ВКФ-метод машинного обучения,

основанного на теории решеток

Дмитрий Виноградов

ФИЦ ИУ РАН

27 февраля 2019 г.

ВКФ-метод и его истоки

Автор предлагает **вероятностно-комбинаторный формальный метод** (ВКФ-метод), развивая когнитивные процедуры логико-комбинаторного ДСМ-метода, модифицируя их:

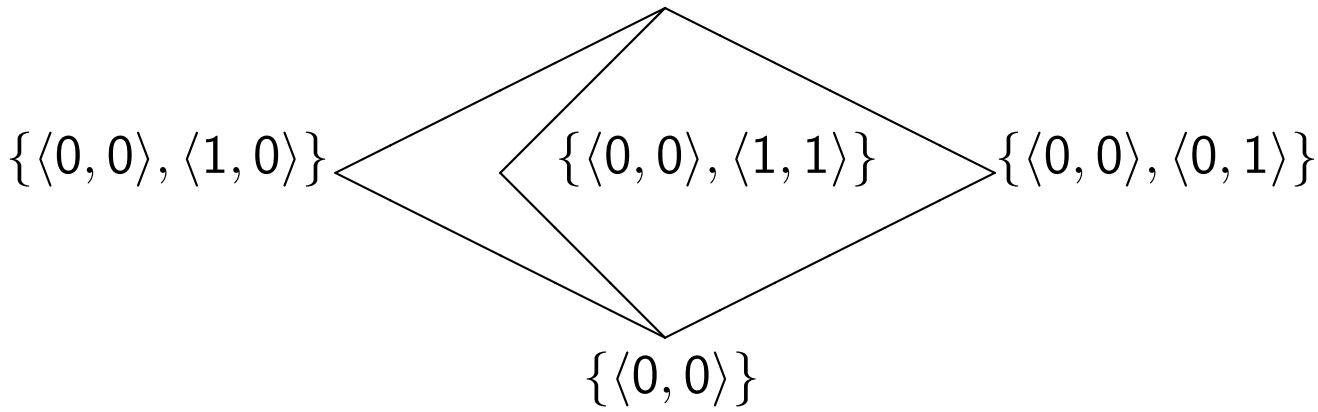
1. индуктивное обобщение обучающих примеров в вероятностно порождаемых ВКФ-гипотезах;
2. абдуктивное уточнение и принятие ВКФ-гипотез (порождая дополнительные гипотезы для объяснения обучающих примеров);
3. предсказание целевого свойства по аналогии с обучающими примерами.

ДСМ-метод был предложен более 35 лет назад в работах В.К.Финна (ВИНИТИ РАН), который уточнил и логически формализовал (в многозначных логиках) идеи логиков и философов XIX-XX веков Д.С.Милля (индуктивная логика), Ч.С.Пирса (абдукция) и К.Поппера (фальсификация). В конце XX века в Западной Европе был создан Анализ Формальных Понятий (АФП) - раздел алгебраической теории решеток. Мы будем использовать его технические результаты.

Недостаточность одной операции сходства

Свойства нижней полурешетки (идемпотентность, коммутативность и ассоциативность) нужны для независимости результата от порядка применения операции сходства $\cap : X \times X \to X$.
Конечную нижнюю полурешетку легко превратить в решетку (с операцией $\cup : X \times X \to X$), добавив наибольший элемент, если его нет.
Без ограничения общности (по теореме Р. Вилле) фрагменты можно представлять битовыми строками с операцией побитового умножения как сходством. Но операция $\cup$ не будет побитовой дизъюнкцией!

$$\mathbb{Z}_2^2 = \{\langle 0,0\rangle, \langle 1,0\rangle, \langle 0,1\rangle, \langle 1,1\rangle\}$$



Кандидаты в гипотезы о причинах

(Формальный) контекст можно понимать как бинарное отношение между элементами множества O, которые мы называем *именами объектов*, и элементами множества F, которые мы называем *признаками*.

$O \times F$	f_1	$\ldots$	f_{j_1}	f_{j_1+1}	$\ldots$	$f_{j_{m-1}}$	$\ldots$	f_{j_m}	$\ldots$	f_n
o_1	0	$\ldots$	0	0	$\ldots$	1	$\ldots$	0	$\ldots$	0
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
o_{i_1}	0	$\ldots$	**1**	1	$\ldots$	**1**	$\ldots$	**1**	$\ldots$	1
o_{i_1+1}	0	$\ldots$	0	0	$\ldots$	1	$\ldots$	1	$\ldots$	0
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
$o_{i_{l-1}}$	1	$\ldots$	**1**	0	$\ldots$	**1**	$\ldots$	**1**	$\ldots$	0
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
o_{i_l}	1	$\ldots$	**1**	1	$\ldots$	**1**	$\ldots$	**1**	$\ldots$	0
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
o_k	0	$\ldots$	1	0	$\ldots$	0	$\ldots$	0	$\ldots$	0

Формальное определение кандидатов

Для подмножества $A \subseteq O$ объектов его *общим фрагментом* называется подмножество $A' = \{f \in F : \forall o \in A\, [oIf]\} \subseteq F$. Полагаем $\emptyset' = F$.
На самом деле, это определение совпадает с последовательным вычислением побитового умножения строк, соответствующих отобранным во множество A объектов.
Для подмножества $B \subseteq F$ признаков его *сходством* называется подмножество $B' = \{o \in O : \forall f \in B\, [oIf]\} \subseteq O$. Полагаем $\emptyset' = O$.
Операции $' : 2^O \to 2^F$ и $' : 2^F \to 2^O$ называются *полярами* и задают *соответствие Галуа*.

Определение

Пару $\langle A, B \rangle$ *назовем* ***кандидатом***, *если* $A = B' \subseteq O$ *и* $B = A' \subseteq F$.
Подмножество $A \subseteq O$ *называем* ***списком родителей*** *кандидата, а* $B \subseteq F$ - ***(общим) фрагментом*** *кандидата.*
Равенство $A = B'$ *соответствует принципу* ***исчерпываемости***.

Контекст, порождающий Булеву алгебру

Пусть $O = \{o_1, o_2, \ldots, o_n\}$ - множество объектов, а $F = \{f_1, f_2, \ldots, f_n\}$ - множество признаков, и формальный контекст равен:

$O \mid F$	f_1	f_2	$\ldots$	f_n
o_1	0	1	$\ldots$	1
o_2	1	0	$\ldots$	1
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
o_n	1	1	$\ldots$	0

Ясно, что любая пара $\langle O \setminus \{o_{j_1}, \ldots, o_{j_k}\}, \{f_{j_1}, \ldots, f_{j_k}\}\rangle$ будет кандидатом, поэтому мы имеем Булеву алгебру всех 2^n подмножеств (=битовых строк). При $n = 32$ обучающая выборка занимает $n \cdot n = 2^{10}$ бит $= 128$ байт. Но чтобы записать результат требуется $n \cdot 2^n = 2^{37}$ бит, т.е. ровно 16 Гигабайт!

В чем проблемы?

1. Потенциальный комбинаторный взрыв: экспоненциальное число ВКФ-гипотез для Булеана.
2. *NP*-полнота и $\#P$-полнота различных задач АФП и ДСМ-метода (С.О. Кузнецов, М.И. Забежайло и др.).
3. Переобучение: возникновение фантомных сходств.

	v_1	...	v_{i_1}	...	v_{i_2}	...	v_{i_k}	...	v_n
v_1	0	...	0	...	1	...	1	...	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
v_{i_1}	0	...	**0**	...	**1**	...	**1**	...	1
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
v_{i_2}	1	...	**1**	...	**0**	...	**1**	...	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
v_{i_k}	1	...	**1**	...	**1**	...	**0**	...	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮	⋱	⋮
v_n	0	...	1	...	0	...	0	...	0

Пример фантомого сходства

Пусть $O = \{o_1 = B737, o_2 = SSJ100, o_3 = IL76, o_4 = A320\}$ будет множеством самолетов, находящихся на ремонте, каждый из которых описывается проблемами из списка
$F = \{f_1 = \text{оперение}, f_2 = \text{двигатель}, f_3 = \text{ругательство}\}$:

$O \mid F$	f_1	f_2	f_3
o_1	1	0	0
o_2	1	0	1
o_3	0	1	1
o_4	0	1	0

Если рассмотреть непустые сходства не менее двух объектов, то мы получим две «настоящие» причины: $\langle \{o_1, o_2\}, \{f_1\} \rangle$ «самолет с поврежденным оперением не летает» и $\langle \{o_3, o_4\}, \{f_2\} \rangle$ «самолет с поврежденным двигателем не летает», и одно «фантомное» сходство $\langle \{o_2, o_3\}, \{f_3\} \rangle$ «самолет, на котором написано ругательство, не летает». Последний кандидат возник из-за случайного совпадения подмножества признаков $\{f_3\}$ у двух примеров o_2 и o_3, каждый из которых имеет свою отличную от других «настоящую» причину.

Контр-примеры и ВКФ-гипотезы

Контр-примером называется объект c, описываемый фрагментом $\{c\}' \subseteq F$ из заданного набора F признаков, но не имеющий целевого свойства.
Говорят, что кандидат $\langle A, B\rangle$, для которого выполняется условие $B \subseteq \{c\}'$, не проходит «запрет контр-примеров».
Любое такое вложение означает, что фрагмент B кандидата $\langle A, B\rangle$ вкладывается в описание $\{c\}'$ контр-примера c. Другими словами, гипотетический механизм есть, а эффект отсутствует. Поэтому сомнительно, что такой кандидат является причиной проявления целевого свойства.
Если кандидат преодолевает все контр-примеры, то он становится **ВКФ-гипотезой** (о причине наличия целевого свойства).
Дополнительно можно потребовать, чтобы число родителей превосходило заданный порог: $|A| \geq b$.
Однако такое ограничение может приводить к «недообучению», когда будут отброшены причины, для которых в обучающей выборке оказалось слишком мало примеров.

Фантомные сходства неустранимы

Теорема

Для $p \geq (-\ln(1-\varepsilon)/n)^{1/b}$ *вероятность появления фантомного сходства* b *случайных* p*-примеров не меньше, чем* $\varepsilon > 0$.

Пусть число n обозначает количество сопутствующих признаков, которыми мы ограничиваемся. Для каждого контр-примера или обучающего примера образуем последовательность n испытаний Бернулли с одинаковой вероятностью успеха p, причем последовательности для разных объектов независимы. Число m будет равно числу контр-примеров.

Теорема

При числе сопутствующих признаков $n \to \infty$ *и вероятности появления этих признаков у контр-примеров и обучающих примеров, равной* $p = \sqrt{\frac{a}{n}}$ $(a \leq 1)$, *вероятность возникновения фантомного сходства двух обучающих примеров, не устраненного никаким из* $m = c \cdot \sqrt{n}$ *контр-примеров, будет стремиться к*

$$1 - e^{-a} - a \cdot e^{-a} \cdot \left[1 - e^{-c \cdot \sqrt{a}}\right] > 0.$$

Фиксированное число контр-примеров

Определение

Назовем ***выжившими*** *на шаге t контр-примеры $\left\langle y_1^k, \ldots, y_t^k \ldots, y_n^k \right\rangle$, для которых* $\forall j \leq t \left[a_j = 1 \Rightarrow y_j^k = 1\right]$.

Будем следить за числом $X_t^{(m)}$ контр-примеров, выживших после одновременного нахождения t-ых признаков m контр-примеров и фантомного сходства. Ясно, что это число должно быть элементом множества $S = \{0, 1, \ldots, m\}$. Нас интересует вероятность $\mathbf{P}\left[X_n^{(m)} = 0\right]$

Производящие функции (многочлены) для распределений $\mathbf{P}\left[X_t^{(m)} = s\right]$ будем обозначать через $\varphi_t^{(m)}(z) = \sum_{j=0}^{m} \mathbf{P}\left[X_t^{(m)} = j\right] \cdot z^j$.

Теорема

$$\varphi_n^{(m)}(z) = \sum_{j=0}^{m} \binom{m}{j} \cdot \prod_{t=1}^{n} \left[p_t^{b+j} + \left(1 - p_t^b\right)\right] \cdot (z-1)^j$$

Произвольное число контр-примеров

Определение

Двойной производящей функцией *для* $\boldsymbol{P}\left[X_n^{(m)} = s\right]$ *назовем формальный ряд*

$$\varphi_n(z,u) = \sum_{m=0}^{\infty}\sum_{s=0}^{m} \boldsymbol{P}\left[X_n^{(m)} = s\right] \cdot z^s \cdot u^m = \sum_{m=0}^{\infty} \varphi_n^{(m)}(z) \cdot u^m.$$

Теорема

$\varphi_n(0,u) = \sum_{j=0}^{\infty} \prod_{t=1}^{n} \left[p_t^{b+j} + \left(1 - p_t^b\right)\right] \cdot \frac{(-u)^j}{(1-u)^{j+\mathbf{1}}}.$

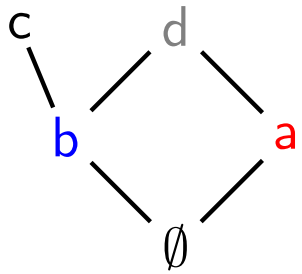
Переобучение неустранимо: эксперименты

А.С. Опарышева в рамках выпускной квалификационной работы (ОИС РГГУ, 2018) исследовала вопрос о подозрительных на «фантомность» ДСМ-гипотез, порожденных системой ДСМ-решатель по фармакологии, созданный с.н.с. ФИЦ ИУ РАН, к.т.н. Д.А. Добрыниным.
В экспериментах исследовались причины мутагенной активности у замещенных нитробензолов. Обучающая выборка была подготовлена в Ливерпульском Университете. Биологически активными было 166 соединений.
Использовались кодирование Фрагментарным Кодом Суперпозиций Подструктур (ФКСП), созданным д.х.н. Авидоном В.В. и к.х.н. Блиновой В.Г., и MNA, предоставленным нам сотрудниками ИБМХ им. В.Н. Ореховича РАН.

кодировка	контр-примеры	число ДСМ-гипотез	число подозрительных
ФКСП	нет	130	13
ФКСП	есть	247	32
MNA	есть	407	105

Алгоритм кодирования битовыми строками

1. топологически сортируем элементы (полу)решетки;
2. в матрице порядка $\geq$ отмечаются столбцы, которые являются побитовым умножением двух предыдущих столбцов ($\cup$-разложимые элементы решетки);
3. все отмеченные столбцы удаляются.



	a	b	c	**d**
a	1	0	0	**0**
b	0	1	0	**0**
c	0	1	1	**0**
d	1	1	0	**1**

$\Rightarrow$

a	b	c
1	0	0
0	1	0
0	1	1
1	1	0

Теорема

Алгоритм кодирования порождает контекст, множество кандидатов которого образует решетку, изоморфную исходной.

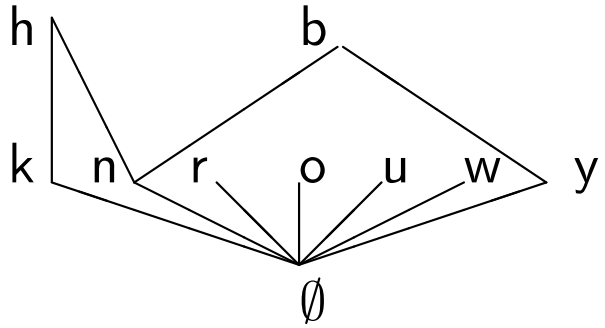
Массив Mushrooms

- Массив взят из Репозитория данных для тестирования алгоритмов машинного обучения http://archive.ics.uci.edu/ml/datasets/Mushroom
- Оцифрованная книга: *Lincoff, G.H.* **The Audubon Society Field Guide to North American Mushrooms**. – NY: Knopf, 1981. – 926 pp.
- Содержит описания 8124 грибов двух видов (съедобные и ядовитые).
- Содержит описания 4208 съедобных и 3916 ядовитых грибов.
- Каждый пример описывался 22 признаками, описывающие различные характеристики грибов. Эти признаки - номинальные, принимающие одно из нескольких значений.

Пример кодирования: грибы

1. форма, поверхность и цвет шляпки
2. синяки?
3. запах
4. присоединение, разреженность, размер и цвет пластинок
5. форма, корень ножки,
6. поверхность ножки над и под колечком, цвет ножки над и под колечком
7. тип и цвет пленки
8. число и тип колечек
9. цвет спор
10. частота встречаемости
11. места произрастания

Пример кодирования: цвет спор



$black = k$	1000000
$brown = n$	0100000
$chocolate = h$	1100000
$green = r$	0010000
$orange = o$	0001000
$purple = u$	0000100
$white = w$	0000010
$yellow = y$	0000001
$buff = b$	0100001

Операции «Замыкай-по-одному»

Операция **замыкай-по-одному-вниз** на кандидате $\langle A, B\rangle$ и объекте $o \in O$ порождает кандидат

$$CbODown(\langle A, B\rangle, o) = \langle A, B\rangle \wedge \langle \{o\}'', \{o\}'\rangle = \langle (A \cup \{o\})'', B \cap \{o\}'\rangle.$$

Операция **замыкай-по-одному-вверх** на кандидате $\langle A, B\rangle$ и признаке $f \in F$ порождает кандидат

$$CbOUp(\langle A, B\rangle, f) = \langle A, B\rangle \vee \langle \{f\}', \{f\}''\rangle = \langle A \cap \{f\}', (B \cup \{f\})''\rangle.$$

Ускорение вычислений:
Если $o \in A$, то $CbODown(\langle A, B\rangle, o) = \langle A, B\rangle$. Аналогично, если $f \in B$, то $CbOUp(\langle A, B\rangle, f) = \langle A, B\rangle$.
В случае Булеана вычисления упрощаются
Если $o_j \notin A$, то $CbODown(\langle A, B\rangle, o_j) = \langle A \cup \{o_j\}, B \setminus \{f_j\}\rangle$, и если $f_j \notin B$, то $CbOUp(\langle A, B\rangle, f_j) = \langle A \setminus \{o_j\}, B \cup \{f_j\}\rangle$.

Алгоритм спаривающей цепи Маркова

Data: множество обучающих (+)-примеров; внешние функции
CbOUp(,) и *CbODown*(,) операций «замыкай-по-одному»
Result: кандидат $\langle A, B\rangle$
$O :=$ (+)-примеры, $F :=$ признаки; $I \subseteq O \times F$ - формальный контекст для (+)-примеров;
$R := O \cup F$; $Min := \langle O, O'\rangle$; $Max := \langle F', F\rangle$;
while $(Min \neq Max)$ **do**
  Выбираем случайный элемент $r \in R$;
  if $(r \in O)$ **then**
    $Min := CbODown(Min, r)$; $Max := CbODown(Max, r)$;
  end
  else
    $Min := CbOUp(Min, r)$; $Max := CbOUp(Max, r)$;
  end
end
$\langle A, B\rangle := Min$;

Algorithm 1: Спаривающая цепь Маркова

Спаривающая цепь Маркова

Состоянием изменяемых переменных в цикле (= состоянием цепи Маркова) является упорядоченная пара кандидатов $\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle$.

Определение

Порядок *на кандидатах:* $\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle$, *если* $B_1 \subseteq B_2$.

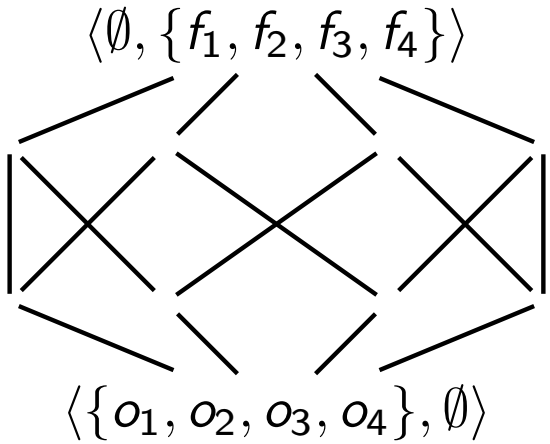
Первоначально меньший кандидат совпадает с наименьшим кандидатом $Min := \langle O, O'\rangle$, а больший - с наибольшим $Max := \langle F', F\rangle$.
В цикле к обоим кандидатам применяется одна и та же операция *CbODown* с выбранным объектом, или *CbOUp* с выбранным признаком.

Лемма

Для всякой упорядоченной пары кандидатов $\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle$ *и любого* $o \in O$ *имеем* $CbODown(\langle A_1, B_1\rangle, o) \leq CbODown(\langle A_2, B_2\rangle, o)$.
Для всякой упорядоченной пары кандидатов $\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle$ *и любого* $f \in F$ *имеем* $CbOUp(\langle A_1, B_1\rangle, f) \leq CbOUp(\langle A_2, B_2\rangle, f)$.

Процесс останавливается, когда меньший кандидат совпадет в большим. Тогда этот общий кандидат и выдается алгоритмом 1.

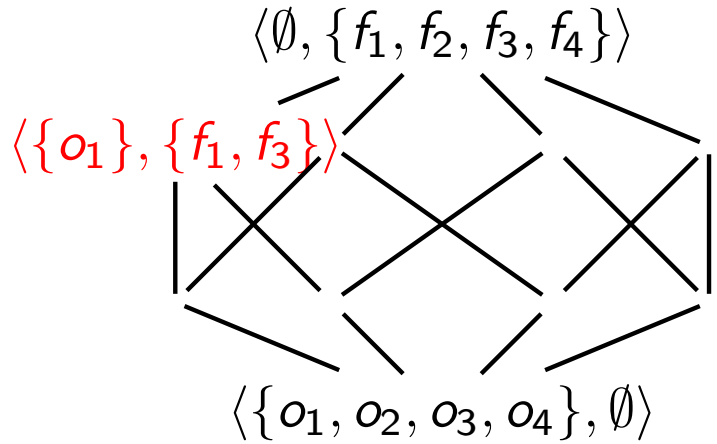
Как работает спаривающая цепь Маркова: шаг 0



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	1	0	1	0	o_1	***1***	***0***	***1***	***0***
o_2	1	0	0	1	o_2	***1***	***0***	***0***	***1***
o_3	0	1	1	0	o_3	***0***	***1***	***1***	***0***
o_4	0	1	0	1	o_4	***0***	***1***	***0***	***1***

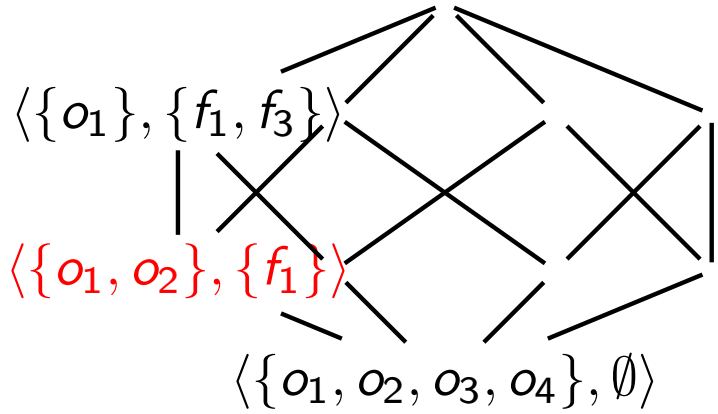
Как работает спаривающая цепь Маркова: выбор o_1



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	***1***	***0***	***1***	***0***	o_1	***1***	***0***	***1***	***0***
o_2	1	0	0	1	o_2	***1***	***0***	***0***	***1***
o_3	0	1	1	0	o_3	***0***	***1***	***1***	***0***
o_4	0	1	0	1	o_4	***0***	***1***	***0***	***1***

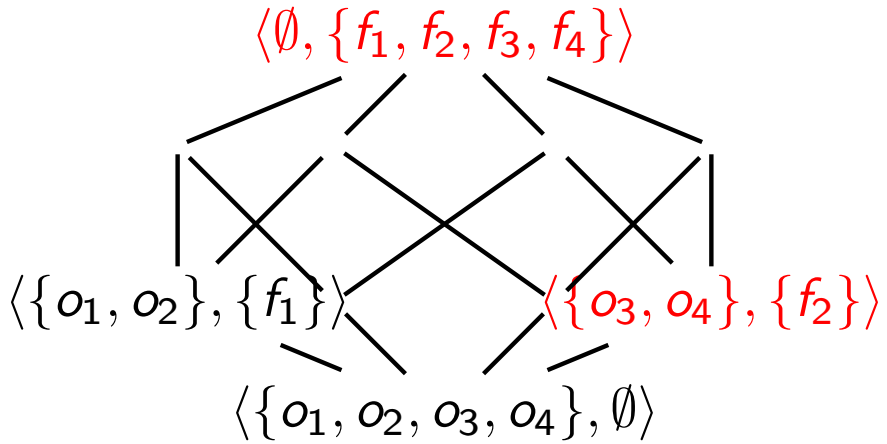
Как работает спаривающая цепь Маркова: выбор o_2



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	***1***	***0***	***1***	***0***	o_1	***1***	***0***	***1***	***0***
o_2	***1***	***0***	***0***	***1***	o_2	***1***	***0***	***0***	***1***
o_3	0	1	1	0	o_3	***0***	***1***	***1***	***0***
o_4	0	1	0	1	o_4	***0***	***1***	***0***	***1***

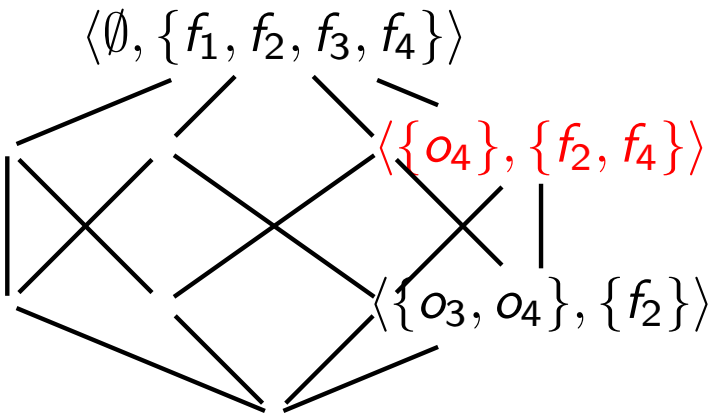
Как работает спаривающая цепь Маркова: выбор f_2



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	1	0	1	0	o_1	1	0	1	0
o_2	1	0	0	1	o_2	1	0	0	1
o_3	0	1	1	0	o_3	***0***	***1***	***1***	***0***
o_4	0	1	0	1	o_4	***0***	***1***	***0***	***1***

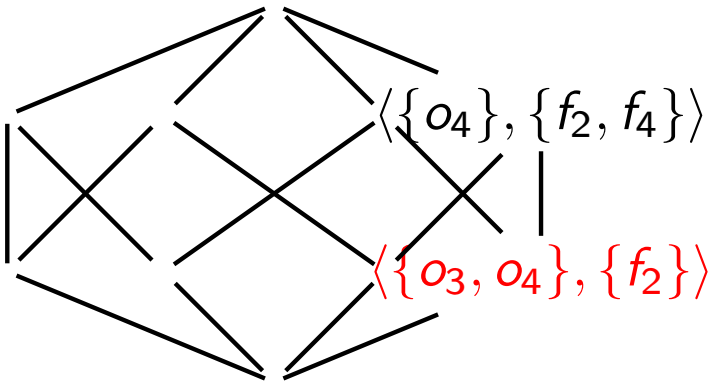
Как работает спаривающая цепь Маркова: выбор o_4



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	1	0	1	0	o_1	1	0	1	0
o_2	1	0	0	1	o_2	1	0	0	1
o_3	0	1	1	0	o_3	***0***	***1***	***1***	***0***
o_4	***0***	***1***	***0***	***1***	o_4	***0***	***1***	***0***	***1***

Как работает спаривающая цепь Маркова: выбор o_3



Пример

верхний	f_1	f_2	f_3	f_4	нижний	f_1	f_2	f_3	f_4
o_1	1	0	1	0	o_1	1	0	1	0
o_2	1	0	0	1	o_2	1	0	0	1
o_3	***0***	***1***	***1***	***0***	o_3	***0***	***1***	***1***	***0***
o_4	***0***	***1***	***0***	***1***	o_4	***0***	***1***	***0***	***1***

Свойства спаривающей цепи Маркова

Теорема

Алгоритм 1 соответствует цепи Маркова.

Теорема

Вероятность того, что состояние цепи Маркова в момент времени t окажется невозвратным, стремится к нулю, когда $t \to \infty$.

Определение

Состояние вида $\langle A, B\rangle = \langle A, B\rangle$ спаривающей цепи Маркова для совпадающей пары кандидатов называется ***эргодическим****. Состояние вида $\langle A_1, B_1\rangle < \langle A_2, B_2\rangle$ называется* ***невозвратным****.*

Следствие

Вероятность $\langle A_1(t), B_1(t)\rangle = \langle A_2(t), B_2(t)\rangle$ стремится к единице, когда $t \to \infty$ (т.е. алгоритм 1 останавливается почти наверное).

Длина траекторий для Булеана

Теорема

Средняя длина траекторий $\sum_{j=1}^{n} T_j$ *для* n*-мерного гиперкуба равна*

$$\boldsymbol{E}[\sum_{j=1}^{n} T_j] = \sum_{j=1}^{n} \frac{n}{j} \approx n \cdot \ln(n) + n \cdot \gamma + \frac{1}{2}.$$

При $n = 32$ средняя длина $\sum_{j=1}^{n} \frac{n}{j} \leq 130$.

Теорема

$\boldsymbol{P}\left[\sum_{j=1}^{n} T_j \geq (1+\varepsilon) \cdot n \cdot \ln(n)\right] \to 0$ *при* $n \to \infty$ *для любого* $\varepsilon > 0$.

При $n = 32$ Булев гиперкуб содержит $4,294,967,296$ вершин. 1000 траекторий спаривающей цепи Маркова породят около $260,000$ элементов гиперкуба.

Ленивые вычисления

Согласно определению

$$CbODown(\langle A, B\rangle, o) = \langle (A \cup \{o\})'', B \cap \{o\}'\rangle.$$

Вычисление сходства $B \cap \{o\}' = (A \cup \{o\})'$ соответствует побитовому умножению соответствующих строк, но операция $(A \cup \{o\})''$ формирования нового списка родителей может потребовать побитово перемножить с полученным ранее сходством почти все объекты, чтобы проверить, обладает ли еще какой-нибудь объект полученным сходством.
Для улучшения ситуации предлагается (лениво) откладывать вычисления второй производной, пока последовательный выбор нескольких объектов для *CbODown* не сменится выбором признака с переходом к операции *CbOUp*. Этом случае нужно использовать равенство $(A \cup \{o\})'' = (B \cap \{o\}')'$.
Используя равенство $(B \cup \{f\})'' = \left(A \cap \{f\}'\right)'$, можно аналогично откладывать вычисления второй производной до тех пор, пока выбор нескольких признаков для *CbOUp* не сменится выбором объекта с переходом к операции *CbODown*.

Алгоритм ленивой спаривающей цепи Маркова

Data: множество обучающих (+)-примеров
Result: кандидат $\langle A_1, B_1 \rangle$
$R := O \cup F$; $\langle A_1, B_1 \rangle := \langle O, O' \rangle$; $\langle A_2, B_2 \rangle := \langle F', F \rangle$; $moveUp := true$;
while $(\langle A_1, B_1 \rangle \neq \langle A_2, B_2 \rangle)$ **do**
  Выбираем случайный элемент $r \in R$;
  if $(r \in O \&\& moveUp)$ **then**
    $B_1 := A_1'$; $B_2 := A_2'$; $moveUp := false$;
  end
  if $(r \in O)$ **then**
    $B_1 := B_1 \cap (\{r\}')$; $B_2 := B_2 \cap (\{r\}')$;
  end
  if $(r \in F \&\& !moveUp)$ **then**
    $A_1 := B_1'$; $A_2 := B_2'$; $moveUp := true$;
  end
  if $(r \in F)$ **then**
    $A_1 := A_1 \cap (\{r\}')$; $A_2 := A_2 \cap (\{r\}')$;
  end
end

Algorithm 2: Ленивая спаривающая цепь Маркова

Выигрыш от ленивых вычислений: теория

Теорема

В ленивой схеме вычислений на каждую пару применений операции замыкания (одной в CbOUp и одной в CbODown) в среднем в классической схеме мы будем делать $\frac{(n+k)^2}{k \cdot n}$ операций замыкания, где k - число обучающих примеров, а n - число признаков, используемых для описания объектов.

Так как

$$\frac{(n+k)^2}{k \cdot n} - 4 = \frac{(n-k)^2}{k \cdot n} \geq 0,$$

то

$$\frac{(n+k)^2}{k \cdot n} \geq 4.$$

В худшем случае ($k = n$) это сокращение вызовов трудоемкой операции в среднем $\frac{4}{2} = 2$ раза.
Чем сильнее различаются k и n, тем больше средний выигрыш от применения ленивой схемы вычислений.

Выигрыш от ленивых вычислений: проверка

Л.А. Якимова в рамках выпускной квалификационной работы (ОИС РГГУ, 2018) исследовала вопрос фактическом преимуществе вычислений ВКФ-кандидатов с помощью ленивого варианта спаривающей цепи Маркова.

- Исходные данные включают описания 8124 грибов, разделенные на две категории (съедобные и ядовитые).
- Обучающая выборка содержит $k = 4208$ примеров (съедобные грибы).
- Контр-примеров (ядовитые грибы) содержится 3916 штук.
- Каждый пример описывался 22 признаками, описывающие различные характеристики грибов (цвет, форма шляпки, места произрастания,частота встречаемости и т.п.). ВКФ-система закодировала эти признаки битовыми строками длины $n = 124$ бит.
- Число $\frac{(n+k)^2}{k \cdot n} = 35,96$. Поэтому максимальное ускорение может составить примерно $\frac{36}{2} = 18$ раз.
- На практике ускорение вычислений по ленивой схеме превысило 17 раз!

Остановленная цепь Маркова

Для устранения слишком длинных траекторий спаривающей цепи Маркова:

Определение

Если $T_1, \ldots, T_r$ – независимые целочисленные случайные величины, имеющие распределение времени склеивания T, то ***верхняя граница склеивания*** *по r испытаниям определяется как $\hat{T} = T_1 + \cdots + T_r$.*

Остановленная цепь Маркова $\mu(\hat{T})$: если спаривающая цепь Маркова μ не склеивается до времени $\hat{T}$, то начинаем заново, иначе выдаем $\langle A_1(T), B_1(T)\rangle = \langle A_2(T), B_2(T)\rangle$ $(T \leq \hat{T})$.

Теорема

Для верхней границы $\hat{T}$ склеивания по $r > 1$ испытаниям любого $\| \mu - \mu(\hat{T}) \|_{TV} \leq \frac{1}{2^r - 1}$ в метрике тотальной вариации.

Алгоритм индуктивного обобщения

Data: множество обучающих (+)- и (-)-примеров; число N порождаемых ВКФ-гипотез
Result: выборка S ВКФ-гипотез
$O :=$ (+)-примеры, $F :=$ признаки; $I \subseteq O \times F$ формальный контекст для (+)-примеров; $C :=$ (-)-примеры; $S := \emptyset$; $i := 0$;
while $(i < N)$ **do**
  породить кандидата $\langle A, B \rangle$ с помощью цепи Маркова;
  $hasObstacle :=$ **false**;
  for $(c \in C)$ **do**
    if $(B \subseteq c')$ **then**
      $hasObstacle :=$ **true**;
    end
  end
  if $(hasObstacle =$ ***false***$)$ **then**
    $S := S \cup \{\langle A, B \rangle\}$;
    $i := i + 1$;
  end
end

Algorithm 3: Процедура индуктивного обобщения

Алгоритм абдуктивного уточнения

Data: выборка S ВКФ-гипотез, внешняя функция $CbODown(\ ,\ )$
операции «замыкай-по-одному-вниз»
Result: расширенная выборка S^+ ВКФ-гипотез
$S^+ := \emptyset$; $O :=$ (+)-примеры; $C :=$ (-)-примеры;
for $(o \in O$ *and* $\langle A, B\rangle \in S)$ **do**
  вычислить $\langle X, Y\rangle := CbODown(\langle A, B\rangle, o)$;
  $Explained(o) :=$ **false**; $hasObstacle :=$ **false**;
  for $(c \in C)$ **do**
    if $(Y \subseteq c')$ **then**
      $hasObstacle :=$ **true**;
    end
  end
  if $(hasObstacle =$ ***false***$)$ **then**
    $S^+ := S^+ \cup \{\langle X, Y\rangle\}$;
    $Explained(o) :=$ **true**;
  end
end

Algorithm 4: Процедура абдуктивного уточнения

Алгоритм предсказания по аналогии

Data: расширенная выборка S^+ ВКФ-гипотез, файл (τ)-примеров
Result: предсказанные свойства (τ)-примеров
$X := (\tau)$-примеры;
for $(o \in X)$ **do**
  $PredictPositively(o) :=$ **false**;
  for $(\langle A, B\rangle \in S^+)$ **do**
    if $(B \subseteq o')$ **then**
      $PredictPositively(o) :=$ **true**;
    end
  end
end

Algorithm 5: Процедура предсказания по аналогии

Надежность ВКФ-гипотез

Зафиксируем $\varepsilon > 0$ - точность предсказания.

Определение

*Объект o назовем ε-**важным**, если суммарная вероятность появления таких ВКФ-гипотез $\langle A, B \rangle$, которые предсказывают его положительно, будет больше ε.*

Теорема

Для n признаков и любых $\varepsilon > 0$ и $1 > \delta > 0$ достаточно породить

$$N \geq \frac{2 \cdot (n+1) - 2 \cdot \log_2 \delta}{\varepsilon}$$

ВКФ-гипотез, чтобы вероятностью $> 1 - \delta$ все ε-важные объекты могли быть предсказаны положительно.

Программная реализация

Автором была создана программная система, получившей название ВКФ-система:

- Программа реализована как библиотека разделяемого доступа. Она была создана в среде Visual Studio Code с использованием библиотеки boost (версия 1_65_1 или более поздней).
- Примеры (обучающие, контр- и представленные для предсказания целевого свойства) представляются объектами класса *boost* :: *dynamic_bitset*. Они сохраняются в контейнерах типа *std* :: *vector* стандартной библиотеки C++. Реализована архивация результатов классами *boost* :: *serialization*.
- Программа использует классы *std* :: *random* для датчиков случайных чисел. Это нужно для спаривающей цепи Маркова (алгоритм 1).
- Для реализации многопоточности используются классы *std* :: *thread*.
- Библиотека платформенно независима: она собирается и линкуется под Windows и под Linux (с использованием классов *boost* :: *dll*). Компиляторы C++: GNU C++ toolset (Linux) и Microsoft Visual BuildTools (Windows).

Достоинства ВКФ-системы

По сравнению с классическим ДСМ-подходом:

- Так как каждая ВКФ-гипотеза порождается независимым запуском цепи Маркова, то ВКФ-программа использует несколько потоков для вычисления индуктивного обобщения. Для ДСМ-системы подобное распараллеливание индукции невозможно.
- ВКФ-система вычисляет процедуру абдуктивного уточнения и принятия ВКФ-гипотез тоже в несколько потоков. В ДСМ-системе распараллеливание шага абдукции возможно, но пока не реализовано.
- Предсказание свойств по аналогии осуществляется в один поток, так как вычислительная сложность этого шага мала в сравнении с шагом индукции.
- На ЦПУ с 4 потоками (i5-4220Y) максимальная нагрузка процессора при вычислении в 4 потока достигает 90%. Для существующей параллельной версии ДСМ-системы она не превосходит 50%.

Массив SPECT Hearts

- Обучающая выборка содержит 40 (+)- и 40 (-)-примеров.
- Тестовая выборка содержит 172 (+)- и 15 (-)-примеров.
- Каждый пример описывался 22 бинарными атрибутами.
- ВКФ-система добавила отрицания исходных признаков, чтобы отсутствие атрибута могло быть частью причины проявления свойства. Поэтому обучающая выборка - это матрица 40×44.
- Точность предсказания простейшей ВКФ-системы достигла 86.1% (151 из 172 (+)-примеров и 10 из 15 (-)-примеров).
- Авторы массива SPECT достигли 84.0% точности своей программой CLIP3, которая реализует обучение покрытию средствами целочисленного программирования.

Массив Mushrooms

- Исходные данные включают описания 8124 грибов, разделенные на две категории (съедобные и ядовитые). Мы случайным образом разделили их на обучающую и тестовую выборки.
- Обучающая выборка содержит 4032 объекта.
- Тестовая выборка содержит 2120 (+)- (съедобные грибы) и 1972 (-)-примеров (ядовитые грибы).
- Каждый пример описывался 22 признаками, описывающие различные характеристики грибов (цвет, форма шляпки, места произрастания,частота встречаемости и т.п.). Эти признаки - номинальные, принимающие одно из нескольких значений.
- ВКФ-система закодировала эти признаки битовыми строками длины 110 бит.
- Точность предсказания простейшей ВКФ-системы достигла 100% для 80/150 ВКФ-гипотез без абдуктивного уточнения (число порожденных гипотез и время вычислений уменьшилось на несколько порядков).

Ваши теоремы ничего не доказывают! (с)

1. Теорема об оценке снизу вероятности возникновения фантомного сходства без учета контр-примеров.
2. Оценка асимптотической вероятности появления фантомного сходства при наличии контр-примеров.
3. Явный вид производящих функций для вероятности возникновения фантомного сходства при наличии контр-примеров.
4. Доказательство того, что алгоритмы вероятностного нахождения сходств задают цепи Маркова.
5. Теорема об остановке с вероятностью единица алгоритма спаривающей цепи Маркова.
6. Оценка среднего времени склеивания и теорема о сильной концентрации времени склеивания около его среднего для случая Булеана.
7. Теорема об оценке изменения вероятностей результатов спаривающей цепи Маркова, остановленной по границе, вычисляемой по r прогонам.
8. Теорема о числе ВКФ-гипотез, чтобы с заданной надежностью можно было предсказать положительно все ε-важные объекты.
9. Оценка эффективности ленивых вычислений на шаге индукции.

Направления будущих исследований

Теперь мы сформулируем открытые проблемы:

- Исследовать вопрос о времени перемешивания для монотонной цепи Маркова. Следует отметить, что в частном случае Булеана подобный результат мной получен.
- Получить оценку среднего времени склеивания в общем случае. Полезно указать, что метрика Хэмминга между верхним и нижним кандидатами не является функцией Ляпунова (может возрастать) в спаривающей цепи Маркова.
- Исследовать асимптотическую вероятность возникновения фантомного сходства, когда число контр-примеров растет, а число признаков сохраняется. Автор надеется, что производящие функции, которые он получил, окажутся при этом полезными.

Надежность предсказания: комментарии

1) Цепь Маркова порождает гипотезы, при этом независимые траектории порождают независимые элементы решетки кандидатов. Тестовые примеры задают подмножества точек (отсекаемые гиперплоскостями) на гиперкубе, где должны оказаться ВКФ-гипотезы. Это дуально парадигме Вапника-Червоненкиса (там гипотезы определяют подмножества, куда должны попасть обучающие точки).
2) Специфика заключается в рассмотрении точек (обучающих и тестовых примеров) в вершинах единичного гиперкуба. Нижние линейные полупространства на точках гиперкуба имеет очень малую емкость. Поэтому метод повторной выборки не дает дополнительного завышающего множителя. Хотя оценка все равно завышена по другим причинам.
3) Мы рассматриваем только ошибки первого рода, когда положительный тестовый пример не предсказывается положительным. Про неправильное предсказание отрицательных примеров речь не идет.

Приложение 1: Надежность предсказания-1

Определение

Объект o, описываемый фрагментом $o' \subseteq F$ (множеством признаков), ***предсказывается положительным*** *с помощью ВКФ-гипотезы $\langle A, B \rangle$, если $B \subseteq o'$.*

Если число признаков равно $n = |F|$, то можно рассматривать вершины n-мерного гиперкуба $\{0,1\}^n \subseteq \boldsymbol{R}^n$.
Каждый объект o, предъявляемый для предсказания, задает семейство нижних полупространств в $\boldsymbol{R}^n$:

Определение

Нижнее полупространство *$H^{\downarrow}(o)$, определяемое объектом o с фрагментом $o' \subseteq F$, задается линейным неравенством $x_{j_1} + \ldots + x_{j_k} < \frac{1}{2}$, где $F \setminus o' = \{f_{j_1}, \ldots, f_{j_k}\}$. Допускается также вырожденное нижнее полупространство $0 < \frac{1}{2}$, соответствующее $o' = F$, и совпадающее со всем $\boldsymbol{R}^n$.*

Приложение 1: Надежность предсказания-2

Лемма

Объект o предсказывается положительным тогда и только тогда, когда в любом нижнем полупространстве $H^{\downarrow}(o)$ содержится фрагмент B хотя одной ВКФ-гипотезы $\langle A, B\rangle$.

Определение

*Объект o назовем ε-**важным**, если суммарная вероятность появления таких ВКФ-гипотез $\langle A, B\rangle$, что $B \in H^{\downarrow}(o)$ будет больше ε.*

Теперь нас будет интересовать только вероятность ошибки «первого рода» (отказ от положительного предсказания): требуется найти такое число N, зависящее от ε и δ, чтобы с вероятностью, большей $1-\delta$, случайная выборка объема N будет образовывать ε-сеть.

Приложение 1: Надежность предсказания-3

Лемма

$\boldsymbol{P}\{B_{p,N} \geq \boldsymbol{E}[B_{p,N}] - 1\} \geq \frac{1}{2}$ *для биномиальной случайной величины* $B_{p,N}$.

Для полупространства $\boldsymbol{P}H > \varepsilon$ выполняется

$$\boldsymbol{P}^N\{S_2 : N \cdot \boldsymbol{P}H - |S_2 \cap H| \leq \frac{\varepsilon \cdot N}{2}\} \leq \boldsymbol{P}^N\{S_2 : |S_2 \cap H| > \frac{\varepsilon \cdot N}{2}\}.$$

Лемма

Для любого ε *при* $N > \frac{2}{\varepsilon}$ *для независимых случайных выборок* S_1 *и* S_2 *ВКФ-гипотез объемов* N *имеем оценку:*

$$\boldsymbol{P}^N\{S_1 : \exists H \in (Sub\downarrow)\,[S_1 \cap H = \emptyset, \boldsymbol{P}H > \varepsilon]\} \leq$$
$$\leq 2 \cdot \boldsymbol{P}^{2N}\{S_1 S_2 : \exists H \in (Sub\downarrow)\,[S_1 \cap H = \emptyset, |S_2 \cap H| > \varepsilon \cdot N/2]\}.$$

Приложение 1: Надежность предсказания-4

Лемма

Для любого ε для двух независимых случайных выборок S_1 и S_2 ВКФ-гипотез объемов N имеем оценку:

$$\boldsymbol{P}^{2N}\{S_1 S_2 : \exists H \in (Sub\downarrow)\left[S_1 \cap H = \emptyset, |S_2 \cap H| > \frac{\varepsilon \cdot N}{2}\right]\} \leq 2^n \cdot 2^{-\frac{\varepsilon N}{2}}.$$

Теорема

Для n признаков и любых $\varepsilon > 0$ и $1 > \delta > 0$ достаточно породить

$$N \geq \frac{2 \cdot (n+1) - 2 \cdot \log_2 \delta}{\varepsilon}$$

ВКФ-гипотез, чтобы вероятностью $> 1 - \delta$ все ε-важные объекты могли быть предсказаны положительно.

Решаем неравенство $2 \cdot 2^n \cdot 2^{-\varepsilon N/2} \leq \delta$ относительно N, чтобы получить утверждение теоремы.

Приложение 2: Остановленная цепь-1

Определение

Если $T_1, \ldots, T_r$ – независимые целочисленные случайные величины, имеющие распределение времени склеивания T, то ***верхняя граница склеивания*** *по r испытаниям определяется как $\hat{T} = T_1 + \cdots + T_r$.*

Если спаривающая цепь Маркова не склеивается до времени $\hat{T}$, то начинаем заново, иначе выдаем $\langle A_1(T), B_1(T)\rangle = \langle A_2(T), B_2(T)\rangle$.

Определение

Для целочисленной случайной величины $\hat{T}$, независимой от целочисленной случайной величины T, ***условное распределение*** *состояний относительно события $B = \{T \le \hat{T}\}$ есть распределение*

$$\mu_{\hat{T},i} = \frac{\boldsymbol{P}[X_T = i, T \le \hat{T}]}{\boldsymbol{P}[T \le \hat{T}]}$$

для любого эргодического состояния i.

Приложение 2: Остановленная цепь-2

Определение

Расстояние ***тотального изменения*** *между распределениями вероятностей* $\mu = (\mu_i)_{i \in U}$ *и* $\nu = (\nu_i)_{i \in U}$ *на конечном пространстве* U *определяется правилом:* $\|\mu - \nu\|_{TV} = \frac{1}{2} \cdot \sum_{i \in U} |\mu_i - \nu_i|$.

Это расстояние является половиной метрики l_1, следовательно, само является метрикой (в частности, симметрично).

Лемма

$\|\mu - \nu\|_{TV} = \boldsymbol{max}_{R \subseteq U} |\mu(R) - \nu(R)|$.

В этой лемме подмножество R, на котором достигается максимум, определяется так: $R = \{i \in U \mid \mu_i > \nu_i\}$.

Приложение 2: Остановленная цепь-3

Лемма

$\boldsymbol{P}[T > \sum_{j=1}^{k} T_j] \leq \boldsymbol{P}[T > \sum_{j=1}^{k-1} T_j] \cdot \boldsymbol{P}[T > T_k]$ *для всех* $1 < k \leq r$.

Это следует из формулы условной вероятности, так как если $T > \sum_{j=1}^{k-1} T_j$, то, применяя ко всем четырем кандидатам
$Min \leq \langle A_1(\sum_{j=1}^{k-1} T_j), B_1(\sum_{j=1}^{k-1} T_j)\rangle < \langle A_2(\sum_{j=1}^{k-1} T_j), B_2(\sum_{j=1}^{k-1} T_j)\rangle \leq Max$
одинаковые операции *CbODown* и *CbOUp*, имеем, что если
$\langle A_1(t + \sum_{j=1}^{k-1} T_j), B_1(t + \sum_{j=1}^{k-1} T_j)\rangle < \langle A_2(t + \sum_{j=1}^{k-1} T_j), B_2(t + \sum_{j=1}^{k-1} T_j)\rangle$
склеивается позднее момента $T_k + \sum_{j=1}^{k-1} T_j = \sum_{j=1}^{k} T_j$, то и склеивание $Min < Max$ совершается позднее момента T_k.

Лемма

$\|\mu - \mu_{\hat{T}}\|_{TV} \leq \frac{\boldsymbol{P}[T > \hat{T}]}{1 - \boldsymbol{P}[T > \hat{T}]}$, *где* $\mu_{\hat{T}}$ - *распределение остановленной на верхней границе* $\hat{T}$ *склеивания по* $r > 1$ *испытаниям, а* μ - *распределение выдачи неостановленной цепи.*

Приложение 2: Остановленная цепь-4

Из определения $T, T_1, \ldots, T_r$ как независимых одинаково распределенных случайных величин, следует, что $\boldsymbol{P}[T > T_j] \leq \frac{1}{2}$ для всех $1 \leq j \leq r$.

Лемма

$\|\mu - \mu_{\hat{T}}\|_{TV} \leq \frac{2^{-r}}{1-2^{-r}} = \frac{1}{2^r-1}$, *где* $\mu_{\hat{T}}$ - *распределение остановленной на верхней границе склеивания по* $r > 1$ *испытаниям, а* μ - *распределение выдачи неостановленной цепи.*

Теорема

Для любого $R \subseteq U$ *с* $\mu(R) = \rho$ *и* $r > log_2(\rho + 1) - log_2(\rho)$ *имеем* $\mu_{\hat{T}}(R) \geq \rho - \frac{1}{2^r-1}$ *для верхней границы* $\hat{T}$ *склеивания по* $r > 1$ *испытаниям.*