Математические вопросы кибернетики и вычислительной техники 30, 92--104, 2008.

Построение ассоциативных правил путем цепного раздробления n-мерного единичного куба цепями

Левон Асланян, Роберт Хачатрян

Институт проблем информатики и автоматизации НАН РА
lasl@sci.am, robert@simartek.am

Аннотация

В работе решена задача поиска ассоциативных правил и приведен альтернативный алгоритму APRIORI метод решения этой задачи, путем цепного раздробления n-мерного куба, по технике Анселя. Описаны инстументы для работы над цепями, выделенные из результатов Тонояна. Приведено краткое описание программной реализации альтернативного подхода.

1. Введение

Сформулируем задачу построения ассоциативных правил, которая как известно, является одной из основных задач области «дата майнинг» - выуживания знаний из больших массивов экспериментальных данных. Рассмотрим множество $I = \{x_1, ..., x_n\}$ состоящее из n различных элементов (item). Будем рассматривать подмножества элементов (itemset), $X \subseteq I$, и если $|X| = k$ то скажем, что дано k-подмножество. D является базой данных, записи которой (record, trasaction) являются подмножествами элементов (в этом представлении запись - это список, но та же информация может быть эквивалентным образом задана и при помощи характеристического вектора). Предполагаем, что записи могут повторяться (multiset), но они содержат дополнительное поле, которое является ключом в базе данных.

Скажем, что запись $T \in D$ способствующая для множества элементов X, если $X \subseteq T$.

Ассоциативное правило это правило $X \Rightarrow Y$ типа «если-то», выполнение которого связано с некоторыми условиями. Пусть X, Y множества элементов, $X \cap Y = 0$:

Отношение числа всех записей T, способствующих X, и числа всех записей D называется **поддержкой** X (support),

$\sup p(X) = |\{T \in D \mid X \subseteq T\}| / |D|$.

Определяется также поддержка для всего правила $X \Rightarrow Y$,

$\sup p(X \Rightarrow Y) = \sup p(X \cup Y)$.

Другим важным свойством правила $X \Rightarrow Y$ является **надежность / обоснованность** (confidence) определяемая как

$conf(X \Rightarrow Y) = \sup p(X \cup Y) / \sup p(X)$

что является условной вероятностью того, что запись содержит Y, когда известно что она содержит X.

Практическая реализация задачи построения ассоциативных правил является предметом активных теоретических и алгоритмических исследований. Известно, что задача естественно разбивается на два этапа. Первый – это этап построения часто повторяющихся фрагментов, тех которые в базе данных встречаются по частотам, превышающим заранее заданное значение поддержки. Второй этап - это собственно этап синтеза правил по частым наборам.

Наиболее известным алгоритмом синтеза правил является **Apriori**. Он строит множество часто встречающихся наборов так называемым способом наращивания. Рассматриваются одноэлементные подмножества и для них при помощи одного прохода по базе данных вычисляются частоты. Далее рассматриваются все двухэлементные подмножества, одноэлементные подмножества которых часты, и проверяется его повторяемость в таблице. Таким образом частые подмножества наращиваются до состояний, когда включающие его подмножества уже не достаточно частые. Вычислительная сложность здесь значительная, это особенно важно т.к. алгоритм приходиться использовать на весьма больших объемах данных.

Существуют ли альтернативные подходы построения правил? В данной работе предлагается один такой подход, который подключает хорошо известные результаты из области геометрии n-мерного единичного куба к задаче алгоритмического построения ассоциативных правил с данной поддержкой и надежностью.

Краткая характеризация этого подхода такова. n-мерный куб B_n это регулярная решетка состоящая из 2^n вершин – бинарных наборов длины n, которые обычно располагаются по слоям - на k-ом слое все те вершины, которые имеют k единиц. Вершины, отличающиеся по одной координате, называются соседними и соединяются ребром. Цепью в B_n называется последовательность соседних вершин. Цепь растущая, если содержит не более одной вершины из одного слоя.

Ж. Ансель показал что B_n можно разбить на растущие цепи с соблюдением определенных условий. Далее он рассмотрел монотонные Булевы функции и построил алгоритм оптимального распознавания этих функций при помощи построенных цепей. Связь этих построений с ассоциативными правилами в том, что частые наборы по заданным параметрам составляют множество вершин нулевого значения некоторой монотонной Булевой функции.

Прямое использование этой техники для распознавания функций сложно, поскольку построение и хранение цепей это задача экспоненциальной вычислительной и емкостной сложности.

Г. Тоноян сумел предложить вычислительный подход к работе с цепями. Это принципиально и существенно упрощает работу алгоритма распознавания. Идеей данной работы является внедрение указанных разработок в область поиска ассоциативных правил.

Имеется важная особенность задачи поиска ассоциативных правил. Известно, что число всех рассматриваемых элементов n – очень велико. Известно также, что частые наборы состоят из малого числа элементов. Согласно этому выдвигается предположение, что существует значение параметра k такое, что все подмножества этой мощности не являются частыми. Получается, что задача поиска частых наборов эквивалентна задаче расшифровки специального класса монотонных Булевых функций, что в свою очередь требует расширения результатов указанных выше согласно некоторым ограничениям. Расширенные результаты внедрены в задачу синтеза частых наборов, обеспечивая тем самым синтез самих наборов – без рассмотрения системы всех его поднаборов, т.е. части, особенно осложняющей процесс построения по наращиванию.

2. Структуры геометрии n-мерного единичного куба

Булевой переменной называют переменную, принимающую значения 0 и 1.

Булевой функцией от n переменных называют однозначное отображение множества всевозможных наборов значений n булевых переменных в $\{0,1\}$. В качестве области определения булевой функции от n переменных можно рассматривать множество B_n вершин единичного n-мерного куба. B_n это множество всех бинарных векторов $\overline{\alpha} = (\alpha_1, ..., \alpha_n)$, которые называют вершинами, точками. Обычно под B_n подразумевают также некоторую структуру, граф, в котором вершины B_n размещаются по горизонтальным слоям, слой содержит все вершины с данным числом единиц k, $1 \le k \le n$, и слои расположены по вертикали, начиная с нулевого слоя до слоев с номером n. Слой k состоит из C_n^k вершин. Две вершины $\overline{\alpha}$ и $\overline{\beta}$ называются соседними если они отличаются точно одной координатой. Соседние вершины соединяются ребром в структуре B_n.

Вершины множества B_n упорядочим следующим образом: будем говорить, что точка $\overline{\alpha} = (\alpha_1, ..., \alpha_n)$ из B_n предшествует точке $\overline{\beta} = (\beta_1, ..., \beta_n)$ из B_n, если $\alpha_i \le \beta_i$, $1 \le i \le n$. Тот факт, что точка $\overline{\alpha}$ предшествует точке $\overline{\beta}$, обозначается через $\overline{\alpha} \le \overline{\beta}$. Если же $\overline{\alpha} \le \overline{\beta}$ и $\overline{\alpha} \ne \overline{\beta}$, то будем писать $\overline{\alpha} \prec \overline{\beta}$. Две различные точки $\overline{\alpha}$ и $\overline{\beta}$ называются сравнимыми, если выполнено одно из условий $\overline{\alpha} \prec \overline{\beta}$ или $\overline{\alpha} \succ \overline{\beta}$.

Известно, что в общем случае для однозначного определения Булевой функции необходимо знать ее значения во всех точках n-мерного единичного куба. Если же функция принадлежит к некоторому классу, более узкому, чем множество всех Булевых функций, то для однозначного определения ее значений во всех точках B_n не обязательно заранее знать значения функции во всех точках B_n, а иногда достаточно знать значения на некотором подмножестве B_n. Так, для однозначного определения симметрической булевой функции от n переменных (эти функции принимают одно и то же значение на слое B_n) достаточно знать ее значения на множестве $G = \{\overline{\alpha}^0, \overline{\alpha}^1, ..., \overline{\alpha}^n\}$ точек $\overline{\alpha}^i = (\alpha_1^i, ..., \alpha_n^i)$ из B_n таких, что

$$\sum_{j=1}^{n} \alpha_j^i = i \quad (0 \le i \le n).$$

Булева функция $f(x_1, x_2, ..., x_n)$ называется монотонной, если из того, что $\overline{\alpha} \prec \overline{\beta}$, следует, что $f(\alpha_1, \alpha_2, ..., \alpha_n) \le f(\beta_1, \beta_2, ..., \beta_n)$. Класс всех монотонных булевых функций от n переменных обозначается через M_n. Задача расшифровки монотонной булевой функции ставится следующим образом.

Пусть произвольная (неизвестная нам) функция $f \in M_n$ задана при помощи некоторого оператора A_f, который по любому набору $\overline{\alpha} \in B_n$ выдает значение $f(\overline{\alpha})$.

Требуется, при помощи обращений к оператору A_f полностью восстановить таблицу значений заданной функции. При этом после каждого обращения к оператору полученное значение функции f в некоторой точке $(\alpha_1, ..., \alpha_n) \in B_n$ по монотонности распространяется на другие точки B_n. Естественно, что надо стремиться к оптимальности этих алгоритмов, т.е. к минимуму шагов. В качестве шага алгоритма обычно рассматривается одно обращение к оператору A_f, т. е. вычисление одного значения исследуемой функции f.

Рассмотрим множество алгоритмов F, решающих указанную задачу. То есть для произвольной монотонной булевой функции $f(x_1,x_2,...,x_n)$ любой алгоритм из F с помощью оператора A_f полностью восстанавливает таблицу значений монотонной функции $f(x_1,x_2,...,x_n)$. Очевидно, что работа любого алгоритма F будет заключаться в следующем. Алгоритм выбирает некоторую точку $\overline{\alpha}\in B_n$ и с помощью оператора A_f вычисляет значение функции $f(x_1,x_2,...,x_n)$ в точке $\overline{\alpha}$. Полученное значение функции в точке $\overline{\alpha}$ заносится в таблицу значений исследуемой функции. По монотонности определяются значения $f(x_1,x_2,...,x_n)$ в других точках B_n (например, если $f(\overline{\alpha})=1$, то для всех точек $\overline{\beta}$, следующих за $\overline{\alpha}$ (по порядку вершин определенному выше), $f(\overline{\beta})=1$ и соответственно этому заполняются разряды таблицы значений функции f). Затем по некоторому правилу выбирается другая точка B_n, обращаются к A_f и заполняют таблицу значений f, и процесс повторяется до тех пор, пока таблица значений не окажется заполненной полностью.

Очевидно, что любой паре — алгоритму F и монотонной функции $f(x_1,x_2,...,x_n)$ — можно сопоставить число $\varphi(F,f)$—число обращений к оператору A_f в процессе восстановления таблицы значений функции $f(x_1,x_2,...,x_n)$ с помощью алгоритма F.

Естественно оценивать качество алгоритма F функцией $\varphi(F,n)=\max\varphi(F,f)$. Имеем условие $f\in M_n$, где M_n—множество всех монотонных функций от n переменных $x_1,\ldots,x_n$. Сложность распознавания всего класса монотонных функций n переменных можно характеризовать функцией $\varphi(n)=\min\varphi(F,n)$, где минимум берется по всем алгоритмам F, решающим поставленную задачу. Мы получим верхнюю и нижнюю оценку для $\varphi(n)$.

Введем следующие общие понятия. Пусть задан некоторый класс N булевых функций и функция f, принадлежащая этому классу. Множество $G(f, N)$ точек из B_n называется разрешающим множеством для пары (f, N), если из того, что

а) функция g принадлежит классу N,

б) на множестве $G(f, N)$ значения функций f и g совпадают,

следует, что $f=g$.

Для восстановления таблицы значений функции достаточно определить значения функции на некотором ее разрешающем множестве.

Разрешающее множество $G(f, N)$ называется тупиковым разрешающим множеством для (f, N), если никакое его подмножество не является разрешающим для пары (f, N).

Обозначим через $V(\alpha)$ множество точек $\overline{\beta}$, удовлетворяющих условию $\overline{\alpha}\prec\overline{\beta}$, а через $N(\overline{\alpha})$ множество точек $\overline{\gamma}$ таких, что $\overline{\gamma}\prec\overline{\alpha}$.

Верхний нуль монотонной функции $f(x_1,x_2,...,x_n)$ есть точка $(\alpha_1,\alpha_2,...,\alpha_n)$ из B_n такая, что $f(\overline{\alpha})=0$, а $f(\overline{\beta})=1$ для всех точек $\overline{\beta}\in V(\alpha)$.

Нижняя единица монотонной функции $f(x_1,x_2,...,x_n)$ есть точка $\overline{\alpha}$ такая, что $f(\overline{\alpha})=1$, а $f(\overline{\gamma})=0$ для любой точки $\overline{\gamma}\in N(\alpha)$.

Обозначим через $P(f)$ множество всех верхних нулей монотонной функции $f(x_1,x_2,...,x_n)$, а через $Q(f)$-множество всех ее нижних единиц. У каждой монотонной булевой функции существует единственное тупиковое разрешающее множество, входящее во все ее разрешающие множества. Этим тупиковым разрешающим множеством для монотонной булевой функции $f(x_1,x_2,...,x_n)$ является множество $G(f)=P(f)\cup Q(f)$.

В. Коробковым был получен следующий результат относительно верхней и нижней оценки $\varphi(n)$.

Теорема Коробкова

$C_n^{[n/2]} + C_n^{[n/2]+1} \le \varphi(n) \le BC_n^{[n/2]}(1+\varepsilon_n)$, где

$B = \frac{8}{\left(\sqrt[3]{16}-1\right)^{3/2}}$ *и* $\varepsilon_n \to 0$ *при* $n \to \infty$

Доказательство использует монотонную функцию алгебры логики $h(x_1, x_2, ..., x_n)$, определенную следующим образом:

$$h(\alpha_1, \alpha_2, ..., \alpha_n) = \begin{cases} 1, если \cdot [n/2]+1 \le \sum_{i=1}^{n} \alpha_i \le n \\ 0, если \cdot 0 \le \sum_{i=1}^{n} \alpha_i \le [n/2] \end{cases}$$

Очевидно, что $G(h)$ в этом случае содержит ровно $C_n^{[n/2]} + C_n^{[n/2]+1}$ точек.

Приведем определения цепи, относительного дополнения:

1. Возрастающая цепь в структуре B_n, это последовательность $\overline{\beta}_1, ..., \overline{\beta}_n$ элементов B_n, такая, что $\overline{\beta}_{i+1}$ получается из $\overline{\beta}_i$ заменой одного нуля (в наборе значений координат) на единицу.
2. Пусть заданы три элемента $\overline{\alpha}_1 \prec \overline{\alpha}_2 \prec \overline{\alpha}_3$ образующие цепь. Относительным дополнением $\overline{\alpha}_2$ на интервале $[\overline{\alpha}_1, \overline{\alpha}_3]$ является четвертый элемент $\overline{\beta}$, который образует вместе с $\overline{\alpha}_1, \overline{\alpha}_2, \overline{\alpha}_3$ двумерный куб (см. рис.1).

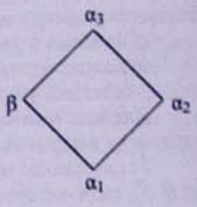

Рис.1.

Лемма Анселя.
Единичный n-мерный куб B_n, наделенный обычным отношением порядка, может быть покрыт множеством из $C_n^{[n/2]}$ попарно непересекающихся цепей, обладающих следующими свойствами :

а) число цепей длины *n-2p+1* равно $C_n^p - C_n^{p-1}$, где $0 \le p \le [n/2]$. Минимальный элемент каждой такой цепи есть набор с *p* единицами и *n-p* нулями, максимальный – с *p* нулями и *n-p* единицами.

b) если заданы три элемента $\overline{\alpha}_1 \prec \overline{\alpha}_2 \prec \overline{\alpha}_3$ образующие цепь и находящиеся на одной и той же цепи длины *n-2p+1*, то относительное дополнение $\overline{\alpha}_2$ на интервале $[\overline{\alpha}_1, \overline{\alpha}_3]$ принадлежит цепи длины *n-2p-1*.

Доказательство этого факта индуктивное, по *n*. Рисунок проясняет, как наибольший элемент $\overline{y}_2$ цепи L_2 удаляется из L_2 и прибавляется к цепи L_1 став ее новым наибольшим элементом.

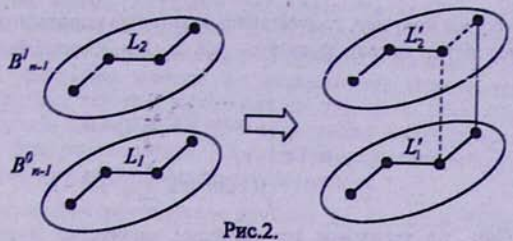

Рис.2.

Теорема. Минимальное число $\varphi(n)$ операций алгоритма распознавания произвольной монотонной булевой функции $f(x_1,x_2,...,x_n)$ равно:

$$\varphi(n) = C_n^{[n/2]} + C_n^{[n/2]+1}$$

Множество всех цепей, построеных по, и удовлетворяющих условиям 1 и 2 леммы Анселя, обозначим через R_n. Множество всех вершин цепей длины l множества R_n обозначим через $[l]_n$. Оптимальный алгоритм распознавания монотонной булевой функции обозначим через F_0. Опишем работу алгоритма F_0.

Оператор A_f вычисляет значения функции $f \in M_n$ на вершинах самых коротких цепей множества B_n. Если известны значения функции f на всех элементах множества $[l]_n$, $0 \le l < n$, то, согласно монотонности функции f, ее значения распространяются на множество $[l+2]_n$ и, согласно лемме Анселя, на каждой цепи L длины $l+2$ значения функции f неизвестны на не более чем двух вершинах. Назовем эти вершины неопределенными вершинами цепи L, соответствующими функции f. Оператор A_f вычисляет значения функции на этих вершинах.

Алгоритм F_0 заканчивает работу, если известны значения функции f на всех вершинах цепи длины n.

До сих пор мы рассматривали известные методы, которые направлены на решение задачи распознавания произвольной монотонной булевой функции $f \in M_n$, значения которой неизвестны на всех точках $\tilde{\alpha} \in B_n$. Рассмотрим класс монотонных булевых функций от n переменных, более узкий чем класс M_n всех монотонных булевых функций. Предположим, что значения функции на точках n-мерного единичного куба, находящихся выше некоторого k-ого слоя ($0<k<[n/2]$), функция принимает значение 1, а на остальных точках неизвестно. Рассмотрим выше описанные технологии для распознавания такого типа монотонных булевых функций.

Дадим оценку $\varphi(n)$ числу операций алгоритма распознавания произвольной вышеописанной монотонной булевой функции.

Теорема. Минимальное число операций $\varphi(n)$ алгоритма распознавания произвольной монотонной булевой функции $f(x_1,x_2,...,x_n)$, при условии, что на точках n-мерного куба, находящихся выше некоторого k-ого слоя, где $0<k<[n/2]$, функция равна единице, есть

$$\varphi(n) = C_n^k + C_n^{k-1}$$

Доказательство. Для доказательства сперва покажем, что $\varphi(n) \geq C_n^k + C_n^{k-1}$, а далее, что $\varphi(n) \leq C_n^k + C_n^{k-1}$.

Нижнюю оценку мы получим, рассматривая некоторую монотонную функцию.
Рассмотрим монотонную булеву функцию $h(x_1, x_2, ..., x_n)$, определенную следующим образом:

$$h(\alpha_1, \alpha_2, ..., \alpha_n) = \begin{cases} 1, если \cdot k \leq \sum_{i=1}^{n} \alpha_i \leq n \\ 0, если \cdot 0 \leq \sum_{i=1}^{n} \alpha_i \leq k-1 \end{cases}$$

Получаем, что $G(h)$, т.е тупиковое разрешающее множество функции $h(x_1, x_2, ..., x_n)$ содержит ровно $C_n^k + C_n^{k-1}$ точек, т.к множеством всех верхних нулей будут точки $(k-1)$-ого слоя n-мерного куба, а множеством всех нижних единиц будут точки k-ого слоя.
Получаем, что $\varphi(n) \geq C_n^k + C_n^{(k-1)}$.

Теперь получим верхнюю оценку.
Известно, что на точках куба выше k-ого слоя функция принимает значения 1. Рассмотрим цепи начинающиеся с k-ого слоя, нам известны значения функции на всех точках таких цепей кроме начальных точек, т.е находящихся на самом k-ом слое. Сделав по одному запросу для каждой цепи начинающейся с k-ого слоя мы полностью определим функцию на этих цепях. Имея значения функции на всех точках цепей начинающихся с k-ого слоя можно использовать свойство b) Леммы Анселя и свойство монотонности для определения значений функции на точках цепей начинающихся с k-1-ого слоя. В результате получается, что на цепях начинающихся с k-1-ого слоя неизвестными остаются значения функций на не более чем двух точках (см. рис.3).
Проделав эту процедуру для оставшихся цепей имеем, что для полного распознавания функции, на цепях начинающихся с k-ого слоя требуется определить значение функции всего лишь на одной точке, а на остальных цепях максимум на двух точках.
Запишем это в виде формулы:

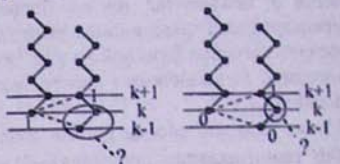

Рис. 3.

$\varphi(n) \leq$(число цепей начинающихся с k-ого слоя)+2(число остальных цепей начинающихся ниже k-ого слоя).
Подставим значения, используя формулу для вычисления количества цепей, описанной в лемме

$$\varphi(n) \leq C_n^k - C_n^{k-1} + 2(C_n^{k-1} - C_n^{k-2} + ... + C_n^1 - C_n^0 + C_n^0).$$

Сделав все сокращения, получаем:

$$\varphi(n) \leq C_n^k + C_n^{k-1}.$$

Объединив оба результата получим:

$$\left.\begin{array}{l} \varphi(n) \geq C_n^k + C_n^{k-1} \\ \varphi(n) \leq C_n^k + C_n^{k-1} \end{array}\right\} \Rightarrow \varphi(n) = C_n^k + C_n^{k-1}.$$

3. Инструменты работы с цепями

Переходим к сопоставлению и анализу имеющихся знаний по цепным разложениям B_n и по вычислениям по цепям. Основные результаты здесь принадлежат Г. Тонояну. Сформулированные результаты привязаны к задаче расшифровки произвольной булевой функции, однако они более универсальны и применимы к решению других аналогичных задач. Поэтому в результате анализа мы выделим так называемые инструменты – независимые процедуры, которые выполняют ту или иную локальную операцию над цепями, и при помощи последовательного применения которых можно решить более глобальные задачи типа распознавания. Конкретным приложением будет являться синтез ассоциативных правил основанный на цепных алгоритмах.

Набор $\overline{\alpha} \in B_n$ назовем l-верхним нулем функции $f \in M_n$, если $f(\overline{\alpha}) = 0$ и для любого набора

(1) $$\overline{\beta} \in \bigcup_{i=2\lfloor n/2 \rfloor}^{l} [i]_n$$

(где $\{x\}$ обозначает дробную часть числа x) из того, что $\overline{\beta} \succ \overline{\alpha}$, следует что $f(\overline{\beta}) = 1$.

Набор $\overline{\alpha} \in B_n$ назовем l-нижней единицей функции $f \in M_n$, если $f(\overline{\alpha}) = 1$ и для любого $\overline{\beta}$ (см. (1)) из того, что $\overline{\beta} \prec \overline{\alpha}$, следует что $f(\overline{\beta}) = 0$.

1. Введем ряд определений, которые несколько отличаются от уже примененных.

Пусть цепь $L = (\overline{\alpha}_{l+1} \prec \ldots \prec \overline{\alpha}_1)$ принадлежит множеству R_n. Скажем, что длина L равна l, а $\overline{\alpha}_i$, $1 \le i \le l+1$, является i-й вершиной цепи L.

Для $\overline{\alpha} \in L$ обозначим через $\overline{\alpha}_{(-k)}$ вершину $\overline{\gamma} \in L$, такую что $\|\overline{\gamma}\| = \|\overline{\alpha}\| - k$, $0 \le \kappa \le \|\alpha\| - \|\alpha_{l+1}\|$. Через $\overline{\alpha}_{(+k)}$ для $\overline{\alpha} \in L$ обозначим вершину $\overline{\beta} \in L$, для которой $\|\overline{\beta}\| = \|\overline{\alpha}\| + k$, $0 \le k \le \|\alpha_1\| - \|\alpha\|$.

Будем считать, что $\overline{\alpha} = (\alpha_1, \ldots, \alpha_n) \in B_n$ удовлетворяет свойству C, если для любого k, $1 \le k \le n$, в $(\alpha_1, \ldots, \alpha_n)$ число нулевых координат среди первых k координат вектора не меньше числа его единичных координат.

Будем считать, что $\overline{\alpha} = (\alpha_1, \ldots, \alpha_n) \in B^n$ удовлетворяет свойству C', если $(\overline{\alpha}_1, \ldots, \overline{\alpha}_n)$ удовлетворяет свойству C, где

$$\overline{\alpha}_i = \begin{cases} 0, & \alpha_i = 1, \\ 1, & \alpha_i = 0. \end{cases}$$

Вершину, которая получается из $\overline{\alpha} = (\alpha_1, \ldots, \alpha_n)$ заменой координат $\alpha_{i_1}, \ldots, \alpha_{i_s}$ соответственно, координатами $\overline{\alpha}_{i_1}, \ldots, \overline{\alpha}_{i_s}$ обозначим через $\overline{\alpha}(i_1, \ldots, i_s)$.

Вершине $\overline{\alpha} = (\alpha_1, \ldots, \alpha_n) \in B^n$ сопоставим набор чисел $K(\overline{\alpha}) = (K_1(\overline{\alpha}), \ldots, K_n(\overline{\alpha}))$ составленных следующим образом:

$$K_1(\overline{\alpha}) = \begin{cases} 2, & \alpha_1 = 0 \\ 1, & \alpha_1 = 1 \end{cases} \quad K_i(\overline{\alpha}) = \begin{cases} K_{i-1}(\overline{\alpha}) + 1, & \alpha_i = 0 \\ K_{i-1}(\overline{\alpha}) - 1, & K_{i-1}(\overline{\alpha}) \ge 2,\ \alpha_i = 1 \\ 1 & K_{i-1}(\overline{\alpha}) = \alpha_i = 1 \end{cases}$$

ИНСТРУМЕНТ 1 "Определение порядкового номера вершины цепи"

Вершина $\overline{\alpha} \in L, L \in R_n$ является $K_n(\overline{\alpha})$-й вершиной цепи L.

На вычислительном уровне – рассматриваются бинарный массив памяти длины n и числовой масив памяти n, в котором числа не превоеходят n. Используются простые операции сравнения и $\pm$, число которых линейно по n.

ИНСТРУМЕНТ 2 "О порядковом номере соседних вершин B_n"

Для вершины $\overline{\alpha} = (\alpha_1, ..., \alpha_n) \in B^n$, *такой что* $\alpha_{i_0} = 0$ *(где* $1 \le i_0 \le n$*), имеем:*

1) $K_n(\overline{\alpha}(i_0))$ *равно* $K_n(\overline{\alpha})$ *тогда и только тогда, когда существует* j, $j \ge 1$, *такое, что* $K_{i_0+j}(\overline{\alpha}) = 1$, *и*

2) $K_n(\overline{\alpha}(i_0))$ *равно* $K_n(\overline{\alpha}) - 1$ *или* $K_n(\overline{\alpha}) - 2$ *в остальных случаях.*

ИНСТРУМЕНТ 3 "Характеризация длины цепей вершин соседних к данному"

Вершины, соседние с вершинами цепи L *длины* l *из множества* R_n *принадлежат цепям длин l—2, l или l+2 того же множества.*

ИНСТРУМЕНТ 4 "Вверх по лестнице"

Если существует r такое, что $K_r(\overline{\alpha}) = 1$, *и* $K_s(\overline{\alpha}) > 1$, $r < s < n$, *то*

$$\overline{\alpha}_{(+1)} = \begin{cases} \alpha(r+1) \text{ при } r < n \\ \phi \quad \text{при } r = n \end{cases}$$

и если не существует такого r, то $\overline{\alpha}_{(+1)} = \overline{\alpha}(1)$.

Пусть $H(\overline{\alpha}) = (n_1, ..., n_s)$, где $n_1, ..., n_s$ – все числа, которые удовлетворяют условиям $K_{n_i - 1}(\overline{\alpha}) = K_{n_i}(\overline{\alpha}) = 1$, $1 \le i \le s$, $n_1 > ... > n_s$, и пусть $H(\overline{\alpha}) = \phi$, если таких чисел нет.

ИНСТРУМЕНТ 5 "Вниз по лестнице"

Набор $\alpha_{(-k)}$ *можно определить следующим образом: если* $H(\overline{\alpha}) = \phi$, *то*

$$\overline{\alpha}_{(-k)} = \begin{cases} (0, \alpha_2, ..., \alpha_n) \text{ при } \alpha_1 = 1 \text{ и } k = 1 \\ \phi \quad \text{в остальных случаях} \end{cases},$$

если $H(\overline{\alpha}) = (n_1, ..., n_s)$, *то*

$$\overline{\alpha}_{(-k)} = \begin{cases} \alpha(n_k, ..., n_1) \text{ при } k \le s \\ \alpha(1, n_s, ..., n_1) \text{ при } \alpha_1 = 1 \text{ и } k = s+1. \\ \phi \quad \text{в остальных случаях} \end{cases}$$

Обозначим множество k-х вершин всех цепей длины l множества R_n через $R(n, l, k)$.

ИНСТРУМЕНТ 6 "Все нижние вершины"

Если $R(n,l,l+1) \ne \emptyset$, *то* $R(n,l,l+1) = \{\overline{\alpha} \in B^n \mid \|\alpha\| = (n-l)/2$ *и* α *удовлетворяет свойству C}.*

ИНСТРУМЕНТ 7 "Все верхние вершины"

$$R(n,l,1)=\left\{\overline{\alpha}=(\alpha_1,\ldots,\alpha_n)\in B^n \mid (\overline{\alpha}_n,\ldots,\overline{\alpha}_1)\in R(n,l,l+1)\right\}$$

ИНСТРУМЕНТ 8 "О цепи дополнения"

Пусть $\overline{\beta}=(\beta_1,\ldots,\beta_n)$ *и* $\overline{\alpha}=(\alpha_1,\ldots,\alpha_n)$ *являются, соответственно, первой и k-й,* $1<k\le l+2$, *вершинами цепи L длины l+2 множества* R_n, $\overline{\alpha}_1\prec\overline{\alpha}\prec\overline{\alpha}_2$ — *цепь,* $\overline{\alpha}_1,\overline{\alpha}_2\in L$, α' — *дополнение* $\overline{\alpha}$ *относительно* $\overline{\alpha}_1$ *и* $\overline{\alpha}_2$ *и* $H(\overline{\beta})=(n_1,\ldots,n_s)$.

Набор α' *является (k-1)-й вершиной цепи l, первая вершина которой*

$$\overline{\gamma}=\begin{cases}\overline{\beta}(n_k), & k\le s\\ \overline{\beta}(1), & k=s+1\end{cases}$$

Пусть имеем $\overline{\alpha}=(\alpha_1,\ldots,\alpha_n)\in B^n$ и некоторое множество $A\subset B^n$. Введем следующие обозначения:

$$\overline{\alpha}\oplus C=\begin{cases}\{\overline{\alpha}(k),\ldots,\overline{\alpha}(n)\}, & \alpha_k=\ldots=\alpha_n=1 \text{ и } \alpha_{k-1}=0 \text{ или } \phi\\ \phi, & \alpha_n=0\end{cases}$$

$$A\oplus C=\{\overline{\alpha}\oplus C\mid\overline{\alpha}\in A\}$$

Множество всех нижних соседних вершин множества A обозначим через $(A)_*$.

ИНСТРУМЕНТ 9 "Все нижние соседние вершины"

$$(R(n,l,l+1))_*=(R(n,l,l+1))\oplus C_1.$$

4. Работа альтернативного алгоритма построения ассоциативных правил

Описанные инструменты созданы и использованы в контексте задачи распознавания произвольной монотонной булевой функции $f\in M_n$, значения которой неизвестны на всех точках $\overline{\alpha}\in B_n$. Снова рассмотрим класс монотонных булевых функций от n переменных, более узкий чем класс M_n всех монотонных булевых функций. Предположим, что значения функции на точках n-мерного единичного куба, находящихся выше некоторого k-ого слоя ($0<k<[n/2]$), функция принимает значение 1, а на остальных точках неизвестно. Рассмотрим выделенные нами Инструменты для распознавания такого типа монотонных булевых функций. Рассмотрим три различных возможных вариантов применения.

1. Нам известно что выше k-ого слоя функция принимает значения 1. Рассмотрим точки k-ого слоя. Для каждой точки определим на цепи какой длины она находится и которым элементом этой цепи она является. Для этого применим процедуру вычисления чисел $K_n(\overline{\alpha})$ для каждой точки k-ого слоя, и в силу **Инстр.1** получим, что вершина $\overline{\alpha}\in L, L\in R_n$ является $K_n(\overline{\alpha})$-й вершиной своей цепи L.

Далее можно определить длину цепи L следующим образом. Знаем что точка $\overline{\alpha}$ имеет k единиц и n-k нулей, где $0\le k\le n$ и известно что $\overline{\alpha}$ является i-ой вершиной своей цепи. Легко посчитать, что первая (наибольшая, верхняя) вершина цепи должна содержать (k+i-1) единиц и следовательно n-(k+i-1) нулей. Учитывая симметричность цепей по отношению к средним слоям, можно посчитать, что последняя вершина цепи состоит из n-(к+i-1) единиц и из (к+i-1) нулей. Так как цепь получается замной одного

нуля в координатах вершины на единицу при каждом пересечении слоя, то длина цепи будет равна разнице числа единиц в первой и последней вершинах цепи, т.е. для цепи $L \in R_n$ получаем, что ее длина равна $l = n - 2(k + i - 1)$.

Из всех точек k-ого слоя выделим те, которые являются последними вершинами своих цепей. Для этих вершин длина цепи равна n-k-k=n-2k. С помощью оператора A_f определяем значения функции на этих точках. Далее рассмотрим цепи длины l+2. В силу симметричности цепей, цепи длины l+2 заканчиваются на k-1- слое. Рассмотрим все точки k-1-ого слоя. Используя **Инстр.6** построим множество $R(n, l+2, l+3)$ – всех последних точек цепей длины l+2, $R(n, l+2, l+3)=\{\overline{\alpha} \in B^n \mid \|\alpha\|=(n-l-2)/2$ *и* $\overline{\alpha}$ *удовлетворяет свойству С}*. К каждой вершине $\overline{\alpha} \in R(n, l+2, l+3)$ l+2 раза применим $\overline{\alpha}_{(+1)}$ и найдем соответствующую ей первую вершину $\overline{\beta}$ цепи длины l+2. Найдем также l+2-ую вершину рассматриваемой цепи, т.е. точку $\overline{\delta}$, предыдущую точке $\overline{\alpha}$. Это точка находится на слое k и для ее нахождения достаточно применить $\overline{\alpha}_{(+1)}$ один раз.

Теперь обратимся к **Инстр.8** Пусть $\overline{\beta} = (\beta_1, ..., \beta_n)$ и $\overline{\delta} = (\delta_1, ..., \delta_n)$ являются, соответственно, первой и k-й, $1 < k \le l+2$, вершинами цепи L длины l+2 множества R_n, $\overline{\alpha}_1 \prec \overline{\delta} \prec \overline{\alpha}_2$ — цепь, $\overline{\alpha}_1, \overline{\alpha}_2 \in L$, $\overline{\delta}'$ — дополнение $\overline{\delta}$ относительно $\overline{\alpha}_1$ и $\overline{\alpha}_2$ и $H(\overline{\beta}) = (n_1, ..., n_s)$ вектор определенный в **Инстр.5**. *Набор* α' *является* (k-1)-*й вершиной цепи длины l, первая вершина которой*

$$\overline{\gamma} = \begin{cases} \overline{\beta}(n_k), & k \le s \\ \beta(1), & k = s+1 \end{cases}.$$

Для каждой найденной точки $\overline{\beta}$, предыдущую точке $\overline{\alpha}$, найдем соответствующую ей точку $\overline{\gamma}$ согласно описанию выше. Из каждой $\overline{\gamma}$ по правилу $\alpha_{(-s)}$ определим точку $\overline{\delta}'$, которая является дополнением для точки $\overline{\delta}$. Точка $\overline{\delta}$ является последней вершиной цепи длины l и находится на k-ом слое, значение функции на этой точке уже вычислено оператором A_f. Распространим по монотонности это значение на точки цепи длины l+2. Оставшиеся неизвестные точки цепи длины l+2 определим с помощью оператора A_f.

На общем шаге рассматривается цепь длины l+m, находится ее первая и последняя вершина, определяется соответствующая первая вершина цепи длины l+m-2, и ее последняя точка, которая является дополнением для предпоследней точки цепи длины l+m. На этом шаге значения функции на всех точках цепи длины l+m-2 уже известны, остается распространить эти значения на цепь длины l+m и с помощью оператора A_f вычислить значение функции на оставшихся неизвестных точках.

2. Как и в предыдущем пункте начнем распознавание функции с k-ого слоя. Определим те точки k-ого слоя, которые являются конечными точками для цепей длины l и с помощью A_f вычислим значение функции на этих точках. Перейдем к цепям длины l+2, и построим мнжество $R(n, l+2, l+3)$. Согласно **Инстр.7** имеет место соотношение

$$R(n, l+2, 1) = \{\overline{\alpha} = (\alpha_1, ..., \alpha_n) \in B^n \mid (\overline{\alpha}_n, ..., \overline{\alpha}_1) \in R(n, l+2, l+3)\}$$

К каждой вершине $\overline{\alpha} \in R(n, l+2, 1)$ применим $\alpha_{(-(l+2))}$ и найдем l+2-ую вершину цепи длины l+2, дополнением которой является l+1-ая вершина цепи длины l, первая вершина которой получается из **Инстр.8**. Такой метод нахождения первых вершин намного упрощает задачу по сравнению со способом описанным в пункте 1., т.к там для определения первой вершины приходилось с помощью $\overline{\alpha}_{(+1)}$ из последней вершины пошагово проходить по всей цепи до достижения первой вершины. В остальном эти методы совпадают.

3. Рассмотрим иной способ распознавания, который отличается от первых двух в начальных обозначениях задачи. Рассмотрим n-мерный единичный куб B_n, каждая вершина которого $\overline{\alpha}=(\alpha_1,...,\alpha_n)$ представляет из себя набор нулей и единиц, где 0 показывает что α_i-ый элемент участвует в транзакции, а 1 что нет. Значение функции на наборе $\overline{\alpha}=(\alpha_1,...,\alpha_n)$ равно 1, если набор част и 0 в обратном случае. В этих определениях частые наборы перемещаются в верхнюю часть куба, и распознаваемая функция принимает следующий вид, на точках куба, находящихся ниже некоторого k-ого слоя, где $n/2<k\le n$, функция принимает значение 0, а на остальных ее значение неизвестно. Все остальные допущения и определения задачи остаются неизменными. Процесс распознавание начинается с цепей, которые начинаются с k-ого слоя, для этого рассматриваются все точки k-ого слоя, вычисляются соответствующие им значения $K_n(\overline{\alpha})$, отбираются те вершины, для которых $K_n(\overline{\alpha})$=1, т.е. это те точки, которые являются начальными точками для цепей начинающихся с k-ого слоя. Вычисляется длина этих цепей и вычисляется значение функции на этих точках, предположим длина цепей есть l. После чего рассматриваются цепи длины l+2, т.е. начинающиеся с k+1-ого слоя. Для этого рассмотрим все точки k+1-ого слоя и выделим те, у которых $K_n(\overline{\alpha})$=1. Вторые точки цепей длины l+2 найдем при помощи $\overline{\alpha}_{(-1)}$, дополнения этих точек находятся на цепях длины l, и являются первыми точками этих цепей, значения функции на которых нам уже известны. Дополнения определяются по **Инстр.8**. Распространим по монотонности значения функции на точки цепей длины l+2. Значения функции на точках оставшихся неопределенными, вычисляются при помощи оператора A_f. Продолжим рассмотрение цепей до тех пор пока не останется неопределенных точек.

4. Программная реализация

Для практического внедрения вышеописанных методов в систему исследования задачи поиска ассоциативных правил была создана ее программная реализация. Для этого была выбрана среда открытого программирования (open source), посвященная задачам анализа данных, известная под названием Orange Canvas. Здесь реализованы алгоритмы пабиолее часто используемых методов раскопок данных (Data Mining) – классификация, кластеризация, деревья решений, поиск ассоциативных правил и т.д. В качестве алгоритма поиска ассоциативных правил используется некая модификация алгоритма Apriori.

Как было отмечено выше, нами предложен альтернативный подход для решения задачи поиска ассоциативных правил на базе монотонных Булевых функций, n-мерного куба и его покрытия цепями Анселя. Без использования результатов Тонояна практически невозможа эффективная программная реализация предложенного метода, так как при больших n оперативные рессурсы обычных компьютеров недостаточны для хранения текущих вычислительных данных. Причина этого в том, что по технике Анселя необходимо построить куб целиком, а также его покрытие и постоянно держать в оперативной памяти. Применение инстументов Тонояна дает возможность вместо постоянного хранения куба и его покрытия, вычислять необходимые точки куба – первые и последние вершины цепей, сосединс вершины заданной точки, относительное дополнение заданной точки на некотором интервале.

Для полной обработки данных иногда бывает недостаточно применеие только ассоциативных правил. Поэтому было решено не только создать программную реализацию алгоритма, а так же внедрить его в существующую систему Orange Canvas, где уже внедрен ряд известных методов анализа данных.

Литература

[1] Коробков Б. К., "О монотонных функциях алгебры логики", сб. 'Проблемы кибернетики', вып. 13, М., 'Наука', стр. 5-28, 1965.

[2] Ансель Ж., "О числе монотонных булевых функций n переменных", 'Кибернетический сборник', Новая серия, вып. 5, М., 'Мир', стр. 53-5, 1968.

[3] Тоноян Г. П., "Разбиение вершин *n*-мерного единичного куба на цепи и расшифровка монотонных булевых функций", *Журнал вычислительной математики и математической физики*, том. 19, N% 6, стр. 1532-1542, 1976.

[4] Kotsiantis S. and Kanellopoulos D., "Association rules mining: A recent overview", GESTS International Transactions on Computer Science and Engineering, vol. 32 (1), pp. 71-82, 2006.

Ասոցիատիվ կանոնների կառուցում՝ *n*-չափանի միավոր խորանարդը շղթաների տրոհելու եղանակով

Լ. Ասլանյան, Ռ.Խաչատրյան

Ամփոփում

Աշխատանքում լուծված է ասոցիատիվ կանոնների փնտրման խնդիրը և բերված է հայտնի APRIORI ալգորիթմի այլընտրանքային տարբերակը այդ խնդրի լուծման համար: Այն հիմնված է n-չափանի միավոր խորանարդը շղթաների տրոհելու վրա, Անսելի կողմից առաջարկված եղանակով: Նկարագրված են գործիքներ շղթաների հետ աշխատելու համար, որոնք դուրս են բերվել Տոնոյանի կողմից ստացված արդյունքներից: Բերված է ծրագրային համակարգի կարճ նկարագիր, որտեղ իրականացված է խնդրի այլընտրանքային լուծման տարբերակը: