассмотрен алгоритм поиска ассоциативных правил – Apriori.

**Ключевые слова:** ассоциативное правило, различные виды ассоциативных правил, обычные ассоциативные правила, алгоритм Apriori, алгоритм поиска ассоциативных правил.

*Введение.* В последнее время неуклонно растет интерес к методам «обнаружения знаний в базах данных» (knowledge discovery in databases). Объемы современных баз данных, которые весьма внушительны, вызвали устойчивый спрос на новые масштабируемые алгоритмы анализа данных. Одним из популярных методов обнаружения знаний стали алгоритмы поиска ассоциативных правил.

Ассоциативные правила позволяют находить закономерности между связанными событиями. Примером такого правила, служит утверждение, что покупатель, приобретающий «Хлеб», приобретет и «Молоко» с вероятностью 75%. Первый алгоритм поиска ассоциативных правил, называвшийся AIS [4] был разработан в 1993 году сотрудниками исследовательского центра IBM Almaden. С этой пионерской работы возрос интерес к ассоциативным правилам; на середину 90-х годов прошлого века пришелся пик исследовательских работ в этой области, и с тех пор каждый год появлялось по несколько алгоритмов.

Впервые это задача была предложена поиска ассоциативных правил для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее еще называют анализом рыночной корзины (market basket analysis).

Ассоциативные правила позволяют находить закономерности между связанными событиями. Примером такого правила служит утверждение, что покупатель, приобретающий

«Хлеб», приобретет и «Молоко» с вероятностью 75%. Впервые эта задача была предложена для поиска ассоциативных правил для нахождения типичных шаблонов покупок, совершаемых в супермаркетах, поэтому иногда ее еще называют анализом рыночной корзины (market basket analysis).

Транзакция – это множество событий, произошедших одновременно. Пусть имеется база данных, состоящая из покупательских транзакций. Каждая транзакция – это набор товаров, купленных покупателем за один визит. Такую транзакцию еще называют рыночной корзиной. После определения понятия транзакция можно перейти к определению ассоциативного правила. Пусть имеется список транзакций. Необходимо найти закономерности между этими событиями. Как в условии, так и следствии правила должны находиться элементы транзакций.

Пусть I={i1,i2,i3,…,in}I={i1,i2,i3,…,in} - множество (набор) товаров, называемых элементами. Пусть D - множество транзакций, где каждая транзакция T – это набор элементов из I, T⊆ I. Каждая транзакция представляет собой бинарный вектор, где t[k]=1, если ik элемент присутствует в транзакции, иначе t[k]=0. Мы говорим, что транзакция T содержит X, некоторый набор элементов из I, если X⊂⊂T. Ассоциативным правилом называется импликация X⇒Y, где X⊂I, Y⊂I и X∩Y=∅. Правило X⇒Y имеет поддержку s (support), если s% транзакций из D, содержат X∪Y, supp(X⇒Y) = supp(X∪Y). Достоверность правила показывает какова вероятность того, что из X следует Y. Правило X⇒Y справедливо с достоверностью (confidence) c, если c% транзакций из D, содержащих X, также содержат Y, conf(X⇒Y) = supp(X∪Y)/supp(X ).

Лифт – это отношение частоты появления условия в транзакциях, которые также содержат и следствие, к частоте появления следствия в целом: lift(XÞY) = conf(XÞY)/supp(Y). Значения лифта, большие единицы, показывают, что условие появляется более часто в транзакциях, содержащих и следствие, чем в остальных.

Покажем на конкретном примере:

75% транзакций, содержащих хлеб, также содержат молоко. 3% от общего числа всех транзакций содержат оба товара сразу.

75% – это достоверность (confidence) правила, 3% это поддержка (support) или.Если Хлеб, то Молоко с вероятностью 75%.

Другими словами, целью анализа является установление следующих зависимостей: если в транзакции встретился некоторый набор элементов X, то на основании этого можно сделать вывод о том, что другой набор элементовY также должен появиться в этой транзакции. Установление таких зависимостей дает нам возможность находить очень простые и интуитивно понятные правила.

Алгоритмы поиска ассоциативных правил предназначены для нахождения всех правил XÞY, причем поддержка и достоверность этих правил должны быть выше некоторых заранее определенных порогов, называемых, соответственно минимальной поддержкой (minsupport) и минимальной достоверностью (minconfidence). Аналогично, поддержка и достоверность ограничиваются сверху порогами максимальной поддержки (maxsupport) и максимальной достоверности (maxconfidence). В результате получаются два окна, в которые должны попасть поддержка и достоверность правила, чтобы оно было предъявлено аналитику [1].

Значения для параметров минимальная (максимальная) поддержка и минимальная (максимальная) достоверность выбираются таким образом, чтобы ограничить количество найденных правил. Если поддержка имеет большое значение, то алгоритмы будут находить правила, хорошо известные аналитикам или настолько очевидные, что нет никакого смысла проводить такой анализ. С другой стороны, низкое значение поддержки ведет к генерации огромного количества правил, что, конечно, требует существенных вычислительных ресур-

сов. Большинство интересных правил находится именно при низком значении порога поддержки, хотя слишком низкое значение поддержки ведет к генерации статистически необоснованных правил. Ассоциативные правила с высокой поддержкой могут применяться для формализации хорошо известных правил, например, в автоматизированных системах для управления процессами или персоналом. Надо отметить, что понятия «высокая» и «низкая» поддержка или достоверность очень сильно зависят от предметной области.Например, в торговле 1% вероятности совместного приобретения хлеба и молока не значит ничего, в то время как вероятность в 1% отказа двигателя самолета совершенно неприемлема, и такое правило становится чрезвычайно важным.

Поиск ассоциативных правил совсем не тривиальная задача, как может показаться на первый взгляд. Одна из проблем – алгоритмическая сложность при нахождении часто встречающих наборов элементов, т.к. с ростом числа элементов экспоненциально растет число потенциальных наборов элементов.

Обычные ассоциативные правила – это правила, в которых как в условии, так и в следствии присутствуют только элементы транзакций и при вычислении которых используется только информация о том, присутствует ли элемент в транзакции или нет. Фактически все приведенные выше примеры относятся к обычным ассоциативным правилам [2].

Для поиска обычных ассоциативных правил в программе служит обработчик «Ассоциативные правила».

**Алгоритм Apriori**

Один из первых алгоритмов, эффективно решающих подобный класс задач, – это алгоритм APriori [5]. Кроме этого алгоритма в последнее время был разработан ряд других алгоритмов: DHP[6], Partition[8], DIC[7] и другие.

Значения для параметров минимальная поддержка и минимальная достоверность выбираются таким образом, чтобы ограничить количество найденных правил. Если поддержка имеет большое значение, то алгоритмы будут находить правила, хорошо известные аналитикам или настолько очевидные, что нет никакого смысла проводить такой анализ. С другой стороны, низкое значение поддержки ведет к генерации огромного количества правил, что, конечно, требует существенных вычислительных ресурсов. Тем не менее, большинство интересных правил находится именно при низком значении порога поддержки. Хотя слишком низкое значение поддержки ведет к генерации статистически необоснованных правил.

Поиск ассоциативных правил совсем не тривиальная задача, как может показаться на первый взгляд. Одна из проблем - алгоритмическая сложность при нахождении часто встречающих наборов элементов, т.к. с ростом числа элементов в I (| I |) экспоненциально растет число потенциальных наборов элементов.

На первом этапе происходит формирование одноэлементных кандидатов. Далее *алгоритм* подсчитывает *поддержку* одноэлементных наборов. Наборы с уровнем *поддержки* меньше установленного, то есть 3, отсекаются. В нашем примере это наборы e и f, которые имеют *поддержку*, равную 1. Оставшиеся наборы товаров считаются часто встречающимися одноэлементными наборами товаров: это наборы a, b, c, d.

Далее происходит формирование двухэлементных кандидатов, подсчет их *поддержки* и *отсечение* наборов с уровнем *поддержки*, меньшим 3. Оставшиеся двухэлементные наборы товаров, считающиеся часто встречающимися двухэлементными наборами ab, ac, bd, принимают участие в дальнейшей работе алгоритма.

Если смотреть на работу алгоритма прямолинейно, на последнем этапе *алгоритм* формирует трехэлементные наборы товаров: abc, abd, *bcd, acd*, подсчитывает их *поддержку* и отсекает наборы с уровнем *поддержки*, меньшим 3. Набор товаров abc может быть назван часто встречающимся.

Однако *алгоритм* Apriori уменьшает количество кандидатов, отсекая - априори - тех, которые заведомо не могут стать часто встречающимися, на основе информации об отсеченных кандидатах на предыдущих этапах работы алгоритма.

*Отсечение* кандидатов происходит на основе предположения о том, что у часто встречающегося набора товаров все подмножества должны быть часто встречающимися. Если в наборе находится *подмножество*, которое на предыдущем этапе было определено как нечасто встречающееся, этот кандидат уже не включается в формирование и подсчет кандидатов.

Так наборы товаров ad, bc, *cd* были отброшены как нечасто встречающиеся, *алгоритм* не рассматривал набор товаров abd, *bcd*, *acd*.

При рассмотрении этих наборов формирование трехэлементных кандидатов происходило бы по схеме, приведенной в верхнем пунктирном прямоугольнике. Поскольку *алгоритм* априори отбросил заведомо нечасто встречающиеся наборы, последний этап алгоритма сразу определил набор abc как единственный трехэлементный часто встречающийся набор (этап приведен в нижнем пунктирном прямоугольнике).

*Алгоритм* Apriori рассчитывает также *поддержку* наборов, которые не могут быть отсечены априори. Это так называемая негативная область (*negative border*), к ней принадлежат наборы-кандидаты, которые встречаются редко, их самих нельзя отнести к часто встречающимся, но все подмножества данных наборов являются часто встречающимися [3].

Для анализа возьмем базу Groceries содержащуюся в пакете arules

```
> data(Groceries)
>
> dim(Groceries)
[1] 9835  169
> Groceries [2,]
transactions in sparse format with
 1 transactions (rows) and
 169 items (columns)
```

*Рисунок 1*. База Groceries

Для анализа данных используем алгоритм Apriori с минимальной поддержкой 0.002, и значимостью 0.8

```
> rules <- apriori(Groceries, parameter=list(support=0.002, confidence=0.8))
Apriori

Parameter specification:
 confidence minval smax arem  aval originalSupport maxtime support minlen maxlen
        0.8    0.1    1 none FALSE            TRUE       5   0.002      1     10
 target   ext
  rules FALSE

Algorithmic control:
 filter tree heap memopt load sort verbose
    0.1 TRUE TRUE  FALSE TRUE    2    TRUE

Absolute minimum support count: 19

set item appearances ...[0 item(s)] done [0.00s].
set transactions ...[169 item(s), 9835 transaction(s)] done [0.00s].
sorting and recoding items ... [147 item(s)] done [0.00s].
creating transaction tree ... done [0.00s].
checking subsets of size 1 2 3 4 5 done [0.00s].
writing ... [11 rule(s)] done [0.00s].
creating S4 object  ... done [0.00s].
```

*Рисунок 2.* Алгоритм Apriori

В результате работы алгоритм вывел 11 правил

```
> summary(rules)
set of 11 rules

rule length distribution (lhs + rhs):sizes
3 4 5
3 3 5

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  3.000   3.500   4.000   4.182   5.000   5.000

summary of quality measures:
    support           confidence          lift            count
 Min.   :0.002034   Min.   :0.8000   Min.   :3.131   Min.   :20.00
 1st Qu.:0.002135   1st Qu.:0.8032   1st Qu.:3.202   1st Qu.:21.00
 Median :0.002339   Median :0.8214   Median :3.223   Median :23.00
 Mean   :0.002385   Mean   :0.8230   Mean   :3.504   Mean   :23.45
 3rd Qu.:0.002491   3rd Qu.:0.8284   3rd Qu.:3.723   3rd Qu.:24.50
 Max.   :0.003152   Max.   :0.8857   Max.   :4.578   Max.   :31.00

mining info:
      data ntransactions support confidence
 Groceries          9835   0.002        0.8
```

*Рисунок 3.* Rules

Изобразим список правил графически:

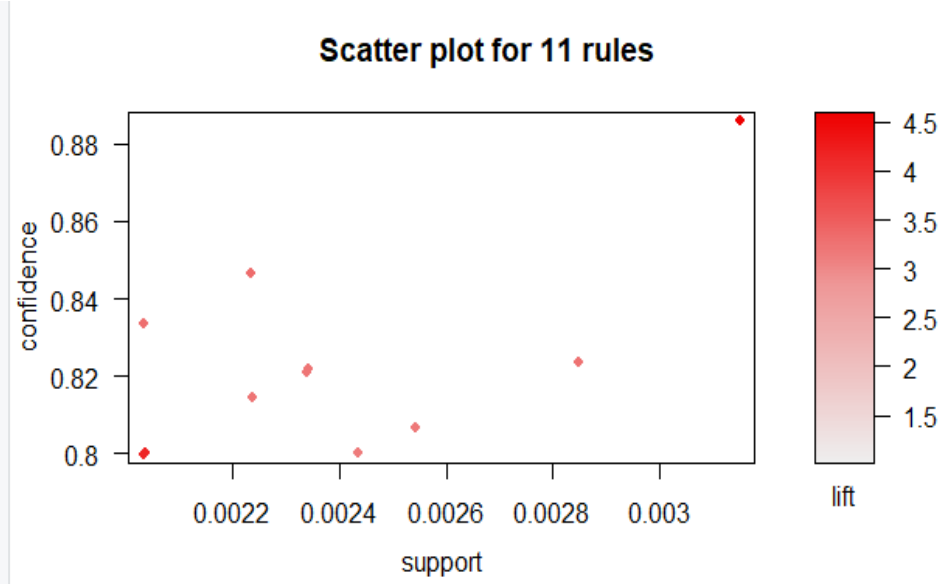


*Рисунок 4.* Графическое отображение списка правил

Проведем сортировку и выберем 3 правила

```
> options(digits = 3)
> inspect(rules[1:3])
    lhs                     rhs          support confidence lift count
[1] {tropical fruit,herbs} => {whole milk} 0.00234 0.821      3.21 23
[2] {herbs,rolls/buns}     => {whole milk} 0.00244 0.800      3.13 24
[3] {hamburger meat,curd}  => {whole milk} 0.00254 0.806      3.16 25
>
```

*Заключение.* Проведя анализ можно сделать вывод, что молоко приобретут в первую очередь те, кто покупает тропические фрукты. Затем следующие по очередности люди, которые покупают роллы и в последнюю очередь молоко приобретут люди вместе с гамбургером и творогом.

***Литература:***

[1] Алгоритмы выделения ассоциативных правил [Электронный ресурс] / Режим доступа : https://ranalytics.github.io/data-mining/054-Association-Rules-Algos.html. - Дата доступа : 21.01.2019.

[2] Ассоциативные правила [Электронный ресурс] / Режим доступа : https://studfiles.net/preview/2385102/page:32/. - Дата доступа : 21.01.2019.

[3] Методы поиска ассоциативных правил [Электронный ресурс] / Режим доступа : https://www.intuit.ru/studies/courses/6/6/lecture/186?page=3. - Дата доступа : 21.01.2019.

[4] R. Agrawal, T. Imielinski, A. Swami. 1993. Mining Associations between Sets of Items in Massive Databases. In Proc. of the 1993 ACM-SIGMOD Int'l Conf. on Management of Data, 207-216.

[5] R. Agrawal, R. Srikant. "Fast Discovery of Association Rules", In Proc. of the 20th International Conference on VLDB, Santiago, Chile, September 1994.

[6] R. Srikant, R. Agrawal. Mining quantitative association rules in large relational tables". In Proceedings of the ACM SIGMOD Conference on Management of Data, Montreal, Canada, June 1996.

[7] J.S. Park, M.-S. Chen, and S.Y. Philip, "An Effective HashBased Algorithm for Mining Association Rules", In Proc. ACM SIGMOD Int'l Conf. Management of Data, ACM Press, New York, 1995.

[8] J.S. Park, M.-S. Chen, and S.Y. Philip, "An Effective HashBased Algorithm for Mining Association Rules", In Proc. ACM SIGMOD Int'l Conf. Management of Data, ACM Press, New York, 1995.

## INVESTIGATION OF ASSOCIATIVE RULES IN INTELLECTUAL ANALYSIS OF DATA

***A.V. TSEKHAN***
*Master student of the Department of Mathematical and Information Support of Economic Systems, YKSUG*

***N.V. MARKOVSKAYA***
*Associate professor of the Department of Mathematical and Information Support of Economic Systems, YKSUG*

*Yanka Kupala State University of Grodno, Republic of Belarus*
*E-mail: tsekhan96@gmail.com, n.markovskaya@grsu.by*

**Abstract.** The task of building models based on associative rules is considered. Analyzed the process of finding associative rules. The algorithm of search for associative rules - Apriori.

**Keywords:** association rule, various types of association rules, common association rules, Apriori algorithm, association rule search algorithm.