**АФФИНИТИВНЫЙ АНАЛИЗ**

Аффинитивный анализ – это метод анализа данных, целью которого является выявление и количественная оценка взаимосвязей (ассоциаций) между событиями, происходящими совместно.

Цель аффинитивного анализа – исследование взаимной связи (ассоциаций) между событиями происходящими совместно и их количественная оценка. Результат выполнения аффинитивного анализа – набор ассоциативных правил.

Примерами приложения ассоциативных правил могут быть следующие задачи:

* выявление наборов товаров, которые в супермаркетах часто покупаются вместе или никогда не покупаются вместе;
* определение доли клиентов, положительно относящихся к нововведениям в их обслуживании;
* определение профиля посетителей веб-ресурса;
* определение доли случаев, в которых новое лекарство показывает опасный побочный эффект;
* выявление связи между параметрами оборудования и получаемыми качественными характеристиками продукта.

**Основные понятия аффинитивного анализа**

Базовым понятием аффинитивного анализа является *транзакция* – не которое множество событий, происходящих совместно.

В результате аффинитивного анализа мы устанавливаем закономерность следующего вида: "Если в транзакции встретился набор элементов A, то можно сделать вывод, что в этой же транзакции должен появиться набор элементов B". Установление таких закономерностей дает нам возможность находить ассоциативные правила.

Ассоциативное правило имеет вид: "Из события A следует событие B": A→B. Основными характеристиками ассоциативного правила являются поддержка и достоверность правила.

*Поддержка ассоциативного правила* – это число транзакций, которые содержат как условие, так и следствие. Например, для ассоциации A→B можно записать S(A→B)=P(A∩B)=(количество транзакций, содержащих А и В)/(общее число транзакций). Правило имеет поддержку s, если s% транзакций из всего набора содержат одновременно наборы элементов A и B или, другими словами, содержат оба товара.

*Достоверность правила* показывает, какова вероятность того, что из события A следует событие B. Достоверность ассоциативного правила A→B представляет собой меру точности правила и определяется как от ношение количества транзакций, содержащих и условие и следствие, к количеству транзакций, содержащих только условие: С(A→B)=Р(A|B)=P(A∩B)/P(А) (количество транзакций, содержащих А и В)/(количество транзакций, содержащих только А). Правило "Из A следует B" справедливо с достоверностью с, если c% транзакций из всего множества, содержащих набор элементов A, также содержат набор элементов B.

*Границы поддержки и достоверности ассоциативного правила*

При помощи использования алгоритмов поиска ассоциативных правил аналитик может получить все возможные правила вида "Из A следует B", с различными значениями поддержки и достоверности. Однако в большинстве случаев, количество правил необходимо ограничивать заранее установленными минимальными и максимальными значениями поддержки и достоверности. Если значение поддержки правила слишком велико, то в результате работы алгоритма будут найдены правила очевидные и хорошо известные. Слишком низкое значение поддержки приведет к нахождению очень большого количества правил, которые, возможно, будут в большей части необоснованными, но не известными и не очевидными для аналитика. Таким образом, необходимо определить такой интервал, который с одной стороны обеспечит нахождение неочевидных правил, а с другой – их обоснованность. Если уровень достоверности слишком мал, то ценность правила вызывает серьезные сомнения. Например, правило с достоверностью в 3% только условно можно назвать правилом. Аналитики могут от давать предпочтение правилам, которые имеют только высокую поддержку или только высокую достоверность, или оба этих показателя. Правила, для которых значения поддержки или достоверности превышают определенный, заданный пользователем порог, называются сильными правилами.

Методики поиска ассоциативных правил обнаруживают все ассоциации, которые удовлетворяют ограничениям на поддержку и достоверность, наложенным пользователем. Это приводит к необходимости рассматривать десятки и сотни тысяч ассоциаций, что делает невозможным обработку та кого количества данных вручную. Число правил желательно уменьшить таким образом, чтобы проанализировать только наиболее значимые из них. Значимость часто вычисляется как разность между поддержкой правила и в целом и произведением поддержки только условия и поддержки только следствия. Если условие и следствие независимы, то поддержка правила примерно соответствует произведению поддержек условия и следствия, то есть SAB≈SASB. Это значит, что хотя условие и следствие часто встречаются вместе, не менее часто они встречаются и по отдельности.

*Субъективные меры значимости ассоциативных правил*

*Лифт* вычисляется следующим образом: L(A→B)=C(A→B)/S(B). Лифт – это отношение частоты появления условия в транзакциях, которые также содержат и следствие, к частоте появления следствия в целом. Значения лифта, большие, чем 1, показывают, что условие чаще появляется в транзакциях, содержащих следствие, чем в остальных. Можно сказать, что лифт является обобщенной мерой связи двух предметных наборов: при значениях лифта больше 1 связь положительная, при 1 она отсутствует, при значениях меньше 1 – отрицательная.

*Левередж* вычисляется следующим образом: T(A→B)=S(A→B) S(A)S(B). Левередж – это разность между наблюдаемой частотой, с кото рой условие и следствие появляются совместно (т.е. с поддержкой ассоциации), и произведением частот появления (поддержек) условия и следствия по отдельности. Из ассоциаций с одинаковым лифтом ассоциация с большим левереджем представляет больший интерес, так как это говорит о том, что данное правило встречаются чаще. Улучшение вычисляется следующим образом: I(A→B)=S(A→B)/(S(A)S(B)). Улучшение показывает, полезнее ли правило случайного угадывания. Если I(A→B)>1, это значит, что вероятнее предсказать наличие набора В с помощью правила, чем угадать случайно.

**Методы поиска ассоциативных правил**

В процессе поиска ассоциативных правил может производится обнаружение всех ассоциаций, поддержка и достоверность для которых превышают заданный минимум. Простейший алгоритм поиска ассоциативных правил рассматривает все возможные комбинации условий и следствий, оценивает для них поддержку и достоверность, а затем исключает все ассоциации, которые не удовлетворяют заданным ограничениям. Число возможных ассоциаций с увеличением числа предметов растет экспоненциально. Поэтому в процессе генерации ассоциативных правил широко используются методики, позволяющие уменьшить количество ассоциаций, которое требуется проанализировать.

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

***Алгоритм AIS***

Первый алгоритм поиска ассоциативных правил, называвшийся AIS, был разработан сотрудниками исследовательского центра IBM Almaden в 1993 году. С этой работы начался интерес к ассоциативным правилам; на середину 90-х годов прошлого века пришелся пик исследовательских работ в этой области, и с тех пор каждый год появляется несколько новых алгоритмов.

В алгоритме AIS кандидаты множества наборов генерируются и подсчитываются «на лету», во время сканирования базы данных.

***Алгоритм SETM***

Создание этого алгоритма было мотивировано желанием использовать язык SQL для вычисления часто встречающихся наборов товаров. Как и алгоритм AIS, SETM также формирует кандидатов «на лету», основываясь на преобразованиях базы данных. Чтобы использовать стандартную операцию объединения языка SQL для формирования кандидата, SETM отделяет формирование кандидата от их подсчета. Неудобство алгоритмов AIS и SETM – излишнее генерирование и подсчет слишком многих кандидатов, которые в результате не оказываются часто встречающимися. Для улучшения их работы был предложен алгоритм Apriori.

***Алгоритм Apriori***

Работа данного алгоритма состоит из нескольких этапов, каждый из этапов состоит из следующих шагов:

* формирование кандидатов;
* подсчет кандидатов.

Формирование кандидатов – этап, на котором алгоритм, сканируя базу данных, создает множество *i*-элементных кандидатов (*i* – номер этапа). На этом этапе поддержка кандидатов не рассчитывается.

Подсчет кандидатов – этап, на котором вычисляется поддержка каждого *i*-элементного кандидата. Здесь же осуществляется отсечение кандидатов, поддержка которых меньше минимума, установленного пользователем (min\_sup). Оставшиеся *i*-элементные наборы называем часто встречающимися.

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.

Алгоритм Apriori рассчитывает также поддержку наборов, которые не могут быть отсечены априори. Это так называемая негативная область, к ней принадлежат наборы-кандидаты, которые встречаются редко, их самих нельзя отнести к часто встречающимся, но все подмножества данных на боров являются часто встречающимися.

***Последовательные шаблоны***

Для расширения возможностей анализа транзакционных данных с учетом временного аспекта, последовательности появления предметов и ориентированности на конкретного клиента существует задача Data Mining под названием *последовательные шаблоны* (sequential pattern, time-serial sequential pattern).

Теория последовательных шаблонов во многом основана на теории ассоциативных правил и, по сути, является ее расширением. В частности, базовыми понятиями в ней также являются транзакция, предметный набор, частота набора, поддержка и т.д. Кроме этого, для поиска последовательных шаблонов широко используется адаптированный алгоритм Apriori и его модификации. Но при рассмотрении последовательных шаблонов необходимо учитывать ряд особенностей. Главная из них заключается в том, что если в ассоциативных правилах рассматривается только факт совместного появления товаров в одной транзакции, то в последовательных шаблонах рассматривается последовательность появления товаров. Последовательный шаблон – это всегда последовательность появления предметов и их групп.

Интуитивно понятно, что типичной последовательностью (шаблоном) может быть только такая последовательность, которая встречается в базе данных достаточно часто. Поэтому при поиске последовательных шаблонов возникает та же проблема, что и при поиске ассоциативных правил. Большое число рассматриваемых предметов порождает огромное количество возможных последовательностей, что приводит к серьезным вычислительным затратам при использовании полного перебора. Но и здесь применим принцип антимонотонности – последовательности, содержащие редкие события, не могут быть частыми, что позволяет существенно снизить пространство поиска.

Применение последовательных шаблонов выходит за рамки анализа рыночной корзины. Они позволяют выявлять типичные последовательности событий в самых разнообразных предметных областях.

***Алгоритм AprioriAll***

Является модификацией алгоритма Apriori. На каждом проходе используются частые последовательности, полученные на предыдущем проходе, для генерации последовательностей-кандидатов, а затем вычисляется их поддержка в процессе нового прохода. В дальнейшем она используется для определения частых последовательностей. Частые предметные наборы, найденные на первом проходе, являются, по сути, частыми 1-последовательностями. Иногда этот процесс называют инициализацией. На каждом проходе базы данных алгоритм AprioriAll обрабатывает последовательности только определенной длины *k*. При этом анализ последовательностей всех длин может оказаться затратной по времени процедурой.

Алгоритм AprioriAll формирует частые последовательности-кандидаты всех возможных длин. Однако если из числа последовательностей определенной длины формируется мало частых последовательностей, то эту длину можно пропустить. Алгоритм использует как параметр длину последовательностей, анализируемых на предыдущем проходе, и возвращает длину последовательностей, которые будут анализироваться на следующем. Иными словами, длина последовательностей, искомых на следующем проходе, определяется длиной последовательностей, найденных на предыдущем.