ФЕДЕРАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР
«ИНФОРМАТИКА И УПРАВЛЕНИЕ» РОССИЙСКОЙ АКАДЕМИИ
НАУК

На правах рукописи

Масляков Глеб Олегович

Корректная классификация над произведением частичных порядков

Специальность 1.2.3 —
«Теоретическая информатика, кибернетика»

Диссертация на соискание учёной степени
кандидата физико-математических наук

Научный руководитель:
д.ф.-м.н., доцент
Дюкова Елена Всеволодовна

Москва — 2023

Оглавление

Введение

Настоящая работа нацелена на создание алгоритмического обеспечения для задач интеллектуального анализа данных, ориентированного на использование аппарата дискретной математики.

Рассматривается одна из центральных задач машинного обучения, а именно задача классификации на основе прецедентов. Под прецедентной (обучающей) информацией понимается совокупность примеров изучаемых объектов, в которой каждый объект представлен в виде числового вектора, полученного на основе измерения или наблюдения ряда его параметров или характеристик, называемых признаками. Каждый пример (обучающий объект или прецедент) приписан к определённому классу объектов. Требуется на основе анализа обучающей информации построить алгоритм, позволяющий классифицировать новые, не входящие в обучающую выборку, объекты.

Основное достоинство логического подхода к задаче классификации на основе прецедентов — возможность получения результата при отсутствии дополнительных предположений вероятностного характера и при небольшом числе прецедентов. Считается, что каждый признак принимает ограниченное число допустимых значений, которые кодируются целыми числами. Для каждого признака задаётся бинарная функция близости между его значениями, что позволяет проводить сравнение описания распознаваемого объекта с описаниями прецедентов. Анализ прецедентной информации сводится к поиску в исходных данных специальных фрагментов описаний объектов, различающих объекты из разных классов. Найденные фрагменты имеют содержательное описание в терминах той прикладной области, в которой решается задача. По их наличию или, наоборот, отсутствию в описании распознаваемого объекта, решается вопрос о его классификации. Большое внимание уделяется вопросам синтеза алгоритмов, безошибочно классифицирующих материал обучения. Такие алгоритмы называются корректными.

К наиболее известным направлениям логической классификации относятся Correct Voting Procedures или CVP (процедуры корректного голосования), предложенные впервые в отечественных работах [1–38, 42, 45, 55–66], а также Logical Analysis of Data или LAD (логический анализ данных) [35, 41, 43, 49, 51, 54, 75, 86, 87] и Formal Concept Analysis или FCA (анализ формальных понятий) [39, 40, 44, 48, 71, 76, 78–83, 91–93]. Все три названных направления CVP, LAD и FCA имеют много общего. С другой стороны, каждый из подходов использует свою терминологию и демонстрирует некоторую оригинальность.

Пусть исследуемое множество объектов M представимо в виде объединения попарно не пересекающихся подмножеств (классов) $K_1, \ldots, K_l$, и пусть объекты из множества M описываются целочисленными признаками $x_1, \ldots, x_n$. Описание объекта S из M имеет вид $(a_1, \ldots, a_n)$, здесь a_j — значение признака x_j для объекта S.

Классические процедуры CVP базируются на поиске фрагментов описаний прецедентов, называемых корректными элементарными классификаторами. Элементарным классификатором **ЭК** называется пара (σ, H), где $H = \{x_{j_1}, \ldots, x_{j_r}\}$ — набор различных признаков, $\sigma = (\sigma_1, \ldots, \sigma_r)$ — набор, в котором σ_i значение признака x_{j_i}, $i = 1,2,\ldots,r$. Близость между объектом $S = (a_1, \ldots, a_n)$ и **ЭК** (σ, H) оценивается функцией $B(S, \sigma, H)$, которая принимает значение 1, если $a_{j_i} = \sigma_i$ для всех $i = 1,2,\ldots,r$, и 0 в противном случае. Если $B(S, \sigma, H) = 1$, то говорят, что объект S содержит **ЭК** (σ, H). **ЭК** (σ, H) корректен для класса K, $K \in \{K_1, \ldots, K_l\}$, если нельзя указать пару прецедентов, одновременно содержащих этот **ЭК**, причём один из них принадлежит классу K, а другой не принадлежит классу K.

В общем случае корректный **ЭК** (σ, H), по отношению к классу K может обладать одним из следующих двух свойств:

1) некоторые обучающие объекты из класса K содержат (σ, H);
2) ни один обучающий объект из класса K не содержит (σ, H).

Корректный **ЭК** первого типа называется *представительным* **ЭК** класса K. Корректный **ЭК** второго типа называется *покрытием* класса K.

На этапе обучения для каждого класса K алгоритм A строит некоторое множество $C^A(K)$ корректных **ЭК** класса K. При классификации произвольного объекта S из M найденные **ЭК** участвуют в процедуре голосования с целью вычисления общей оценки принадлежности распознаваемого объекта этому классу. В модели голосования по представительным **ЭК** множество $C^A(K)$ состоит из представительных **ЭК** класса K (необязательно всех). Оценка $\Gamma_1(S,K)$ принадлежности объекта S к классу K вычисляется по формуле

$$\Gamma_1(S,K) = \frac{1}{W} \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)} B(S,\sigma,H),$$

здесь $P(\sigma,H)$ — вес **ЭК** (σ,H), $W = \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)}$. В качестве $P(\sigma,H)$ обычно берется число обучающих объектов из K, содержащих (σ,H). Объекту S присваивается класс с наивысшей оценкой. Если таких оценок несколько, то происходит отказ от классификации.

Аналогично устроена модель голосования по покрытиям класса. Однако в данной модели оценка $\Gamma_2(S,K)$ принадлежности объекта S к классу K, вычисляется по формуле

$$\Gamma_2(S,K) = \frac{1}{W} \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)} \left(1 - B(S,\sigma,H)\right).$$

В CVP в основном используются модели голосования по тупиковым корректным **ЭК** классов. Корректный для класса K **ЭК** (σ,H) называется *тупиковым*, если не является корректным любой **ЭК** (σ,H) такой, что $\sigma' \subset \sigma$, $H' \subset H$. Тупиковые корректные **ЭК** считаются наиболее информативными.

Направление LAD ориентировано на поиск так называемых логических закономерностей или patterns. Логическая закономерность класса K это представительный для класса K **ЭК**. Наиболее информативными считаются логические закономерности, оптимальные с точки зрения некоторого заранее выбранного функционала. Например, ищутся логические закономерности, содержащиеся в наибольшем числе прецедентов (наибольшие логические закономерности) [49].

В направлении FCA для задачи классификации ключевым термином является положительная ДСМ-гипотеза [44]. Каждая положительная для класса K ДСМ-гипотеза порождает представительный **ЭК** (σ, H) класса K, обладающего следующим свойством: любой представительный **ЭК** (σ', H') класса K такой, что $\sigma \subset \sigma'$, $H \subset H'$, содержится в меньшем числе прецедентов.

Таким образом, алгоритмы CVP, LAD и FCA ориентированы на поиск представительных **ЭК**, но каждое направление по разному определяет информативность **ЭК**. Это обуславливает различие в методологии поиска требуемых представительных **ЭК**.

Направление CVP опирается на теорию труднорешаемых перечислительных задач [1, 7–13, 16–21, 29, 31, 42, 46, 55, 56]. Алгоритмы LAD в значительной степени используют методы теории целочисленного программирования и при этом, как правило, строят небольшое количество представительных **ЭК**, что позволяет лучше интерпретировать полученные результаты. Схема работы классифицирующего алгоритма в LAD полностью аналогична схеме работы алгоритма CVP.

В [44] В. К. Финн предложил так называемый метод автоматического порождения гипотез (или ДСМ-метод), который позднее в 1990-х годах был адаптирован В. К. Финном и его учениками для задач машинного обучения (см., например, [48]). В [40, 71, 82] С. О. Кузнецов описал ДСМ-метод в терминах FCA.

ДСМ-классификатор действует более строго по сравнению с классификаторами из CVP и LAD. На первом этапе для каждого класса K строится некоторое множество $C^A(K)$ представительных **ЭК** класса K, порождаемых положительными для класса K гипотезами. Объект S относится к классу K, если S содержит хотя бы один **ЭК** из $C^A(K)$, и не содержит ни одного **ЭК** из $C^A(K')$, $K' \neq K$. В противном случае происходит отказ от классификации.

Настоящая работа посвящена развитию методов направления CVP.

Первые процедуры CVP, а именно, тестовые алгоритмы и алгоритмы с представительным наборами (или алгоритмы типа «Кора») предложены в [3–6, 13, 37], их описание с использованием понятий **ЭК** впервые приведено в [28]. Тестовый алгоритм базируется на понятии теста, введённого С.В. Яблонским и И.А. Чегис в [45] для решения проблемы выявления неисправности контактных схем.

На практике встречаются задачи, когда не удается найти достаточное количество информативных корректных **ЭК**. Подобная ситуация возникает, например, в случае целочисленных данных высокой значности (под значностью признака понимается число его различных значений). Существуют различные способы решения этой проблемы.

Один из этих способов заключается в выполнении «корректной» перекодировки исходных признаков с целью понизить их значность [27]. В результате такой перекодировки объекты из разных классов остаются различимыми. Построение наилучшей в смысле качества распознавания корректной перекодировки — сложная оптимизационная задача.

Другой способ основан на идеях алгебро-логического подхода [34]. Этот подход базируется на использовании произвольных **ЭК** (не обязательно корректных) и объединяет идеи логического и алгебраического подходов. Об алгебро-логической коррекции говорят, когда каждый базовый алгоритм однозначно определяется некоторым **ЭК** (не обязательно корректным) и корректирующая функция является булевой. В логических корректорах оценки принадлежности распознаваемого объекта к классам вычисляются на основе процедуры голосования по (тупиковым) корректным наборам **ЭК**.

При поиске тупиковых корректных **ЭК** и тупиковых корректных наборов **ЭК** возникает необходимость рассматривать сложные в вычислительном плане задачи, которые в теории алгоритмической сложности дискретных задач называют труднорешаемыми [77]. Среди этих задач центральное место принадлежит монотонной дуализации — задаче построения сокращённой дизъюнктивной нормальной формы монотонной булевой функции, заданной конъюнктивной нормальной формой [1, 7, 70]. Задача допускает гиперграфовую формулировку [84] и матричную формулировку с использованием понятия неприводимого покрытия булевой матрицы [10–13, 29–33]. Труднорешаемость монотонной дуализации имеет два аспекта: экспоненциальный рост числа решений при увеличении размера задачи и сложность их нахождения (перечисления). Наиболее эффективными считаются алгоритмы с полиномиальным шагом (алгоритмы с полиномиальной задержкой). Полиномиальные алгоритмы построены лишь для некоторых частных случаев монотонной дуализации (см., например, [68]). В настоящее время сформировано два основных направления исследований.

Первое направление нацелено на построение так называемых инкрементальных алгоритмов монотонной дуализации, когда алгоритму разрешено просматривать решения, найденные на предыдущих шагах. При этом оценка сложности шага алгоритма даётся для худшего случая (для самого сложного варианта задачи). В [70] построен инкрементальный алгоритм монотонной дуализации с квазиполиномиальным шагом, определяемым фактически не только размером входа задачи, но и размером её выхода.

Второе направление основано на построении асимптотически оптимальных алгоритмов дуализации, впервые предложенных в [8]. В этом случае алгоритму разрешено делать лишние полиномиальные шаги при условии, что их число почти всегда должно быть достаточно мало по сравнению с числом всех решений задачи. В рамках данного направления удалось построить алгоритмы монотонной дуализации, эффективные в типичном случае (эффективные для почти всех вариантов задачи). Асимптотически оптимальные алгоритмы используют матричную формулировку задачи монотонной дуализации и являются лидерами по скорости счёта. Существуют обобщения задачи монотонной дуализации на целочисленный случай, основанные на модификации понятия неприводимого покрытия булевой матрицы, для которых в [10, 13] построены асимптотически оптимальные алгоритмы.

Современные прикладные задачи классификации не всегда могут быть описаны в рамках классической постановки, когда отдельные значения признака сравниваются использованием простых отношений «равно» и «не равно». В ряде случаев возникает необходимость рассматривать более сложные отношения на множествах допустимых значений признаков. Например, когда на этих множествах заданы отношения частичных порядков и описания объектов представляют собой элементы декартова произведения конечных частично упорядоченных множеств.

Пусть $M = N_1 \times \cdots \times N_n$, где N_i — частично упорядоченное множество значений признака x_i, $i \in \{1,2,\ldots,n\}$. Запись $a \preceq b$ ($b \succeq a$) означает, что b следует за a. Элементы a, b из частично упорядоченного множества N_i называются сравнимыми, если a следует за b или b следует за a. В противном случае a и b несравнимы. Если все элементы множества N_i попарно сравнимы, то множество N_i называется *линейно* упорядоченным или *цепью*. Если все различные элементы множества N_i попарно несравнимы, то множество N_i называется *антилинейно* упорядоченным или *антицепью*. Обозначим $a \prec b$, если $a \preceq b$ и $a \neq b$. Объект $S = (a_1, \dots, a_n) \in M$ следует за объектом $S' = (b_1 \dots b_n) \in M$, если a_i следует за b_i при $i = 1{,}2, \dots, n$. Таким образом, на множестве объектов из M естественным образом возникает отношение частичного порядка.

Введём обобщённую функцию близости $\tilde{B}(S, \sigma, H)$ между объектом $S = (a_1, \dots, a_n)$ и **ЭК** (σ, H), которая принимает значение 1, если $a_{j_i} \preceq \sigma_i$, $i = 1{,}2, \dots, r$, и 0 в противном случае. При построении процедур классификации, учитывающих частичную упорядоченность данных, требуется ввести более общие понятия корректного **ЭК**, представительного **ЭК** и покрытия класса, используя обобщённую функцию близости. Актуальным является построение асимптотически оптимальных методов поиска корректных **ЭК** общего вида, опирающихся на получение асимптотических оценок типичного числа и типичного ранга таких ЭК.

Как правило, результат классификации существенно зависит от того, какие частичные порядки заданы на множествах значений признаков. При этом выбор частичных порядков, обеспечивающих высокое качество классификации, путём полного перебора возможных вариантов бесперспективен в силу колоссальной вычислительной сложности. Важными являются вопросы синтеза вычислительно эффективных процедур выбора частичных порядков на множествах значений признаков, гарантирующих корректную классификацию.

Хорошо известно, что использование алгоритмов стохастических композиций приводит к улучшению качества классификации и повышению скорости обучения логических классификаторов. На данный момент, сильнейшими алгоритмами классификации являются стохастические композиции над решающими деревьями, такие как CatBoost или LGBM [67, 74]. Поэтому актуальным является создание стохастических композиций над логическими классификаторами в случае, когда информация представлена в виде декартова произведения конечных частично упорядоченных множеств.

Целью данной работы является обобщение процедур CVP на случай, когда данные представляют собой элементы декартова произведения конечных частично упорядоченных множеств. В классическом варианте порядок на множестве прецедентов фактически не установлен, так как отдельные значения каждого признака несравнимы между собой.

Рассмотрены следующие конкретные **задачи**.

1. Описание в рамках терминологии CVP единой схемы синтеза алгоритмов логической классификации, включающей все три названных направления, а именно CVP, LAD и FCA.
2. Обобщение классических понятий CVP с целью создания схемы классификации для процедур CVP в случае частично упорядоченных данных. Разработка алгоритмов поиска корректных **ЭК** на основе эффективного решения задачи дуализации над произведением частичных порядков.
3. Постановка задачи дуализации над произведением частичных порядков в матричном виде. Получение асимптотических оценок типичных значений количественных характеристик множества решений задачи дуализации над произведением цепей. Построение асимптотически оптимального алгоритма дуализации над произведением цепей.
4. Разработка и исследование эффективных методов задания частичных порядков на множествах значений признаков, обеспечивающих высокое качество классификации.
5. Создание стохастических композиций построенных процедур классификации над произведением частичных порядков.

Научная новизна. Впервые создана единая схема синтеза логических процедур классификации по прецедентам, включающая направления CVP, LAD и FCA, и для направления CVP построены корректные логические классификаторы над произведением частичных порядков. Развит асимптотически оптимальный поход к труднорешаемым перечислительным дискретным задачам, возникающим на этапе обучения построенных классификаторов. Поставлена задача выбора «корректных» частичных порядков (гарантирующих корректность классификации) на множествах значений признаков, и намечены пути её решения. Предложена быстрая процедура выбора «некорректных» линейных порядков на множествах значений признаков. Разработаны и экспериментально исследованы практические модели стохастических композиций логических классификаторов над произведением частичных порядков. Все полученные результаты являются новыми.

Теоретическая и практическая значимость. Диссертационная работа содержит как теоретические, так и практические результаты.

Рассмотрены методологические аспекты алгоритмов логической классификации. Показано, что известные алгоритмы логической классификации, описанные в традициях разных научных направлений, могут быть синтезированы в рамках единой схемы.

Создана общая схема синтеза корректных логических алгоритмов классификации по прецедентам над произведением частичных порядков, согласно которой классические модели корректного голосования — это классификаторы над произведением антицепей. Показано, что в случае представления данных в виде декартова произведения конечных частичных порядков синтез процедур CVP, сводится к решению задачи дуализации над произведением частичных порядков.

Дана матричная формулировка задачи дуализации над произведением частичных порядков, в рамках которой установлено, что в общем случае анализ прецедентной информации приводит к необходимости находить так называемые упорядоченные тупиковые покрытия целочисленной матрицы. На основе изучения метрических (количественных) свойств множества упорядоченных тупиковых покрытий целочисленной матрицы получена асимптотика типичного числа решений задачи дуализации над произведением цепей, и для этой задачи разработан асимптотически оптимальный алгоритм. Построены и реализованы асимптотически оптимальные классификаторы над произведением конечных цепей.

Исследована актуальная и ранее не изучавшаяся задача выбора на этапе предварительного анализа обучающей выборки «хороших» частичных порядков. Разработана «быстрая» процедура линейного упорядочения множеств допустимых значений признаков, эффективная по времени вычислений и позволяющая повысить качество классификации, но не гарантирующая корректность классификации. Поставлена задача выбора «корректных» частичных порядков на множествах допустимых значений признаков, и показано, что эта задача может быть решена на основе построения неприводимых покрытий специальной булевой матрицы.

На базе идей бэггинга и бустинга построены и экспериментально исследованы стохастические композиции логических классификаторов, использующих голосование по тупиковым представительным **ЭК** общего вида.

Степень достоверности полученных результатов обеспечивается доказательствами сформулированных утверждений и теорем, а также результатами экспериментов, проведённых автором.

Апробация работы. Основные результаты работы докладывались на конференциях «Математические методы распознавания образов (ММРО-18)» (Таганрог, 2017), «Дискретные модели в теории управляющих систем» (Подмосковье, 2018), «Математические методы распознавания образов (ММРО-19)» (Москва, 2019), «Конференция по искусственному интеллекту (КИИ-2019)» (Ульяновск, 2019), «Интеллектуализация обработки информации (ИОИ-13)» (Москва, 2020), «Информационные технологии и нанотехнологии (ИТНТ-2021)» (Самара, 2021), «Математические методы распознавания образов (ММРО-20)» (Москва, 2021), «Информационные технологии и нанотехнологии (ИТНТ-2023)» (Самара, 2023).

Публикации. По тематике работы опубликовано 15 научных работ [2, 16–26, 56, 57, 60], при этом статьи [17, 22, 24] имеют англоязычные версии [55, 58, 59]. В журналах, рекомендованных ВАК опубликованы 4 работы [17, 21, 22, 24]. В журналах, индексируемых в Web Of Science Core Collection, опубликованы 2 статьи [55, 58]. В журналах, индексируемых в Scopus, опубликованы 4 статьи [56, 57, 59, 60]. Указанные статьи подготовлены в рамках участия в проектах РФФИ №16-01-00445 и №19-01-00430.

Работа состоит из введения, четырёх глав, заключения и списка литературы.

Во введении сформулированы основные цели и задачи, описаны основные результаты и структура диссертационной работы.

В первой главе приведено описание единой схемы синтеза алгоритмов логической классификации, включающей направления CVP, LAD и FCA. Показано, что каждое направление вводит специальный частичный порядок на множестве $\mathcal{P}(K)$ представительных **ЭК** класса K и в качестве наиболее информативных **ЭК** рассматривает максимальные относительно заданного порядка элементы множества $\mathcal{P}(K)$. Элемент частично упорядоченного множества называется *максимальным*, если за ним не следует ни один другой элемент из этого множества.

В частности, для процедур голосования по тупиковым представительным **ЭК** на множестве $\mathcal{P}(K)$ задаётся отношение частичного порядка $\preceq_1$, согласно которому **ЭК** $(\sigma_1, H_1) \in \mathcal{P}(K)$ следует за **ЭК** $(\sigma_2, H_2) \in \mathcal{P}(K)$ (т.е. $(\sigma_2, H_2) \preceq_1 (\sigma_1, H_1)$), если $\sigma_1 \subseteq \sigma_2$, $H_1 \subseteq H_2$. Доказано следующее

Утверждение 1.5.1. ***ЭК*** *$(\sigma, H) \in \mathcal{P}(K)$ является тупиковым представительным для класса K тогда и только тогда, когда (σ, H) — максимальный относительно частичного порядка $\preceq_1$ элемент множества $\mathcal{P}(K)$.*

Аналогичные утверждения доказаны для направлений LAD и FCA.

Во второй главе дано описание схемы синтеза процедур CVP для случая частично упорядоченных данных.

Понятия корректного **ЭК** класса K, представительного **ЭК** класса K и покрытия класса K переносятся на рассматриваемый случай заменой функции близости $B(S, \sigma, H)$ между объектом S и **ЭК** (σ, H) на функцию $\tilde{B}(S, \sigma, H)$. При этом предполагается, что частично упорядоченное множество значений признаков N_i, $i = 1,2,\ldots,n$, содержит наибольший элемент k_i.

Пусть (σ,H) — **ЭК**, в котором $H = \{x_{j_1},\ldots,x_{j_r}\}$, $\sigma = (\sigma_1,\ldots,\sigma_r)$, $\sigma_i \in N_{j_i}$, $i = 1, 2, \ldots, r$. **ЭК** (σ,H) сопоставим набор $S_{(\sigma,H)} = (\gamma_1,\ldots,\gamma_n)$ из M, в котором $\gamma_t = \sigma_i$ при $t \in \{j_1, \ldots, j_r\}$, и $\gamma_t = k_t$ при $t \notin \{j_1, \ldots, j_r\}$.

Корректный для класса K **ЭК** назовём тупиковым, если любой другой **ЭК** (σ', H') такой, что $S_{(\sigma,H)} \preceq S_{(\sigma',H')}$, не является корректным **ЭК** класса K.

Через $R(K)$ обозначим множество прецедентов из класса K. $R(K)^+$ — множество объектов из M, которые следуют за хотя бы одним объектом из $R(K)$. Элемент $S \in M$ называется независимым от $R(K)$, если $S \in M \setminus R(K)^+$. Задача построения множества $I(R(K))$, содержащего все максимальные элементы множества $M \setminus R(K)^+$ известна как задача дуализации над произведением частичных порядков и относится к классу труднорешаемых.

Утверждение 2.2.1. *Покрытие* (σ, H) *класса* K *является тупиковым покрытием класса* K *тогда и только тогда, когда* $S(\sigma, H) \in I(R(K))$.

Пусть $\overline{K} = M \setminus K$. Будем рассматривать $\overline{K}$ как отдельный класс, т.е. будем считать, что есть всего два класса K и $\overline{K}$.

Утверждение 2.2.2. ***ЭК*** (σ, H) *является тупиковым представительным для класса* K *тогда и только тогда, когда* $S(\sigma, H) \in I(R(\overline{K}))$ *и* $S(\sigma, H) \in R(K)^+$.

Таким образом показано, что поиск тупиковых корректных **ЭК** общего вида сводится к решению задачи дуализации над произведением частичных порядков.

В общем случае существование представительных для класса K **ЭК** не гарантировано. Пусть $\tilde{\tilde{M}} = \tilde{\tilde{N}}_1 \times \cdots \times \tilde{\tilde{N}}_n$, $\tilde{\tilde{N}}_i$ совпадает с N_i, $i = 1, 2, \ldots, n$, но на $\tilde{\tilde{N}}_i$ задано обратное отношение порядка, т.е. $a \preceq b$ в $\tilde{\tilde{N}}_i$ тогда и только тогда, когда $b \preceq a$ в N_i.

Зададим отображение $\psi : M \to M \times \tilde{\tilde{M}}$ следующим образом. Отображение ψ переводит объект $S = (a_1, \ldots, a_n)$ из M в объект $\psi(S) = (a_1, \ldots, a_n, a_{n+1}, \ldots, a_{2n})$ из $M \times \tilde{\tilde{M}}$, в котором $a_{i+n} = a_i$ при $i \in \{1, 2, \ldots, n\}$, т.е. признаковое описание объекта S дублируется с обратным отношением порядка.

Пусть $\psi(A), A \subset M$, — образ A при отображении ψ. Имеет место следующая

Теорема 2.2.1. *Если классы множества* M *не пересекаются, то любой прецедент из класса* $\psi(K)$ *порождает тупиковый представительный* ***ЭК*** *класса* $\psi(K)$.

Согласно теореме 2.2.1 существует такое преобразование признакового описания множества M, которое обеспечивает корректность классификации.

Отметим, что этап обучения — это самый сложный в вычислительном плане этап логической классификации из-за необходимости решать задачу дуализации над произведением частичных порядков, число решений которой растёт экспоненциально с ростом размера входа задачи. Поэтому описанный метод преобразования признакового пространства применим только в случае небольшого числа признаков..

В данной главе дано описание практических моделей логической классификации над произведением частичных порядков, основанных на стохастической композиции над обобщёнными логическими классификаторами. Предлагаемые модели основаны на известных способах ансамблирования (бэггинг и бустинг), в которых в качестве базового классификатора использован алгоритм голосования по представительным **ЭК**. Эти модели не гарантируют корректность классификации, но демонстрируют высокое качество классификации на реальных задачах, что показало проведённое подробное экспериментальное исследование.

В третьей главе разработаны эффективные методы задания частичных порядков на множествах значений признаков, обеспечивающих корректность классификации. Описана быстрая процедура линейного некорректного упорядочения значений признаков и приведены результаты её тестирования на реальных задачах.

Пусть A — классификатор над произведением частичных порядков, строящий все тупиковые представительные **ЭК** класса K. Тогда справедлива

Теорема 3.1.1. *Алгоритм A классифицирует правильно объект S' из $R(K)$ тогда и только тогда, когда $S' \in M \setminus R(\overline{K})^+$.*

Частичный порядок на множестве M называется $(A{,}K)$-*корректным*, если алгоритм A правильно классифицирует каждый объект из $R(K)$. Построим булеву матрицу B_K. Каждой паре объектов $(S'{,}S'')$, где $S' \in R(K)$ и $S'' \in R(\overline{K})$, соответствует строка в матрице B_K. Каждому признаку x_j,

$j \in \{1,2,\ldots,n\}$, и каждой паре (a,b), $a \in N_j$, $b \in N_j$, $a \neq b$, соответствует столбец (x_j, a, b) матрицы B_K. Элемент матрицы B_K, расположенный на пересечении строки (S', S'') и столбца (x_j,a,b), равен 1, если значение признака x_j равно a и b у объектов S' и S'' соответственно.

Набор столбцов H матрицы B_K называется *покрытием*, если каждая строка матрицы B_K в пересечении хотя бы с одним из столбцов, входящих в H, дает 1. Покрытие матрицы B_K называется *неприводимым*, если любое его собственное подмножество покрытием не является.

С использованием утверждения теоремы 3.1.1 доказана

Теорема 3.3.1. *Частичный порядок, заданный на множестве M, является (A,K)-корректным тогда и только тогда, когда существует покрытие H матрицы B_K такое, что для любого $j \in \{1,2,\ldots,n\}$ и для любых $a,b \in N_j$, $b \prec a$, столбец (x_j,a,b) не входит в H.*

Частичный порядок на множестве объектов из M называется *линейным (антилинейным) на множестве M*, если каждое множество N_j, $j = 1, 2, \ldots, n$ является цепью (антицепью).

Рассмотрим (A, K)-корректный линейный порядок на множестве M. Согласно теореме 3.3.1 существует покрытие H матрицы B_K такое, что для любого столбца $(x_j, a, b) \in H$, $a,b \in N_j$, не выполнено $b \prec a$. Поскольку множество N_j является цепью, то $a \prec b$. Поэтому справедливо

Следствие 3.3.1. *Линейный порядок на множестве M является (A,K)-корректным тогда и только тогда, когда существует покрытие H матрицы B_K такое, что $a \prec b$ для любого столбца (x_j,a,b) из покрытия H.*

Пусть на множестве M задан (A, K)-корректный антилинейный порядок. Тогда для всех $j = 1, 2, \ldots, n$ и для всех $a,b \in N_j$ не выполнено $b \prec a$. Следовательно, для любого покрытия матрицы B_K выполнены условия теоремы 3.3.1 Поэтому справедливо

Следствие 3.3.2. *Антилинейный порядок на множестве M является (A,K)-корректным для любого класса K из $\{K_1, \dots, K_l\}$.*

С целью увеличения числа признаков с линейно упорядоченным множеством значений была разработана генетическая процедура поиска покрытия матрицы B_K, по длине близкого к минимальному. Необходимость использования генетического алгоритма обусловлена большими размерами матрицы B_K и её разреженностью по числу единиц.

Для линейного упорядочения множества значений отдельного признака использовался алгоритм топологической сортировки с линейным временем работы. В случае невозможности линейного упорядочения на множестве значений признака устанавливался антилинейный порядок.

Отметим, что каждое покрытие матрицы B_K может порождать несколько (A, K)-корректных частичных порядков, из-за чего описанная процедура выбора корректного частичного порядка имеет высокую степень неопределённости. Этого недостатка лишена описанная в данной главе процедура выбора линейного порядка на множестве M, основанная на оценке информативности значений отдельных признаков и не гарантирующая корректность классификации.

Пусть $\mu_{ij}^{(1)}(a)$ и $\mu_{ij}^2(a)$, $i \in \{1,2,\dots,l\}$, $j \in \{1,2,\dots,n\}$, $a \in N_j$, — соответственно доля прецедентов класса K_i и доля прецедентов не из класса K_i, у которых признак x_j принимает значение a. Величина $\mu_{ij}(a) = \mu_{ij}^1(a) - \mu_{ij}^2(a)$ служит мерой важности значения a признака x_j в классе K_i. Для каждого класса $K_i, i \in \{1,2,\dots,l\}$, и для каждого признака x_j, $j \in \{1,2,\dots,n\}$, зададим следующий линейный порядок: $\forall y,z \in N_j$, $y \preceq z$ тогда и только тогда, когда $\mu_{ij}(y) \geq \mu_{ij}(z)$. Описанная процедура обеспечивает высокое качество классификации и незначительное время счёта, что подтверждено результатами экспериментального исследования.

В четвёртой главе приведена матричная формулировка задачи дуализации над произведением частичных порядков. Показано, что данная задача сводится к поиску так называемых упорядоченных тупиковых покрытий целочисленной матрицы. Понятие упорядоченного тупикового покрытия целочисленной матрицы обобщает известное понятие тупикового покрытия целочисленной матрицы. Для дуализации над произведением цепей построен асимптотически оптимальный алгоритм RUNC-M+. Его теоретическое обоснование базируется на приведённой ниже теореме 4.2.1.

Введем обозначения: M^k_{mn} — совокупность всех матриц размера $m \times n$ с элементами из $\{0, 1, \ldots, k-1\}$, $k \geq 2$; E^r_k, $r \leq n$, — множество всех наборов вида $(\sigma_1, \ldots, \sigma_r)$, в которых $\sigma_i \in \{0, 1, \ldots, k-1\}$, $k \geq 2$, при $i = 1, 2, \ldots, r$. Рассмотрим $\sigma \in E^r_k$, $\sigma = (\sigma_1, \ldots, \sigma_r)$, $\sigma_i < k-1$, $i = 1, 2, \ldots, r$. Через $Q_i(\sigma)$, $i \in \{1, 2, \ldots, r\}$, обозначим множество наборов $(\beta_1, \ldots, \beta_r)$ в E^r_k, таких что $\beta_i = \sigma_i + 1$ и $\beta_j \leq \sigma_j$ при $j \in \{1, 2, \ldots, r\} \setminus \{i\}$.

Пусть H — набор из r различных столбцов матрицы $L \in M^k_{mn}$. Множество различных строк подматрицы матрицы L, образованной столбцами набора H, можно рассматривать как некоторое подмножество E^H наборов из E^r_k. Набор столбцов H называется упорядоченным тупиковым σ-покрытием матрицы L, если выполнены два следующих условия:

1) E^H не содержит набор $(\beta_1, \ldots, \beta_r) \in E^r_k$, в котором $\beta_j \leq \sigma_j$ при $j \in \{1, 2, \ldots, r\}$;
2) если $i \in \{1, 2, \ldots, r\}$, то E^H содержит хотя бы один набор из $Q_i(\sigma)$.

Если $L \in M^2_{mn}$ и набор столбцов H является упорядоченным тупиковым $(0, 0, \ldots, 0)$-покрытием матрицы L, то H — неприводимое покрытие матрицы L.

Квадратную подматрицу порядка r матрицы $L \in M^k_{mn}$ назовем упорядоченной σ-подматрицей, если для множества E^H, выполнено $E^H \cap Q_i(\sigma) \neq \varnothing$ при $i \in \{1, 2, \ldots, r\}$.

Обозначим: ϕ_d, $d > 0$, — интервал $(\frac{1}{2}\log_d mn - \frac{1}{2}\log_d \log_d mn - \log_d \log_d \log_d n, \frac{1}{2}\log_d mn - \frac{1}{2}\log_d \log_d mn + \log_d \log_d \log_d n)$; $\Pi_r(\sigma) = (\sigma_1 + 1)^{r-1} \ldots (\sigma_r + 1)^{r-1}, \sigma \in E^r_{k-1}$.

Пусть $L \in M^k_{mn}$, $\sigma \in E^r_{k-1}$. Положим $B(L, \sigma)$ — множество всех упорядоченных тупиковых σ-покрытий матрицы L; $S(L, \sigma)$ — множество всех упорядоченных σ-подматриц матрицы L;

$$\Sigma_1(L) = \sum_{r=1}^{n} \sum_{\sigma \in E^r_{k-1}} |B(L, \sigma)|;$$

$$\Sigma_2(L) = \sum_{r=1}^{n} \sum_{\sigma \in E^r_{k-1}} |S(L, \sigma)|.$$

Теорема 4.2.1 *Если $m^\alpha \le n \le d^m$, $\alpha > 1$, $d = k/(k-1)$, то для почти всех матриц L из M^k_{mn} при $n \to \infty$ справедливо*

$$\Sigma_1(L) \sim \Sigma_2(L) \sim \sum_{r\in\phi_d} \sum_{\sigma\in E^r_{k-1}} \Pi_r(\sigma) C^r_n C^r_m r! k^{-r^2}$$

и длины почти всех упорядоченных тупиковых покрытий матрицы L принадлежат интервалу ϕ_d.

Из теоремы 4.2.1 следуют оценки типичных значений количественных характеристик множества неприводимых покрытий булевой матрицы:

Следствие 4.2.1. *Если $m^\alpha \le n \le 2^m$, $\alpha > 1$, то для почти всех матриц L из M^2_{mn} при $n \to \infty$ справедливо*

$$\Sigma_1(L) \sim \Sigma_2(L) \sim \sum_{r\in\phi_2} C^r_n C^r_m r! 2^{-r^2},$$

и длины почти всех неприводимых покрытий матрицы L принадлежат интервалу ϕ_2.

В заключении приводятся положения диссертации, выносимые на защиту и задаются направления дальнейших исследований.

Список литературы включает 93 публикации.

Глава 1. Логический подход в задаче классификации по прецедентам

Рассматривается задача классификации по прецедентам. Исследуется некоторое множество объектов M. Известно, что M представимо в виде объединения l попарно не пересекающихся подмножеств $K_1, \ldots, K_l$, именуемых классами. Объекты из множества M описываются признаками $x_1, \ldots, x_n$, каждый из которых является некоторой наблюдаемой или измеряемой характеристикой этих объектов и имеет ограниченное число допустимых значений. Такие признаки называются целочисленными. Имеется конечный набор $S_1, \ldots, S_m$ объектов из множества M, о которых известно, каким классам они принадлежат. Это прецеденты или обучающие объекты. Пусть их описания имеют вид $S_1 = (a_{11}, \ldots, a_{1n})$, $S_2 = (a_{21}, \ldots, a_{2n}), \ldots,$ $S_m = (a_{m1}, \ldots, a_{mn})$, здесь a_{ij} — значение признака x_j для объекта S_i. Требуется по предъявленному набору значений признаков $(a_1, \ldots, a_n)$, описывающему некоторый объект S из множества M, о котором, вообще говоря, не известно, какому классу он принадлежит, определить (распознать) этот класс.

В данной главе приводится постановка задачи классификации по прецедентам и даётся обзор трёх основных подходов к построению логических алгоритмов классификации, а именно CVP, LAD и FCA. Каждый из указанных подходов имеет собственную терминологию. Для повышения наглядности сходства и различия описания подходов даны с использованием понятий, характерных для подхода, базирующегося на построении процедур корректного голосования.

Даётся общая схема описания логических классификаторов, согласно которой каждый логический классификатор на этапе обучения задаёт частичный порядок или предпорядок на множестве представительных **ЭК**, и ищет максимальные относительно заданного предпорядка элементы этого множества.

Основные результаты, представленные в этой главе, опубликованы в [60].

1.1 Основные понятия

Введем основные понятия, используемые при синтезе процедур CVP.

Пусть H — набор из r различных признаков вида $H = \{x_{j_1}, \ldots, x_{j_r}\}$, $\sigma = (\sigma_1, \ldots, \sigma_r)$, σ_i — допустимое значение признака x_{j_i}, $i = 1, 2, \ldots, r$. Пара (σ, H) называется *элементарным классификатором* (**ЭК**) ранга r.

Близость объекта $S = (a_1, \ldots, a_n)$ из множества M и **ЭК** (σ, H), $\sigma = (\sigma_1, \ldots, \sigma_r)$, $H = (x_{j_1}, \ldots, x_{j_r})$, оценивается величиной $B(\sigma, S, H)$, равной 1, если $a_{j_t} = \sigma_t$ при $t = 1, 2, \ldots, r$, и равной 0 в противном случае. Если $B(\sigma, S, H) = 1$, то говорят, что объект S *порождает (содержит)* **ЭК** (σ, H).

ЭК (σ, H) является *корректным* для класса K, $K \in \{K_1, \ldots, K_l\}$, если нельзя указать пару обучающих объектов S' и S'' таких, что $S' \in K$, $S'' \notin K$ и $B(\sigma, S', H) = B(\sigma, S'', H) = 1$. Корректный **ЭК** (σ, H) называется *тупиковым*, если любой **ЭК** вида (σ', H') такой, что $\sigma' \subset \sigma$, $H' \subset H$, не является корректным для класса K.

Таким образом, класс K не имеет корректных **ЭК**, если существуют два прецедента S' и S'' таких, что $S' \in K$, $S'' \notin K$ и описание S' совпадает с описанием S''. Поэтому предполагается, что любые два класса множества M не пересекаются.

Набор U, состоящий из произвольных **ЭК** (необязательно корректных), называется *корректным* для класса K, если нельзя указать пару обучающих объектов S' и S'' таких, что $S' \in K$, $S'' \notin K$ и $B(\sigma, S', H) = B(\sigma, S'', H)$ для любого **ЭК** (σ, H) из набора U.

Множество прецедентов из класса K и не из класса K обозначаются соответственно через $R(K)$ и $R(\overline{K})$.

1.2 Процедуры корректного голосования (CVP)

Логические классификаторы, ориентированные на построение процедур корректного голосования можно условно разделить на два типа. Во-первых, это классификаторы, основанные на голосовании по корректным **ЭК**, и, во-вторых, классификаторы, основанные на голосовании по корректным наборам **ЭК** или логические корректоры.

Считается, что каждый классифицирующий алгоритм A на этапе обучения строит для каждого класса K некоторое множество корректных **ЭК** $C^A(K)$. Классификация объекта S осуществляется на основе вычисления величины $B(\sigma, S, H)$ для каждого построенного **ЭК** (σ, H), т.е. каждый элемент множества $C^A(K)$, $K \in \{K_1, \ldots, K_l\}$, участвует в процедуре голосования. В результате для класса K вычисляется оценка $\Gamma(S, K)$ принадлежности объекта S классу K. Таким образом, алгоритм A из рассматриваемого семейства распознающих алгоритмов определяется множествами $C^A(K_1), \ldots, C^A(K_l)$. Алгоритмы отличаются и способом вычисления оценки $\Gamma(S, K)$. Рассмотрим основные модели.

В общем случае **ЭК** (σ, H), по отношению к классу K может обладать одним из следующих двух свойств:

1) некоторые обучающие объекты из класса K содержат (σ, H);
2) ни один обучающий объект из класса K не содержит (σ, H).

В первом случае считается, что распознаваемый объект S принадлежит классу K, если S содержит **ЭК** (σ, H). Во втором случае S принадлежит классу K, если S не содержит **ЭК** (σ, H).

Корректный **ЭК** первого типа называется *представительным* **ЭК** класса К. Представительный для класса K **ЭК** (σ, H) называется *тупиковым* представительным **ЭК** класса K, если он тупиковый корректный для класса K.

Понятие **ЭК**, представительного **ЭК** и тупикового представительного **ЭК** может быть дано в терминах теории логических функций. Поясним сказанное на примере бинарных признаков.

В рассматриваемом случае **ЭК** (σ, H), $H = \{x_{j_1}, \ldots, x_{j_r}\}$, $\sigma = (\sigma_1, \ldots, \sigma_r)$, — это элементарная конъюнкция B над переменными

$x_1, \ldots, x_n$ вида $x_{j_1}^{\sigma_1} \wedge \cdots \wedge x_{j_r}^{\sigma_r}$, которая обращается в 1 на описании объекта S, если объект S содержит **ЭК** (σ, H), т.е. описание объекта S принадлежит интервалу истинности N_B конъюнкции B.

Пусть $f_K(x_1, \ldots, x_n)$ — частичная булева функция, определенная на описаниях объектов обучающей выборки и принимающая значение 1 только на описаниях обучающих объектов из K.

Пусть $N_{\overline{f}_K}$ — множество булевых наборов, на которых f_K принимает значение 0. ЭК B называется *допустимой* для f_K, если $N_B \cup N_{\overline{f}_K} = \varnothing$, и $N_B \cup N_{f_K} \neq \varnothing$. ЭК B называется *максимальной* для функции f_K, если не существует допустимой для f_K ЭК B' такой, что $N_B \subset N_{B'}$.

Нетрудно видеть, что представительный **ЭК** класса K — это допустимая конъюнкция для функции f_K. Тупиковый представительный **ЭК** для класса K — это максимальная конъюнкция для f_K.

Набор признаков H называется *тестом*, если в каждом классе K, $K \in \{K_1, \ldots, K_l\}$, каждый прецедент содержит представительный для класса K **ЭК** вида (σ, H). Тест называется *тупиковым*, если любое его собственное подмножество тестом не является.

ЭК второго типа называется *покрытием* класса K. Покрытие (σ, H) класса K называется *тупиковым* покрытием класса K, если не является покрытием для K любой **ЭК** вида (σ', H'), где $\sigma' = \sigma_1, \ldots, \sigma_{t-1}, \sigma_{t+1}, \ldots, \sigma_r)$, $H' = H \setminus \{x_{j_t}\}$, $t \in \{1, 2, \ldots, r\}$.

Переформулируем понятие представительного **ЭК** класса K, используя понятие покрытия класса. Пусть $\overline{K} = M \setminus K$. Будем рассматривать множество объектов из $\overline{K}$ как отдельный класс, т.е. будем считать, что у нас всего два класса K и $\overline{K}$. Тогда, как нетрудно видеть, (тупиковый) представительный **ЭК** класса K является (тупиковым) покрытием для класса K и не является покрытием для класса $\overline{K}$.

Модели тестовых алгоритмов и алгоритмов голосования по представительным **ЭК** основаны на построении для каждого класса K множества представительных **ЭК** этого класса $C^A(K)$.

В простейших модификациях этих моделей число голосов, поданных **ЭК** из $C^A(K)$ за принадлежность объекта S к классу K, вычисляется по формуле

$$\Gamma_1(S,K) = \frac{1}{W} \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)} B(S,\sigma,H),$$

здесь $P(\sigma,H)$ — вес **ЭК** (σ,H), $W = \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)}$. В качестве $P(\sigma,H)$ обычно берется число обучающих объектов из K, содержащих (σ,H). По похожей схеме работают полные корректные решающие деревья.

В моделях голосования по (тупиковым) покрытиям класса множество $C^A(K)$ состоит из (тупиковых) покрытий класса K. **ЭК** из $C^A(K)$ голосует за принадлежность распознаваемого объекта классу K, если этот **ЭК** не встречается в описании распознаваемого объекта S. Принадлежность объекта S классу K в простейшей модификации оценивается величиной

$$\Gamma_2(S,K) = \frac{1}{W} \sum_{(\sigma,H)\in C^A(K)} P_{(\sigma,H)} \left(1 - B(S,\sigma,H)\right).$$

Сравнение описанных выше моделей по качеству распознавания приведено в [27]. Наиболее информативными являются **ЭК** небольшого ранга. Поэтому при решении прикладных задач, как правило, либо ограничивают ранг **ЭК**, либо рассматривают только тупиковые корректные **ЭК** (при этом не обязательно все). Например, алгоритм КОРА, предложенный в [5], использует **ЭК** с рангом меньшим или равным трем. Модель голосования по покрытиям класса выигрывает по скорости счета, если число прецедентов из K существенно меньше числа прецедентов из $\overline{K}$.

В задачах с большой значностью признаков, как правило, почти все корректные **ЭК** имеют большой ранг и, как следствие, каждый такой **ЭК** содержится в небольшом числе прецедентов. При этом под значностью признака понимается число его различных значений, и вещественнозначные данные часто трактуются как целочисленные высокой значности. Задачи, в которых значность признаков слишком большая, сложны для классических логических алгоритмов распознавания. Существуют различные способы решения этой проблемы. Один из этих способов основан на идеях алгебраического подхода [34] и базируется на использовании произвольных

ЭК (не обязательно корректных **ЭК**). Алгебраический подход применяется тогда, когда требуется скорректировать работу нескольких алгоритмов, каждый из которых безошибочно классифицирует лишь часть обучающих объектов. Цель коррекции — сделать так, чтобы ошибки одних алгоритмов были скомпенсированы другими алгоритмами. И качество результирующего алгоритма оказалось лучше, чем качество каждого из базовых алгоритмов в отдельности. Об алгебро-логическом подходе говорят, когда каждый базовый алгоритм однозначно определяется некоторым **ЭК** и корректирующие функции являются булевыми [15, 64–66]. Оценки принадлежности распознаваемого объекта к классам вычисляются с использованием процедуры голосования по корректным наборам **ЭК**.

Рассмотрим набор **ЭК** $U = \{(\sigma_1, H_1), \ldots, (\sigma_t, H_t)\}$. Бинарный вектор $U(S) = (q_1(S), \ldots, q_t(S))$, в котором $q_i(S) = B(\sigma_i, S, H_i)$ при $i \in \{1, 2, \ldots, t\}$, называется *откликом* набора U на объекте S. Очевидно, что набор **ЭК** U является корректным для класса K тогда и только тогда, когда не существует двух прецедентов с одинаковыми откликами, один из которых принадлежит K, а другой K не принадлежит. Очевидно также, что для корректного для класса K набора **ЭК** U всегда существует частичная булева функция $F(U, K)$, выполняющая роль корректирующей. Функция $F(U, K)$ определена на откликах прецедентов и принимает значение 1 только на откликах прецедентов из K.

Особо следует отметить корректные наборы **ЭК**, имеющие в качестве корректирующей функции монотонную булеву функцию. Такие наборы называются *монотонными.* Очевидным является следующее

Утверждение 1.2.1 [15]. *Набор **ЭК** U является монотонным корректным набором для класса K тогда и только тогда, когда для любых двух обучающих объектов S' и S'', $S' \in K$, $S'' \notin K$, в наборе U можно указать **ЭК** (σ, H) такой, что $B(\sigma, S', H) = 1$, $B(\sigma, S'', H) = 0$.*

Заметим, что если набор **ЭК** U является монотонным корректным набором для класса K и состоит из одного **ЭК** (σ, H), то (σ, H) — представительный **ЭК** для класса K. Набор признаков H, являющийся тестом, порождает для каждого класса K корректный набор **ЭК**, в котором каждый **ЭК** — представительный **ЭК** для K, имеющий вид (σ, H).

Логический корректор на этапе обучения для каждого класса K строит семейство W_K, состоящее из корректных для K наборов **ЭК**. Далее осуществляется голосование по наборам из семейства W_K. Корректность классификации обеспечивается за счет корректности каждого голосующего набора **ЭК**.

Распознавание объекта S по корректному набору **ЭК** U класса K осуществляется следующим образом. Для каждого объекта S' из обучающей выборки, принадлежащего классу K, выписывается вектор $U(S')$, который сравнивается с вектором $U(S)$ (сравниваются отклики набора U на объектах S и S'). Далее запись $U(S') \preceq U(S)$ означает, что каждая координата вектора $U(S')$ не превосходит соответствующую координату вектора $U(S)$. Объект S не получает голос за принадлежность классу K, если не выполнено $U(S') \preceq U(S)$. Если U — монотонный корректный набор **ЭК** класса K и $U(S') \preceq U(S)$, то объект S получает голос за принадлежность классу K. Если же U не является монотонным корректным набором **ЭК** класса K, то S получает голос за принадлежность к классу K только в случае $U(S') = U(S)$.

На практике проверено, что целесообразно использовать корректные наборы **ЭК** небольшой длины, в частности, тупиковые. Корректный набор **ЭК** для класса K называется тупиковым, если любое его собственное подмножество не является корректным для K набором **ЭК**.

Решение прикладных задач показывает, что монотонный логический корректор (логический корректор, использующий только монотонные корректные наборы **ЭК**) работает лучше немонотонного логического корректора.

В простейших моделях логических корректоров для снижения вычислительной сложности делается следующее. Обучающая выборка делится на базовую и настроечную подвыборки. По базовой подвыборке строятся корректные наборы **ЭК**, по настроечной подвыборке оценивается распознающая способность этих наборов. При этом для каждого класса с помощью генетического алгоритма отбираются наборы **ЭК** ранга 1 с распознающей способностью, близкой к максимальной.

В более сложных моделях используются **ЭК** произвольного ранга и для сокращения временных затрат строятся так называемые локальные базисы классов. Локальному базису соответствует корректный набор **ЭК** достаточно большой мощности, который используется для построения семейства тупиковых корректных наборов **ЭК**, обладающих хорошей распознающей способностью. Локальные базисы либо строятся итеративно с использованием бустинга, либо стохастическим способом.

В [15] предложена общая схема синтеза корректных логических процедур классификации, в рамках которой описаны алгоритмы голосования по корректным **ЭК** и существующие логические корректоры. В этой же работе построен и исследован логический корректор с поляризуемой булевой функцией в качестве корректирующей. Монотонная булева функция является частным случаем поляризуемой булевой функции.

На этапе построения семейств корректных наборов **ЭК** также, как и при построении логических процедур классификации, основанных на голосовании по корректным **ЭК**, приходится решать сложные дискретные задачи, среди которых центральное место принадлежит задаче монотонной дуализации.

Задача формулируется следующим образом: дана конъюнктивная нормальная форма, реализующая монотонную булеву функцию $F(x_1, \ldots, x_n)$. Требуется построить сокращенную дизъюнктивную нормальную форму функции F.

Задача монотонной дуализации может быть также сформулирована на языке гиперграфов. Дан гиперграф $\mathcal{H}$ с n вершинами и m ребрами. Требуется отыскать все минимальные вершинные покрытия гиперграфа $\mathcal{H}$.

Как правило, используется матричная формулировка монотонной дуализации. Пусть $L = \|a_{ij}\|_{m\times n}$ — булева матрица, и пусть H — набор столбцов матрицы L. Набор H называется *покрытием* матрицы L, если каждая строка матрицы L в пересечении хотя бы с одним столбцом из H даёт 1. Покрытие H называется *неприводимым*, если любое собственное подмножество H не является покрытием L. Через $P(L)$ обозначается множество всевозможных неприводимых покрытий матрицы L. Требуется построить множество $P(L)$.

Поиск искомых корректных **ЭК** или корректных наборов **ЭК** сводится к поиску неприводимых покрытий булевой матрицы, которая специальным образом строится по обучающей выборке ([3, 6]). В [10] показано, что при поиске корректных **ЭК** можно обойтись без построения вспомогательной булевой матрицы, если ввести понятие (тупикового) покрытия целочисленной матрицы.

Теоретические оценки эффективности алгоритмов монотонной дуализации базируются на оценке сложности одного шага, т.е. сложности поиска одного нового решения. Наиболее эффективным считается алгоритм, имеющий полиномиальный от размера входа шаг. Такой алгоритм называется алгоритмом с полиномиальной задержкой. Однако полиномиальные алгоритмы удалось построить лишь для некоторых частных случаев монотонной дуализации, например, для случая 2-КНФ. В настоящее время сформировались два основных направления исследований.

Первое направление нацелено на построение так называемых инкрементальных алгоритмов, когда алгоритму разрешено просматривать решения, найденные на предыдущих шагах. При этом оценка сложности шага алгоритма даётся для худшего случая (для самого сложного варианта задачи). В [70] построен инкрементальный алгоритм монотонной дуализации с квазиполиномиальным шагом, определяемым фактически не только размером входа задачи, но и размером её выхода.

Следует отметить, что инкрементальный подход интересен в основном для теории, поскольку в худшем случае число решений дуализации (размер выхода задачи) растёт экспоненциально с ростом размера её входа. Однако инкрементальные алгоритмы хорошо показывают себя на разреженных матрицах, так как в этом случае общее число решений достаточно мало.

Второе направление основано на построении асимптотически оптимальных алгоритмов дуализации. В этом случае алгоритму разрешено делать лишние полиномиальные шаги при условии, что их число почти всегда должно быть достаточно мало по сравнению с числом всех решений задачи. В рамках данного направления удалось построить алгоритмы монотонной дуализации, эффективные в типичном случае (эффективные для почти всех вариантов задачи). Эти алгоритмы имеют теоретическое обоснование и являются лидерами по скорости счёта.

Асимптотически оптимальные алгоритмы дуализации можно разделить на два типа. К алгоритмам первого типа относятся широкоизвестные АО1 [8] и АО2 [12], а также алгоритмы из [84]. Наилучшие результаты на практике показывают асимптотически оптимальные алгоритмы дуализации второго типа, к которым относятся алгоримы PUNC, RUNC и RUNC-M [63].

Для описания схемы работы алгоритмов дуализации первого типа, введём основные понятия. Будем говорить, что столбец с номером j покрывает строку с номером i матрицы L, если $a_{ij} = 1$.

Пусть H — набор столбцов булевой матрицы L. Если строка i в пересечении хотя с одним столбцом $j \in H$ даёт 1, то будем говорить, что набор H покрывает строку i.

Обозначим через $E(L)$ множество $\{(i, j),\ i \in \{1, 2, \ldots, m\}, j \in \{1, 2, \ldots, n\}\ :\ a_{ij} = 1\}$ номеров единичных элементов матрицы L. Два элемента (i, j) и (t, l) из $E(L)$ называются *совместимыми*, если $a_{il} = 0$, $a_{tj} = 0$. Набор Q элементов из $E(L)$ называется *совместимым*, если любые два различных элемента (i, j) и (t, l) из $E(L)$ совместимы. Совместимый набор Q — *максимальным*, если не существует совместимого набора Q' элементов из $E(L)$ такого, что $Q \subset Q'$.

Пусть Q — совместимый набор элементов из $E(L)$. Столбец j называется *запрещённым* для Q, если существует элемент $(t, l) \in Q$ такой, что столбец j покрывает строку t. В противном случае будем говорить, что столбец j совместим с набором Q.

Пусть $B \subseteq E(L)$. Обозначим через $H(B)$ набор столбцов $\{j \in \{1,2,\ldots,n\} :\ \exists(i, j) \in B\}$.

Будем говорить, что набор Q элементов из $E(L)$ *порождает* набор столбцов $H(Q)$. Строка i матрицы L покрыта набором Q элементов из $E(L)$, если набор столбцов $H(Q)$ покрывает строку i. Совместимый набор Q назовем *покрывающим*, если все строки матрицы L покрыты набором Q. Очевидно, что покрывающий набор является максимальным.

Покрывающий набор $Q = \{(i_1, j_1), \ldots, (i_r, j_r)\}$ называется верхним, если для любого покрывающего набора $Q' = \{(t_1, j_1), \ldots, (t_r, j_r)\}$ верны неравенства $t_u \geq i_u$, $u = 1, 2, \ldots, n$.

Задача построения $P(L)$ сводится к перечислению верхних наборов из $E(L)$.

Асимптотически оптимальный алгоритм монотонной дуализации первого типа строит дерево решений, вершины которого, не считая корня, являются совместимыми наборами элементов из $E(L)$. Корень дерева — пустой набор. Висячим вершинам соответсвует либо верхний набор, либо лишний шаг. Алгоритм монотонной дуализации первого типа за один шаг итеративно строит одну ветвь дерева (каждая ветвь начинается либо в корне, либо в одной из внутренних вершин).

Каждому совместимому набору Q соответствует единичная подматрица матрицы L. Таким образом, алгоритмы дуализации первого типа перечисляют с полиномиальной задержкой единичные подматрицы матрицы L (необязательно все).

В описании схемы работы алгоритмов дуализации первого типа допускаются лишние шаги. В [29] показано, что не существует асимптотически оптимального алгоритма первого типа, перечисляющего неприводимые покрытия без повторений с полиномиальной задержкой при $P \neq NP$.

Опишем схему работы алгоритмов дуализации второго типа. Введём используемые далее понятия и обозначения.

Набор столбцов H матрицы L называется *совместимым*, если подматрица L^H матрицы L, образованная набором столбцов H, содержит все строки вида $(1, 0, 0, \ldots, 0)$, $(0, 1, 0, \ldots, 0), \ldots,$ $(0, 0, 0, \ldots, 1)$, то есть содержит единичную подматрицу с точностью до перестановки строк матрицы L.

Работу асимптотически оптимального алгоритма дуализации второго типа можно представить в виде одностороннего обхода ветвей дерева решений, вершины которого, за исключением корня, — совместимые наборы столбцов. Корень дерева — пустой набор столбцов. Висячие вершины либо являются неприводимыми покрытиями, либо соответствуют лишним шагам алгоритма. При этом каждый шаг алгоритма осуществляется за полиномиальное время.

Каждый шаг алгоритма является итеративной процедурой, в результате которой строится одна ветвь дерева, начинающаяся либо в корне, либо в некоторой построенной ранее внутренней вершине. При переходе от вершины к вершине меняется состояние алгоритма.

Таким образом, алгоритмы дуализации второго типа перечисляют с полиномиальной задержкой максимальные совместимые наборы столбцов матрицы L, и каждому максимальному совместимому набору столбцов матрицы L соответствует максимальная единичная подматрица матрицы L.

Пусть $S(L)$ — множество всех единичных подматриц матрицы L. Обозначим $\phi_2(m,n)$ интервал $(\frac{1}{2}\log_2 mn - \frac{1}{2}\log_2 mn \log_2 mn - \log_2\log_2\log_2 n,\ \frac{1}{2}\log_2 mn - \frac{1}{2}\log_2 mn \log_2 mn + \log_2\log_2\log_2 n)$. Тогда справедлива следующая:

Теорема 1.2.1 [8]. *Если $m^\alpha \leq n \leq 2^m, \alpha > 1$, то для почти всех булевых матриц L размера $m \times n$, при $n \to \infty$ справедливо*

$$|P(L)| \sim |S(L)| \sim \sum_{r \in \phi_2} C_n^r C_m^r r! 2^{-r^2},$$

и длины почти всех покрытий из $P(L)$ принадлежат интервалу ϕ_2.

Теорема была доказана Е.В. Дюковой в 1977 году [8]. Поскольку каждое неприводимое покрытие является совместимым набором столбцов, из этой теоремы следует асимптотическая оптимальность алгоритмов дуализации первого и второго типов.

1.3 Логический анализ данных (LAD)

Рассматриваемый подход был впервые предложен в 1986 году П. Хаммером [75]. Данный подход имеет много общего с методами CVP и в отчечественной школе развивался в работах Ю.И. Журавлёва, В.В. Рязанова и О.В. Сенько ([35, 36, 41, 43, 86]).

Направление LAD изначально было разработано для анализа наборов данных, признаки которых принимают только значения из множества $\{0, 1\}$. Однако во многих реальных задачах приходится сталкивать с целочисленной или даже вещественнозначной информацией. Чтобы сделать такие задачи приемлемыми для технологий LAD, данные необходимо преобразовать в двоичный формат.

Рассмотрим случай, когда признак x_j, $j \in \{0,1,\dots,n\}$, является целочисленным (т.е. принимает лишь ограниченное число значений). Не ограничивая общности суждений, можно считать, что множество его значений есть $\{0, 1, \dots, k-1\}$, $k \in \mathbb{N}$. Бинарная функция $\alpha(a, b)$, a, $b \in \{0, 1, \dots, k-1\}$, принимает значение 1, если $a = b$, и 0 в противном случае. Тогда каждому допустимому значению a признака x_j сопоставляется набор, состоящий из нулей и единиц $(\alpha(a, 0), \alpha(a, 1), \dots, \alpha(a, k-1)$. Такой подход называется *one-hot кодированием.*

Пусть x_j, $j \in \{0,1,\dots,n\}$, является вещественным признаком, т.е. множество его значений не ограничено, и на нём задан линейный порядок. Не ограничивая общности суждений, можно считать, что значения признака x_j есть вещественные числа. Рассмотрим $k-1$ вещественных чисел $t_1, t_2, \dots, t_{k-1}$ таких, что $t_1 < t_2 < \dots < t_{k-1}$. Вводится пороговая функция $\alpha(a, t_1, t_2, \dots, t_{k-1})$, $a, t_1, \dots, t_{k-1} \in \mathbb{R}$, принимающая значение 0, если $a < t_1$; 1, если $t_1 \le a < t_2$ и т.д., k если $a \ge t_{k-1}$. Таким образом, каждому значению признака x_j сопоставлено вещественное число из множество $\{t_1, t_2, \dots, t_{k-1}\}$. Данное множество конечно и следовательно может быть кодировано целыми числами $0, 1, \dots, k-1$. Данная процедура называется *дискретизацией* или *понижением значности.* В результате применения процедуры дискретизации, признак x_j становится целочисленным и может быть бинаризован *one-hot* кодированием.

Отдельной темой исследования является вопрос поиска порогов для процедуры дискретизации. Классическим является следующий подход, предложенный впервые в [51]. Он обеспечивает такое понижение значности, при котором объекты из разных классов остаются различимыми. Особенностью данного подхода является использование отдельного набора порогов для каждого класса.

В качестве порогов для признака x_j, $j \in \{0,1,\ldots,n\}$ рассматриваются наблюдаемые на обучающей выборке значения признака $a_{1j}, a_{2j}, \ldots, a_{mj}$, где a_{ij} — значение признака x_j прецедента S_i. Не ограничивая общности суждений, будем считать, что среди них нет одинаковых значений (в противном случае просто имеем меньшее число пороговых значений). Бинарная функция $\alpha_j(S, a_{ij})$, $S = (a_1, a_2, \ldots, a_n) \in M$ принимает значение 1, если $a_j \geq a_{ij}$, и 0 в противном случае. Бинарная переменная y_{ij} принимает значение 1, если значение a_{ij} используется в процедуре дискретизации в качестве порога, и ноль в противном случае. Задача поиска минимального набора порогов, обеспечивающего разделимость прецедентов из класса K и прецедентов из $\overline{K}$ формулируется в виде задачи целочисленного линейного программирования:

$$
\begin{aligned}
&\sum_{j=1}^{n}\sum_{i=1}^{m} y_{ij} \to \min_{y_{ij}},\\
&\sum_{j=1}^{n}\sum_{i=1}^{m} |\alpha_j(S, a_{ij}) - \alpha_j(S', a_{ij})| \cdot y_{ij} \geq h,\ S \in R(K),\ S' \in R(\overline{K}),
\end{aligned}
\tag{1.1}
$$

где h — заранее выбранное натуральное число.

На практике использование приведенной выше модели оптимизации имеет существенные недостатки. Число порогов в найденном решении может быть слишком мало для отдельных признаков, что уменьшает разделяющую способность классификатора. Поэтому часто оказывается, что приближенное решение задачи 1.1 может быть лучше оптимального с точки зрения времени работы и итогового качества классификации.

В [41] была предложена оптимизационная модель для выбора таких порогов, при которых объекты разных классов оказываются наиболее различимы. При этом число порогов для каждого признака ограничено. В качестве меры различимости объектов при выборе порогов используется суммарное число признаков, по которым объекты различаются между собой. Для корректной классификации пороги должны быть подобраны так, чтобы любые два прецедента из разных классов оказались различимы. В терминах целочисленного линейного программирования данная задача может быть записана в виде:

$$\sum_{S\in R(K)}\sum_{S'\in R(\overline{K})}\sum_{j=1}^{n}\left(1-\prod_{j=1}^{m}(1-|\alpha(S,a_{ij})-\alpha(S',a_{ij})|\cdot y_{ij})\right)\to\max_{y_{ij}},$$
$$\sum_{i=1}^{n}\sum_{j=1}^{m}|\alpha(S,a_{ij})-\alpha(S',a_{ij})|\cdot y_{ij}\geq h,\ S\in R(K),\ S'\in R(\overline{K}), \qquad (1.2)$$
$$\sum_{j=1}^{m}y_{ij}\leq h,\ i=1,2,\ldots,n,$$

где h — заранее выбранное натуральное число.

Решением задачи 1.2 фактически является набор порогов для вещественных признаков. При этом число порогов для каждого вещественного признака не превосходит h.

Ключевым в направлении LAD является понятие *логической закономерности* класса K. *Логической закономерностью* (ЛЗ) класса K называют представительный для класса K **ЭК**. В английской литературе вместо логической закономерности используют термин “pattern”. При описании направления LAD также часто используется язык теории булевых функций.

Положим $R_K(\sigma,H)=\{S\in R(K):B(S,\sigma,H)=1\}$, $|R_K(\sigma,H)|$ — мощность множества $R_K(\sigma,H)$.

Пусть **ЭК** (σ,H) — представительный **ЭК** для класса K. **ЭК** (σ,H) называется *максимальной ЛЗ (наибольшей ЛЗ, maximum pattern)*, если $|R_K(\sigma,H)|\geq|R_K(\sigma',H')|$ для любого другого представительного *ЭК* (σ',H') класса K. В дальнейшем будем использовать термин *наибольшая ЛЗ* для удобства.

Пусть $S\in M$ — произвольный объект из множества M. **ЭК** (σ,H) называется *S-максимальной ЛЗ (S-наибольшей ЛЗ, S-maximum pattern)*, если **ЭК** (σ,H) содержит объект S и $|R_K(\sigma,H)|\geq|R_K(\sigma',H')|$ для любого другого представительного ЭК (σ',H') класса K, который содержит объект S.

ЭК (σ,H) называется *сильной ЛЗ (strong pattern)*, если не существует другого представительного **ЭК** (σ',H') класса K такого, что $R_K(\sigma,H)\subset R_K(\sigma',H')$.

Пример. Рассмотрим задачу классификации с шестью прецедентами $S_1 = (0{,}1{,}0{,}0)$, $S_2 = (0{,}1{,}0{,}1)$, $S_3 = (0{,}1{,}1{,}0)$, $S_4 = (0{,}0{,}1{,}1)$, $S_5 = (0{,}0{,}1{,}1)$, $S_6 = (1, 0, 1, 0)$.. При этом S_1,S_2,$S_3 \in K_1$, S_4,S_5,$S_6 \in K_2$.

ЭК $((1{,}0), \{x_1, x_2\})$ является наибольшей ЛЗ класса K_1, поскольку содержится во всех прецедентах класса K_1 и не содержится в прецедентах класса K_2. Однако он не является тупиковым представительным **ЭК** класса K_1, поскольку **ЭК** $((1), \{x_2\})$ — представительный для класса K_1.

ЭК $((0), \{x_3\})$ является тупиковым представительным **ЭК** класса K_1, но не является наибольшей ЛЗ, поскольку содержится только в двух прецедентах из класса K_1.

В LAD схема работы классифицирующего алгоритма полностью аналогична схеме работы алгоритма CVP. На этапе обучения для каждого класса K строится некоторое множество $C^A(K)$ логических закономерностей, например, строятся сильные логические закономерности. На следующем этапе каждый элемент множества $C^A(K)$ «голосует» за отнесение объекта S классу K. Для оценки принадлежности распознаваемого объекта S классу K суммируются величины $B(S, \sigma, H)$, $(\sigma{,}H) \in C^A(K)$ (возможно суммирование с весами).

В LAD большое внимание уделяют построению алгоритмов классификации с высокой интерпретируемостью. Это обеспечивается за счёт относительно небольшой мощности множества $C^A(K)$.

Для решения проблемы поиска всех наибольших ЛЗ в [43] был предложен комбинаторный алгоритм. Введём следующие обозначения для описания комбинаторного алгоритма. Пусть $\rho_j(S, K)$, $j = 1, 2, \ldots, n$, — число прецедентов класса K, у которых признак x_j принимает значения отличные от объекта S. Пусть также $\rho(S, K) = \max_j \rho_j(S, K)$.

Сам алгоритм может быть описан при помощи рекурсивной процедуры $CombinatorialOpt(A, B, (\sigma, H), m)$. Данная процедура принимает на вход множество объектов $A \subseteq M$, множество объектов $B \subseteq M$, **ЭК** (σ, H) и целое число m.

Для поиска наибольших ЛЗ класса K процедура запускается с параметрами $A = R(K)$, $B = R(\overline{K})$, σ — пустой набор, $H = \varnothing$ и $m = 0$. Здесь m означает число прецедентов из класса K, в которых содержится максимальный **ЭК**.

В теле процедуры осуществляется поиск объекта $S \in B$ такого, что $\rho(S, K)$ минимально среди всех $S' \in B$. Далее для каждого **ЭК** $((\sigma_j), \{x_j\})$ ранга 1, $j \in \{1, 2, \dots, n\}$, содержащегося в объектах из A и не содержащегося в объекте S осуществляется проверка, содержится ли данный **ЭК** в объектах из B.

Обозначим множество объектов из A, которые содержат $((\sigma_j), \{x_j\})$ как $S(A)$. Тогда, если $((\sigma_j), \{x_j\})$ не содержится в объектах из B и $|S(A)| > m$, то $m = |S(A)|$, а **ЭК** $(\sigma{\cdot}(\sigma_j), H{\cup}\{x_j\})$ становится промежуточным решением. Если же **ЭК** $((\sigma_j), \{x_j\})$ содержится в объектах из B, то происходит вызов процедуры $CombinatorialOpt(S(A), S(B), ((\sigma)\cdot(\sigma_i), H \cup \{x_i\}, m)$. Если в результате данного вызова было найдено решение лучше текущего, то следует обновить текущее решение и переменную m.

После обхода всех таких **ЭК** процедура возвращает текущее решение и переменную m.

Очевидно, что алгоритм голосования по всем наибольшим ЛЗ не гарантирует корректность классификации. Поэтому на практике комбинаторный алгоритм тривиальным образом модифицируется для поиска S-наибольших ЛЗ класса K для каждого прецедента S из K. При голосовании по множеству $C^A(K)$, содержащем для всех S из K все S-наибольшие ЛЗ для каждого класса K очевидно будет обеспечена корректная классификация.

Отметим, что комбинаторный алгоритм может быть применён для поиска наибольших и S-наибольших ЛЗ и в случае целочисленных данных без предварительной бинаризации.

Пусть $S = (a_1, a_2, \dots, a_n)$, $a_j \in \{0, 1\}$ при $j = 1{,}2, \dots, n$. Тогда задача поиска S-наибольшей ЛЗ может быть представлена в виде задачи нелинейного целочисленного программирования:

$$
\begin{aligned}
&\sum_{S'=(b_1,\dots,b_n)\in R(K)} \prod_{a_j \neq b_j} (1 - y_j) \to \max_{y_j},\\
&\sum_{a_j \neq c_j} y_j \geq 1, S'' = (c_1, \dots, c_n) \in R(\overline{K}),\\
&y_j \in \{0, 1\}.
\end{aligned}
\tag{1.3}
$$

Решением задачи 1.3 будет набор бинарных переменных y_j, $j = 1{,}2,\ldots,n$. Рассмотрим **ЭК** (σ, H), $\sigma = (a_{j_1}, a_{j_2}, \ldots, a_{j_r})$, $H = \{x_{j_1}, x_{j_2}, \ldots, x_{j_r}\}$, такой, что $x_t \in H$ при $y_t = 1$ и $x_t \notin H$ при $y_t = 0$, $t = 1{,}2,\ldots,n$.

Как правило, задачи нелинейного целочисленного программирования тяжело решать даже приближёнными методами, поэтому предлагается решать задачу целочисленного программирования, эквивалентную исходной. Рассмотрим бинарную переменную $z_{S'} = \prod_{a_i \neq b_i}(1 - y_i)$, $S' = (b_1, b_2, \ldots, b_n) \in R(K)$. Обозначим $w(S')$ — число признаков, значения которых различаются у объектов S и S'. Имеем следующую задачу линейного программирования:

$$
\begin{aligned}
&\sum_{S' \in R(K) \setminus \{S\}} z_{S'} \to \max_{z,y}, \\
&\sum_{a_j \neq c_j} y_j \geq 1,\ S'' = (c_1, c_2, \ldots, c_n) \in R(\overline{K}), \\
&w(S') \cdot z_{S'} + \sum_{a_j \neq b_j} y_j \leq w(S'),\ S' = (b_1, b_2, \ldots, b_n) \in R(K) \setminus \{S\}, \\
&z_{S'} + \sum_{a_j \neq b_j} y_j \geq 1,\ S' = (b_1, b_2, \ldots, b_n) \in R(K) \setminus \{S\}.
\end{aligned}
\tag{1.4}
$$

Число бинарных переменных, появляющихся в задаче 1.4, равно $n + |R(K)| - 1$, что в случае больших наборов данных приводит к очень большим целочисленным линейным программам. Ввиду вычислительной сложности обработки таких больших целочисленных линейных программ часто используются приближённые методы решения задач целочисленного линейного программирования.

В [87] был предложен MILP (Mixed 0-1 Integer and Linear Programming) подход, позволяющий найти S-наибольшие ЛЗ определённого ранга. Рассмотрим набор из $2n$ бинарных переменных $z_1, z_2, \ldots, z_{2n}$. При $j = 1, 2, \ldots, n$ переменная z_j принимает значение 1, если $x_j \in H$ и $\sigma_j = 1$, и 0 в противном случае. При $j = n, n+1, \ldots, 2n$ переменная z_j принимает значение 1, если $x_j \in H$ и $\sigma_j = 0$, и 0 в противном случае. Введём также переменную $y_{S'}$, $S' = (b_1, b_2, \ldots, b_n) \in R(K)$ равную единице, если

$\sum_{j=1}^{n}(b_j z_j + \bar{b}_j z_{j+n}) < d$, и нулю в противном случае. Рассмотрим задачу линейного целочисленного программирования:

$$\begin{aligned}
&\sum_{S' \in R(K)\setminus\{S\}} y_{S'} \to \min_z, \\
&\sum_{j=1}^{n}(b_j z_j + \bar{b}_j z_{j+n}) + n y_{S'} \geq d, S' = (b_1, b_2, \ldots, b_n) \in R(K) \setminus \{S\}, \\
&\sum_{j=1}^{n}(b_j z_j + \bar{b}_j z_{j+n}) \leq d - 1, S' = (b_1, b_2, \ldots, b_n) \in R(\overline{K}), \\
&\sum_{j=1}^{2n} z_j = d,
\end{aligned} \tag{1.5}$$

где d — натуральное число, не превосходящее n. В [87] была доказана теорема

Теорема 1.3.1 [87]**.** *Если z^* является решением задачи 1.5, то соответствующий* ***ЭК*** (σ, H) *ранга d является S-наибольшей ЛЗ класса K.*

Описанный подход может быть использован для поиска сильных ЛЗ класса. Для этого рассмотрим модификацию задачи 1.5:

$$\begin{aligned}
&\sum_{S' \in R(K)\setminus\{S\}} y_{S'} \to \min_z, \\
&\sum_{j=1}^{n}(b_j z_j + \bar{b}_j z_{j+n}) + n y_{S'} \geq d,\ S' = (b_1, b_2, \ldots, b_n) \in R(K) \setminus \{S\}, \\
&\sum_{j=1}^{n}(b_j z_j + \bar{b}_j z_{j+n}) \leq d - 1,\ S' = (b_1, b_2, \ldots, b_n) \in R(\overline{K}), \\
&\sum_{j=1}^{2n} z_j = d, \\
&z_j + z_{n+j},\ j = 1{,}2, \ldots{,}2n.
\end{aligned} \tag{1.6}$$

В [87] была доказана теорема

Теорема 1.3.2 [87]. *Если* (z^*, y^*) *является решением задачи* 1.6, *то соответствующий* ***ЭК*** (σ, H) *ранга* d *является сильной ЛЗ класса* K *степени* d.

1.4 Анализ формальных понятий (FCA)

В начале 80-х годов В.К. Финн изобрёл так называемый ДСМ-метод автоматического порождения гипотез [44], в котором формализованы различные когнитивные процедуры, основанные на понятии сходства. Позднее была отмечена схожесть ([40, 82]) между ДСМ-методом и методами направления FCA, предложенным Р. Вилле в [93]. Окончательно связь между направлениями установлена в [71]. При этом в ДСМ-методе есть понятия, аналогов которым в направлении FCA не существует. Подробный обзор ДСМ-метода и направления FCA есть в [83].

Опишем основные понятия из FCA. Отметим, что обычно методы FCA, как и методы LAD, работают с бинарной информацией. Однако *one-hot* преобразование применяется в FCA и в этом случае.

Пусть $\tilde{X}$ — множество бинарных признаков мощности n', полученное в результате *one-hot* преобразования множества X. Говорят, что объект $S = (a_1, \ldots, a_n')$ обладает признаком $\tilde{x}_j \in \tilde{X}$, $j \in \{1,2, \ldots, n'\}$, если $a_j = 1$.

Формальным контекстом называется тройка $(R, \tilde{X}, I)$, где $R \subseteq M$, I — бинарное отношение между множествами R и $\tilde{X}$. Иначе говоря, формальный контекст представляет собой булеву матрицу L, строками которой являются описания объектов из R посредством признаков из $\tilde{X}$. Далее запись $(S,x) \in I$ означает, что объект $S \in R$ обладает признаком $x \in \tilde{X}$.

Операторами вывода для формального контекста $C = (R, \tilde{X}, I)$ и множеств $A \subseteq R$, $B \subseteq \tilde{X}$ называются соответственно множества $A^* = \{\tilde{x} \in \tilde{X} \mid (S,x) \in I \ \forall S \in A\}$ и $B^* = \{S \in R \mid (S,x) \in I \ \forall \tilde{x} \in \tilde{X}\}$.

Таким образом, A^* — это оператор, возвращающий столбцы матрицы L, которые в пересечении с заданными строками образуют «максимальную» подматрицу, имеющую $|A|$ строк и не имеющую элементов, равных 0. Аналогично, B^* — это оператор, возвращающий строки матрицы L, которые в пересечении с заданными столбцами образуют «максимальную» подматрицу, имеющую $|B|$ столбцов и не имеющую элементов, равных 0.

Формальным понятием формального контекста $(R, \tilde{X}, I)$ называется пара (A, B), $A \subseteq R$, $B \subseteq \tilde{X}$, где $A^* = B$ и $B^* = A$. Заметим, что паре (A,B) соответствует «максимальная» подматрица матрицы L, состоящая из единичных элементов.

Заметим, что общее число формальных понятий может расти экспоненциально с ростом размера задачи. В качестве примера рассмотрим формальный контекст $K = (R, \tilde{X}, I)$, где $\tilde{X} = \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n\}$, а множество R состоит из n объектов $S_1, S_2, \ldots, S_n$, и для $S_i = (a_{i1}, a_{i2}, \ldots, a_{in})$, $i = 1, 2, \ldots, n$, $a_{ij} = 0$ при $i = j$, и $a_{ij} = 1$ при $i \neq j$, $j = 1, 2, \ldots, n$. Не трудно видеть, что пара $(R \setminus \{S_{i1}, S_{i2}, \ldots, S_{ik}\}, \{\tilde{x}_{i1}, \tilde{x}_{i2}, \ldots, \tilde{x}_{ik}\})$ будет являться формальным понятием формального контекста K для любых $k = 1, 2, \ldots, n$ и любых $i_1, i_2, \ldots, i_k$, $0 < i_1 < i_2 < \cdots < i_k$. Нетрудно видеть, что формальных понятий такого вида 2^n.

Обозначим $\mathfrak{B}(K)$ — множество всех формальных понятий формального контекста K. Рассмотрим бинарное отношение $\preceq$ на множестве $\mathfrak{B}(K)$ такое, что для любых $(A_1, B_1) \in \mathfrak{B}(K)$ и любых $(A_2, B_2) \in \mathfrak{B}(K)$ $(A_1, B_1) \preceq (A_2, B_2)$ тогда и только тогда, когда $A_1 \subseteq A_2$ и $B_2 \subseteq B_1$. В [93] было доказано, что множество $\mathfrak{B}(K)$ является полной решёткой.

Задача поиска всех формальных понятий для произвольного формального контекста $K = (R, \tilde{X}, I)$ является перечислительной. В [81] были предложен алгоритм *Close-by-One* построения решётки формальных понятий с полиномиальной задержкой.

Шагу алгоритма *Close-by-One* соответствует процедура: *CbODown*. Процедура *CbODown* принимает на вход формальное понятие (A, B) и объект $S \in R$. Тогда *CbODown* возвращает пару $((A \cup \{S\})'', B \cap \{S\}')$. Аналогичная *CbODown* процедура *CbOUp*, которая принимает на вход формальное понятие (A, B) и признак $\tilde{x} \in \tilde{X}$ и возвращает пару $(A \cap \{x\}', (B \cap \{x\})'')$. Впервые процедуры под этими именами предложены в [91]. Справедливо утверждение

Утверждение 1.4.1 [91]**.** *Для произвольных формального понятия (A, B) формального контекста $(R, \tilde{X}, I)$, объекта $S \in R$ и признака $x \in H$ формальными понятиями будут пары $CbODown(A, B, S)$ и $CbOUp(A, B, \tilde{x})$.*

Алгоритм *Close-by-One* имеет сложность $O(|R| \cdot |\tilde{X}| \cdot |L|)$, где L — множество всех формальных понятий формального контекста $(R, \tilde{X}, I)$.

В [78] построены эффективные инкрементальные алгоритмы. В [79] разработана параллельная версии алгоритма *Close-by-One*. В [80] дано подробное теоретическое и экспериментальное сравнение различных алгоритмов построения решётки формальных понятий.

Рассмотрим формальные контексты $C_K = (R(K), \tilde{X}, I)$ и $C_{\overline{K}} = (R(\overline{K}), \tilde{X}, I)$. Введём обозначения: H^+ — оператор вывода для формального контекста C_K и множества признаков H; H^- — оператор вывода для формального контекста $C_{\overline{K}}$ множества признаков H.

Набор признаков $H \subseteq \tilde{X}$ называется *положительной ДСМ-гипотезой* для класса K, если пара (H^+,H) является формальным понятием формального контекста C_K и $H^- = \varnothing$.

Положительной ДСМ-гипотезе $H = \{\tilde{x}_{j_1}, \ldots, \tilde{x}_{j_r}\}$ класса K поставим в соответствие **ЭК** (σ, H'), $\sigma = (\sigma_1, \ldots, \sigma_r)$, $H' = \{x_{t_1}, \ldots, x_{t_r}\}$, $H' \subseteq X$, такой что при любом $i \in \{1,2, \ldots, r\}$ признак x_{t_i} принимает значение σ_i тогда и только тогда, когда бинарный признак $\tilde{x_{j_i}}$ принимает значение 1. Будем говорить, что H *порождает* **ЭК** (σ, H'). Нетрудно видеть, что (σ, H') — представительный **ЭК** класса K.

Пример. Рассмотрим задачу классификации с тремя прецедентами $S_1 = (1,1,1)$, $S_2 = (0,1,1)$ и $S_3 = (1,0,0)$. При этом S_1, $S_2 \in K_1$, $S_3 \in K_2$. После *one-hot* преобразования прецеденты будут иметь соответственно описания (0,1,0,1,0,1), (1,0,0,1,0,1), (0,1,1,0,1,0). В данном примере **ЭК** $((1,1),\{x_2,x_3\})$ порождён положительной для класса K_1 ДСМ-гипотезой $H = (\tilde{x}_4, \tilde{x}_6)$. Этот **ЭК** не является тупиковым представительным **ЭК** класса K_1, поскольку $((1),\{x_2\})$ — представительный **ЭК** для K_1. Отметим, что тупиковый представительный **ЭК** $((1), \{x_2\})$ класса K_1 не порождается ни одной положительной ДСМ-гипотезой класса K_1.

Обучение ДСМ-метода заключается в построении для каждого класса K множества $C^A(K)$ представительных **ЭК**, порождённых положительными для данного класса ДСМ-гипотезами. В упрощённом варианте процедура принятия решения в отношении классифицируемого объекта $S \in M$ выглядит следующим образом:

1) Если объект S содержит хотя бы **ЭК** из $C^A(K)$, и не содержит ни одного **ЭК** из $C^A(K')$ для каждого $K' \neq K$, то объект S принадлежит классу K;
2) Если условие 1) не выполняется ни для одного класса, то алгоритм отказывается от классификации объекта S.

Пример. Рассмотрим задачу классификации с тремя прецедентами $S_1 = (1{,}1{,}1)$, $S_2 = (0{,}1{,}1)$ и $S_3 = (1{,}0{,}0)$. При этом S_1, $S_2 \in K_1$, $S_3 \in K_2$. После *one-hot* преобразования прецеденты будут иметь соответственно описания (0,1,0,1,0,1), (1,0,0,1,0,1), (0,1,1,0,1,0). В данном примере **ЭК** $((1{,}1),\{x_2{,}x_3\})$ порождён положительной для класса K_1 ДСМ-гипотезой $H = (\tilde{x}_4{,}\tilde{x}_6)$. Этот **ЭК** не является тупиковым представительным **ЭК** класса K_1, поскольку $((1), \{x_2\})$ — представительный **ЭК** для K_1. Отметим, что тупиковый представительный **ЭК** $((1),\{x_2\})$ класса K_1 не порождается ни одной положительной ДСМ-гипотезой класса K_1.

Задача построения гипотез для ДСМ-метода вычислительно сложна. Трудность этой проблемы была изучена в [39] и продемонстрирована следующими теоремами для положительных ДСМ-гипотез.

Теорема 1.4.1 [39]**.** *Задача проверки существования положительной гипотезы, состоящей из $h \in \mathbb{N}$ признаков, является NP-полной.*

Теорема 1.4.2 [39]**.** *Задача поиска числа всех положительных гипотез является $\#P$-полной.*

Поэтому на практике часто используются методы, строящие не всё множество положительных для класса K ДСМ-гипотез, а лишь небольшую его часть. Например, ВКФ-метод ([92]) использует Марковские процессы для стохастической генерации ДСМ-гипотез.

1.5 Общая схема синтеза логических процедур классификации

Пусть $\mathcal{P}(K)$ — множество всех представительных для класса K **ЭК**.

Бинарное отношение $\preceq$, заданное но множестве $\mathcal{P}(K)$, называется отношением *частичного предпорядка*, а множество $\mathcal{P}(K)$, соответственно, *частично предупорядоченным*, если это отношение:

1) *рефлексивно* ($(\sigma,H) \preceq (\sigma,H)$ для любого $(\sigma,H) \in \mathcal{P}(K)$);
2) *транзитивно* ($(\sigma,H) \preceq (\sigma',H')$ и $(\sigma',H') \preceq (\sigma'',H'')$ влечёт $(\sigma,H) \preceq (\sigma'',H'')$).

Частичный предпорядок $\preceq$ называется *частичным порядком*, если он антисимметричен ($(\sigma,H) \preceq (\sigma',H')$ и $(\sigma',H') \preceq (\sigma,H)$ влечёт $(\sigma,H) = (\sigma',H')$).

Запись $(\sigma, H) \prec (\sigma', H')$ означает, что $(\sigma,H) \preceq (\sigma',H')$ и $(\sigma',H') \npreceq (\sigma,H)$. **ЭК** $(\sigma,H) \in \mathcal{P}(K)$ называется *максимальным* элементом в $\mathcal{P}(K)$ относительно частичного предпорядка $\preceq$, если не существует **ЭК** $(\sigma',H') \in \mathcal{P}(K)$ такого, что $(\sigma,H) \prec (\sigma',H')$.

Зададим на множестве $\mathcal{P}(K)$ отношение частичного порядка $\preceq_1$ следующим образом. Будем считать, что $(\sigma_1, H_1) \in \mathcal{P}(K)$ следует за $(\sigma_2, H_2) \in \mathcal{P}(K)$ (т.е. $(\sigma_2, H_2) \preceq_1 (\sigma_1, H_1)$), если $H_1 \subseteq H2$ и значения из наборов σ_1, σ_2, соответствующие одинаковым признакам, у данных **ЭК** совпадают. Непосредственно из определения тупикового представительного **ЭК** следует

Утверждение 1.5.1. ***ЭК*** *$(\sigma, H) \in \mathcal{P}(K)$ является тупиковым представительным для класса K тогда и только тогда, когда (σ, H) — максимальный относительно частичного порядка $\preceq_1$ элемент множества $\mathcal{P}(K)$.*

Зададим на множестве $\mathcal{P}(K)$ отношение частичного предпорядка $\preceq_2$ следующим образом. Будем считать, что $(\sigma_1, H_1) \in \mathcal{P}(K)$ следует за $(\sigma_2, H_2) \in \mathcal{P}(K)$, если $|R_K(\sigma_1, H_1)| \geq |R_K(\sigma_2, H_2)|$. Непосредственно из определения наибольшей логической закономерности следует

Утверждение 1.5.2. ***ЭК*** *$(\sigma, H) \in \mathcal{P}(K)$ является наибольшей логической закономерностью для класса K тогда и только тогда, когда (σ, H) — максимальный относительно частичного предпорядка $\preceq_2$ элемент множества $\mathcal{P}(K)$.*

Зададим на множестве $\mathcal{P}(K)$ отношение частичного предпорядка $\preceq_3$ следующим образом. Будем считать, что $(\sigma_1, H_1) \in \mathcal{P}(K)$ следует за $(\sigma_2, H_2) \in \mathcal{P}(K)$, если $R_K(\sigma_2, H_2) \subseteq R_K(\sigma_1, H_1)$. Непосредственно из определения сильной логической закономерности следует

Утверждение 1.5.3. ***ЭК*** *$(\sigma, H) \in \mathcal{P}(K)$ является сильной логической закономерностью для класса K тогда и только тогда, когда (σ, H) — максимальный относительно частичного предпорядка $\preceq_3$ элемент множества $\mathcal{P}(K)$.*

Зададим на множестве $\mathcal{P}(K)$ отношение частичного порядка $\preceq_4$ следующим образом. Будем считать, что $(\sigma_1, H_1) \in \mathcal{P}(K)$ следует за $(\sigma_2, H_2) \in \mathcal{P}(K)$, если $R_K(\sigma_2, H_2) \subseteq R_K(\sigma_1, H_1)$ и $H_2 \subseteq H_1$. Справедливо

Утверждение 1.5.4. ***ЭК*** *$(\sigma, H) \in \mathcal{P}(K)$ порождается положительной для класса K ДСМ-гипотезой тогда и только тогда, когда (σ, H) — максимальный относительно частичного порядка $\preceq_4$ элемент множества $\mathcal{P}(K)$.*

Доказательство. *Достаточность.* Пусть ЭК $(\sigma,H) \in \mathcal{P}(K)$ является максимальным элементом в $\mathcal{P}(K)$ относительно порядка $\preceq_4$. Рассмотрим формальный контекст $C_K = (R(K),\tilde{X},I)$, а также множество признаков $\tilde{H}$ из $\tilde{X}$, порождающее ЭК (σ,H). По определению $\tilde{H}^+ = R_K(\sigma,H)$. Покажем, что $R_K(\sigma,H)^+ = \tilde{H}$.

Предположим противное. Пусть $R_K(\sigma,H)^+ = H'$, $\tilde{H} \subset H'$. Это означает, что существует признак $x_i \in X$, $x_i \notin H$, и его допустимое значение σ', такие что признак x_i принимает значение σ' на всех объектах из $R_K(\sigma,H)$. Заметим, что **ЭК** $((\sigma_1, \ldots, \sigma_r, \sigma'), H \cup x_i)$ — корректный для класса K, так как (σ,H) не содержится в прецедентах не из класса $\overline{K}$. Рассматриваемый **ЭК** содержится в прецедентах из класса K, поскольку он содержится в объектах из множества $R_K(\sigma,H)$.

Следовательно, $((\sigma_1, \ldots, \sigma_r, \sigma'), H \cup x_i)$ — представительный **ЭК** для класса K.

Нетрудно видеть, что $R_K((\sigma_1, \ldots, \sigma_r, \sigma'), H \cup x_i) = R_K(\sigma, H)$. Это значит, что $(\sigma, H) \prec_4 ((\sigma_1, \ldots, \sigma_r, \sigma'), H \cup x_i)$. Противоречие .

Необходимость. Очевидно ■

Таким образом, на основании утверждений 1–4 общая схема работы логического классификатора A выглядит следующим образом:

1) на множестве $\mathcal{P}(K)$ устанавливается частичный предпорядок $\preceq$;
2) для каждого класса K строится множество $C^A(K)$ максимальных относительно частичного предпорядка $\preceq$ элементов;
3) на основе вычисления близости объекта S и элементов множеств $C^A(K)$ принимается решение о принадлежности объекта S каком-либо классу, или происходит отказ от распознавания.

Глава 2. Процедуры CVP над произведением частичных порядков

В данной главе предложены модели логических классификаторов при условии, что признаковые описания объектов — это элементы декартова произведения конечных частично упорядоченных множеств. Данные модели представляют собой обобщение классических моделей голосования по представительным **ЭК** классов и покрытиям классов. Показано, что задача синтеза обобщённых корректных логических алгоритмов классификации по прецедентам сводится к задаче дуализации над произведением частичных порядков. Предложено несколько методов повышения качества классификации, основанных на идее стохастических композиций.

Основные результаты, представленные в этой главе, опубликованы в [2, 22, 23, 57, 58]

2.1 Задача дуализации над произведением частичных порядков

Рассмотрим произвольное частично упорядоченное множество P. Если $x, y \in P$ и $x \preceq y$ ($y \succeq x$), то будем говорить, что y следует за x, а x предшествует y. Элементы x,y из частично упорядоченного множества P называются сравнимыми, если x следует за y или y следует за x. В противном случае x и y несравнимы. Если все элементы множества P попарно сравнимы, то множество P называется *линейно* упорядоченным или *цепью.* Если все различные элементы множества $\mathcal{P}(K)$ попарно несравнимы, то множество $\mathcal{P}(K)$ называется *антилинейно* упорядоченным или *антицепью.*

Элемент x частично упорядоченного множества P называется *наибольшим*, если он следует за каждым элементом из P. Элемент x частично упорядоченного множества P называется *наименьшим*, если он предшествует каждому элементу из P.

Пусть $R \subseteq P$. Элемент $x \in P$ называется *верхней гранью* подмножества R, если он следует за любым элементом из R. Элемент $x \in P$ называется *точной верхней гранью* подмножества R, если он является верхней гранью подмножества R и предшествует каждой верхней грани подмножества R. Аналогичным образом определяется *точная нижняя грань* подмножества R. Частично упорядоченное множество P называется *решёткой*, если любое его двухэлементное подмножество имеет точную верхнюю и точную нижнюю грани.

Пусть $P = P_1 \times \cdots \times P_n$, где $P_1, \ldots, P_n$ — конечные частично упорядоченные множества. Считается, что элемент $x = (x_1, \ldots, x_n) \in P$ следует за элементом $y = (y_1, \ldots, y_n) \in P$ (x предшествует y), если x_i следует за y_i (x_i предшествует y_i) при $i = 1,2,\ldots,n$. Запись $x \prec y$ означает, что $x \preceq y$ и $x \neq y$.

Пусть $R \subseteq P$. Введём обозначения: $R^+ = R \cup \{x \in P |\ \exists y \in R,\ y \prec x\}$ — множество элементов, следующих за элементами из R; $R^- = R \cup \{x \in P |\ \exists y \in R, y \prec x\}$ — множество элементов, предшествующих элементам из R. Элемент x множества $P \setminus R^+$ ($P \setminus R^-$) называется максимальным (минимальным) независимым от R элементом множества P, если для любого другого элемента y множества $P \setminus R^+$ ($P \setminus R^-$) отношение $x \prec y$ ($y \prec x$) не выполняется.

Обозначим через $I(R^+)$ — множество, состоящее из максимальных независимых от R элементов множества P, через $I(R^-)$ — множество, состоящее из минимальных независимых от R элементов множества P. Ставятся задачи построения для заданного R множеств $I(R^+)$ и $I(R^-)$. Множества $I(R^+)$ и $I(R^-)$ являются двойственными (дуальными): если $Q = I(R^-)$ или $Q = I(R^+)$, то соответственно $I(Q^+) = I(R^+)$ и $I(Q^-) = I(R^-)$. Множества $I(R^+)$ и $I(R^-)$ являются антицепями. Далее каждая из поставленных задач называется дуализацией над произведением частичных порядков.

Простейшим случаем дуализации над произведением цепей является дуализация монотонной булевой функции F, заданной своей КНФ (монотонная дуализация). Нетрудно показать, что монотонная дуализация — это задача построения множества $I(R^-)$ при условии, что P — n-мерный булев куб, $P_i = \{0,1\}$ при $i = 1,2,\ldots,n$, в каждом P_i задан порядок $0 \prec 1$ и

R — множество нулей функции F, содержащее множество верхних нулей функции F. Элементы из $I(R^-)$ — это нижние единицы функции F.

В [50] для случая, когда каждое P_i, $i = 1,2,\ldots,n$, является цепью и $|P_i| \geq 2$, на базе алгоритма, предложенного в [70], построен квазиполиномиальный инкрементальный алгоритм. Аналогичный результат получен и для некоторых других специальных конечных частичных порядков (полурешётки, ограниченной ширины, решётки временных интервалов, леса ([69]). Для порядков более общего вида (решёток) в [46] доказана невозможность построения полиномиального инкрементального алгоритма (иначе $P = NP$), и для случая дистрибутивных решёток построен инкрементальный алгоритм с субэкспоненциальной сложностью шага.

В данной работе для дуализации над произведением цепей построен асимптотически оптимальный алгоритм RUNC-M+. Этот алгоритм строит $I(R^+)$ для случая, когда каждое множество $P_i = \{0,1,\ldots,k-1\}$, $i = 1,2,\ldots,n$, — цепь, и элементы в P_i упорядочены по возрастанию. Обоснование асимптотической оптимальности алгоритма RUNC-M+ приведено в четвёртой главе и заключается в установлении асимптотического равенства двух величин, а именно, типичного числа шагов этого алгоритма и типичного числа решений задачи (мощности $I(R^+)$). С этой целью дана матричная постановка задачи построения $I(R^+)$ для случая произведения цепей и усовершенствована техника получения необходимых асимптотических оценок, ранее применявшаяся для обоснования асимптотической оптимальности алгоритмов монотонной дуализации. В четвёртой главе приведена матричная формулировка задачи построения максимальных независимых элементов для произведения конечных частичных порядков P, опирающаяся на более общее, чем рассмотренное в [16] для случая цепей, понятие упорядоченного тупикового покрытия матрицы, строками которой являются наборы из $R \subseteq P$.

2.2 Процедуры CVP, основанные на построении корректных элементарных классификаторов общего вида

В данном разделе предложена более общая постановка логической классификации, нацеленная на решение задач, в которых каждый признак принимает значения из некоторого конечного частично упорядоченного множества чисел.

Пусть $M = N_1 \times \dots \times N_n$, где N_j, $j \in \{1{,}2, \dots, n\}$, — конечное множество значений признака x_j, на котором задан частичный порядок. Будем считать, что каждое множество N_j, $j \in \{1{,}2, \dots, n\}$, имеет наибольший элемент k_j. Элемент $k_j \in N_j$ называется наибольшим, если он следует за любым элементом из N_j. Если наибольший элемент в N_j отсутствует, то N_j дополним таким элементом.

Близость объекта $S = (a_1, \dots, a_n)$ из M и **ЭК** (σ, H), $H = \{j_1, \dots, j_r\}$, $\sigma = (\sigma_1, \dots, \sigma_r)$, $\sigma_i \in N_{j_i}$, при $i = 1{,}2, \dots ,r$, будем оценивать величиной $\tilde{B}(\sigma, S, H)$, равной 1, если $a_{j_i} \preceq \sigma_i$ при $i = 1, 2, \dots, r$, и равной 0 в противном случае. Будем говорить, что объект S *содержит* **ЭК** (σ, H), если $\tilde{B}(\sigma, S, H) = 1$.

Данные в первой главе понятия корректного **ЭК** класса K, представительного **ЭК** класса K, покрытия класса K и теста полностью переносятся на рассматриваемый общий случай, если $B(\sigma, S, H) = 1$ заменить на $\tilde{B}(\sigma, S, H) = 1$.

ЭК (σ, H) называется *корректным* для класса K, $K \in \{K_1, \dots, K_l\}$, если нельзя указать пару обучающих объектов S' и S'' таких, что $S' \in K$, $S'' \notin K$ и $\tilde{B}(\sigma, S', H) = \tilde{B}(\sigma, S'', H) = 1$. Корректный **ЭК** класса K называется *представительным* для класса K, если он содержится хотя бы в одном прецеденте класса K. Корректный **ЭК** класса K называется *покрытием* класса K, если он не содержится ни в одном прецеденте класса K.

Пусть (σ, H) — **ЭК**, в котором $H = \{j_1, \dots, j_r\}$, $\sigma = (\sigma_1, \dots, \sigma_r)$, $\sigma_i \in N_{j_i}$, при $i = 1{,}2, \dots ,r$. **ЭК** (σ, H) сопоставим набор $S(\sigma, H) = (\gamma_1, \dots, \gamma_t)$ из $M = N_1 \times \dots \times N_n$, в котором $\gamma_t = \sigma_i$ при $t = j_i$, $i = 1, 2, \dots, r$, и $\gamma_t = k_t$ при $t \notin \{j_1, \dots, j_r\}$.

Покрытие (σ, H) класса K назовём *тупиковым*, если любой **ЭК** (σ', H') такой, что $S(\sigma, H) \prec S(\sigma', H')$, не является покрытием класса K. Представительный для класса K **ЭК** (σ, H) назовём *тупиковым*, если любой **ЭК** (σ', H') такой, что $S(\sigma, H) \prec S(\sigma', H')$, не является представительным для класса.

Через $R(K)$ обозначим множество прецедентов из класса K.

Утверждение 2.2.1. *Покрытие (σ, H) класса K является тупиковым покрытием класса K тогда и только тогда, когда $S(\sigma, H) \in I(R(K))$.*

Пусть $\overline{K} = M \setminus K$. Будем рассматривать $\overline{K}$ как отдельный класс, т.е. будем считать, что есть всего два класса K и $\overline{K}$. Очевидным является

Утверждение 2.2.2. ***ЭК** (σ, H) является тупиковым представительным для класса K тогда и только тогда, когда $S(\sigma, H) \in I(R(\overline{K}))$ и $S(\sigma, H) \in R(K)^+$.*

Из утверждений 2.2.1 и 2.2.2 следует, что в случае частично упорядоченных данных при построении логических классификаторов, основанных на голосовании по тупиковым покрытиям класса и тупиковым представительным **ЭК** класса, возникает задача построения максимальных независимых элементов декартова произведения частичных порядков.

Рассмотрим частный случай, когда каждое N_j является цепью, $j = 1,2,\ldots,n$. Тогда N_j представимо в виде $\{0, 1, \ldots, k_j - 1\}$, где k_j — общее число различных значений признака x_j. Пусть $S = (a_1, a_2, \ldots, a_n) \in R(K)$, а $S' = (b_1, b_2, \ldots, b_n) \in R(\overline{K})$. Обозначим $l(S, S')$ — целочисленный вектор, j-ый элемент которого равен $\max(0, b_j - a_j)$, $j = 1,2,\ldots,n$; L_S — матрица, строками которой являются вектора $l(S, S')$ для всех прецедентов $S' \in R(\overline{K})$.

Пусть $C = (c_1, c_2, \ldots, c_n) \in I(L_S)$. Обозначим $C(S)$ — целочисленный вектор, j-ая компонента которого равна $\min(k_j - 1, c_j + a_j)$. Рассмотрим также **ЭК** (σ, H), порождённый объектом $C(S)$. Очевидно следующее

Утверждение 2.2.3. ***ЭК** (σ, H) содержится в объекте S и является тупиковым представительным для класса K.*

В некоторых случаях, нахождение тупиковых представительных **ЭК** при помощи утверждения 2.2.3 может быть вычислительно более эффективным, чем прямое решение задачи дуализации для множества $R(\overline{K})$.

Замечание. Если каждое множество N_i, $i \in \{1,2,\ldots,n\}$, — антицепь с добавленным наибольшим элементом, то $R(K)^+ = R(K), B(\sigma, S, H) = \tilde{B}(\sigma, S, H)$ и, следовательно, в этом случае мы имеем классическую постановку, рассмотренную в первой главе.

В общем случае существование представительных **ЭК** класса K не гарантировано. Также как и в классическом случае для корректности алгоритма голосования по (тупиковым) представительным **ЭК** необходимо, чтобы описания объектов из разных классов были несравнимы.

Покажем, что в случае непересекающихся классов существует такое преобразование признакового описания множества M, в результате которого для каждого класса K всегда образуется непустое множество тупиковых представительных **ЭК** и любой прецедент из K порождает хотя бы один **ЭК** из данного множества.

Обозначим через $\tilde{\tilde{P}}$ множество, совпадающее с множеством P, но с обратным отношением порядка, т.е. $x \preceq y$ в $\tilde{\tilde{P}}$ тогда и только тогда, когда $y \preceq x$ в P.

Пусть $\tilde{\tilde{M}} = \tilde{\tilde{N}}_1 \times \cdots \times \tilde{\tilde{N}}_n$. Зададим отображение $\psi : M \to M \times \tilde{\tilde{M}}$ следующим образом. Отображение ψ переводит объект $S = (a_1, \ldots, a_n)$ из M в объект $\psi(S) = (a_1, \ldots, a_n, a_{n+1}, \ldots, a_{2n})$ из $M \times \tilde{M}$, в котором $a_{j+n} = a_j$ при $j \in \{1,2,\ldots,n\}$. Иными словами, признаковое описание объекта S дублируется с обратным отношением порядка.

Пусть $\psi(A), A \subset M$ — образ A при отображении ψ. Имеет место следующая

Теорема 2.2.1. *Если классы множества M не пересекаются, то любой прецедент из класса $\psi(K)$ порождает тупиковый представительный* ***ЭК*** *класса $\psi(K)$.*

Доказательство. Пусть $S \in R(K)$. Покажем, что элемент $\psi(S)$ не принадлежит $\psi(R(\overline{K})^+)$ (независим от $\psi(R(\overline{K}))$ в $\psi(M)$). Действительно, пусть $\psi(S') \preceq \psi(S)$, $S' \in R(\overline{K})$. Тогда имеем: $S' \preceq S$ и $S \preceq S'$. Откуда в силу антисимметричности заданного отношения частичного порядка получаем $S' = S$, что противоречит условию $R(\overline{K} \cap R(K) = \varnothing$. В силу конечности существует максимальный независимый от $\psi(R(\overline{K}))$ элемент $\psi(S'')$, такой, что $\psi(S) \preceq \psi(S'')$. Отсюда следует утверждение теоремы. ∎

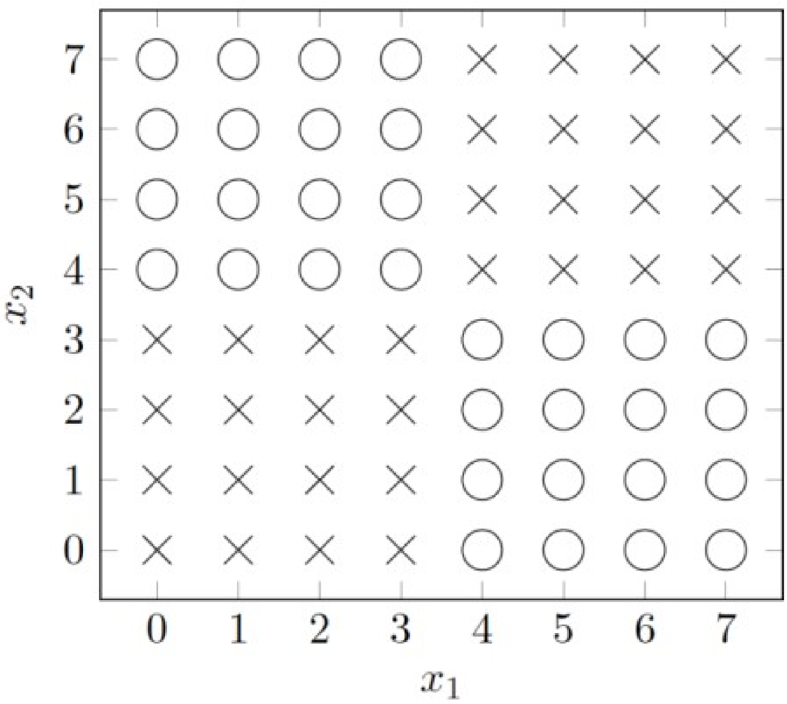

Рисунок 2.1 — Модельный пример 1

На рисунках 2.1 и 2.2 представлены модельные примеры, наглядно демонстрирующие преимущество более общей постановки логической классификации. Рассматриваются две задачи классификации, каждая из которых с двумя классами и системой признаков $\{x_1, x_2\}$. Обучающие объекты классов K_1 и K_2 изображены соответственно в виде крестиков и в виде кружочков. Множество допустимых значений каждого признака — множество целых чисел.

Рассмотрим первый пример (рисунок 2.1). Нетрудно видеть, что для любого значения признака x_1 число объектов каждого класса, у которых признак x_1 принимает данное значение, равно четырём. То же верно и для признака x_2. В классическом варианте каждый **ЭК** ранга 2 является корректным либо для класса K_1, либо для класса K_2, и только один объект из обучающей выборки его порождает. Поэтому любой корректный **ЭК** любого из двух классов обладает низкой информативностью. Такой **ЭК** не порождается ни одним объектом, не входящим в обучающую выборку.

Схожая ситуация и со вторым примером (рисунок 2.2). Существует всего два корректных **ЭК** ранга 1, а именно, **ЭК** $((0), x_1)$ и $((0), x_2)$, и каждый из них порождаются лишь несколькими объектами класса K_2.

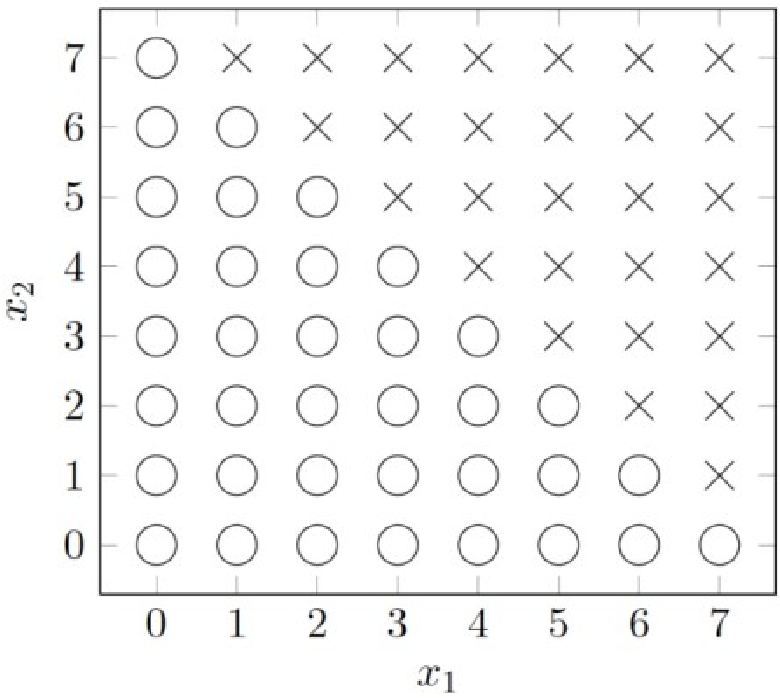

Рисунок 2.2 — Модельный пример 2

Введём на множестве допустимых значений признаков x_1 и x_2 естественный порядок $(0 \prec 1 \prec 2 \prec \dots \prec 7)$. И в первом, и во втором примере будем рассматривать преобразованное множество $\psi(M)$. Объекты из множества M описываются признаками x_1, x_2, $\tilde{\tilde{x}}_1$, $\tilde{\tilde{x}}_2$.

Нетрудно видеть, что в модельном примере 1 тупиковыми представительными для класса K_1 будут **ЭК** $((3{,}3{,}0{,}0), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$ и **ЭК** $((7{,}7{,}4{,}4), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$. Аналогично для класса K_2 тупиковыми представительными будут **ЭК** $((7{,}3{,}4{,}0), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$ и **ЭК** $((3{,}7{,}0{,}4), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$. Каждый из этих **ЭК** содержит 16 обучающих объектов.

В модельном примере 2 тупиковыми представительными для класса K_1 будут сразу 8 **ЭК**. Это все **ЭК** вида $((i{,}7 - i{,}0{,}0), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$, где $i = 0{,}1, \dots, 7$. Аналогично для класса K_2 тупиковыми представительными будут все **ЭК** вида $((7{,}7{,}i{,}8 - i), \{x_1, x_2, \tilde{x_1}, \tilde{x_2}\})$, где $i = 1{,}2, \dots, 7$. Указанные **ЭК** содержат от семи до двадцати обучающих объектов.

2.3 Стохастические композиции процедур CVP над произведением частичных порядков

На данный момент одной из основных техник для улучшения качества прогнозирования произвольного классификатора является построение стохастических композиций (ансамблей) классификаторов. Опишем основные алгоритмы стохастических композиций.

В 1990 году Лео Брейнман описал метод с названием *бэггинг* [52]. Под данным названием этот метод наиболее известен сегодня. Основная идея бэггинга заключается в усреднении ответов базовых алгоритмов, каждый из которых обучался на своей обучающей выборке. Обучающая выборка для каждого базового алгоритма генерируются из одной основной путем случайного равномерного сэмплирования объектов с возвращением (бутстрэп). Использование метода бутстрэпа позволяет уменьшить скоррелированность ответов алгоритмов, что, как показано в [52], позволяет уменьшить дисперсию прогноза.

Ещё один способ снизить скоррелированность базовых классификаторов — обучать классификаторы на разных обучающих выборках, полученных из исходной путём отбрасывания случайного подмножества признаков. Данный метод получил название *метод случайных подпространств*.

Методы ансамблирования, усредняющие значения базовых классификаторов и использующие для их обучения различные случайные подмножества объектов или признаков обучающей выборки, предложены в отечественных работах в семидесятых годах прошлого века. Основной целью этих исследований было сокращение времени счета, связанного с низкой вычислительной эффективностью ЭВМ.

На основе идеи бэггинга в [9] предложены и обоснованы стохастические композиции тестовых алгоритмов. Показано, что при определённых условиях стохастическая модель тестового алгоритма даёт почти всегда (для почти всех обучающих выборок определенного размера) такое же качество классификации, как и детерминированный тестовый алгоритм, использующий для обучения полную выборку.

В [38] метод случайных подпространств успешно использован для построения стохастических моделей тестовых алгоритмов. Предложена оценка минимального количества тупиковых тестов, необходимого для представительности построенной выборки из множества тупиковых тестов. Обе работы направлены на уменьшение времени работы алгоритмов при сохранении качества.

С развитием технических возможностей ЭВМ данные идеи продолжают развиваться в сторону улучшения качества классификации. Одним из наиболее эффективных алгоритмов, использующих ансамблирование, является случайный лес. Случайный лес — алгоритм ансамблирования над решающими деревьвями, использующий одновременно идеи и метода случайных подпространств, и бэггинга. При этом метод случайных подпространств используется для построения каждого узла внутри одного дерева. Алгоритм был предложен в [53].

Другой популярный метод построения стохастических композиций — *бустинг*. В отличие от бэггинга и метода случайных подпространств, в бустинге прогнозы базовых классификаторов сильно скоррелированы друг с другом, поскольку каждый следующий классификатор строится так, чтобы компенсировать ошибки предыдущего.

Алгоритмы бэггинга и метод случайных подпространств основном применяются для классификаторов с низким смещением, поскольку обладают свойством понижения дисперсии прогноза. Алгоритм бустинга в свою очередь уменьшает смещение прогноза, но при этом увеличивает дисперсию. Поэтому в основном бустинг используется для ансамблирования слабых, сильно недообученных классификаторов.

В бустинге для поиска очередного классификатора, минимизирующих ошибку предыдущего, часто бывает необходимо использовать приближённые методы оптимизации. *Градиентный бустинг* — это разновидность бустинга с дифференцируемой функцией потерь, в котором для минимизации ошибки следующий классификатор учится давать прогнозы в направлении уменьшения функции потерь ансамбля, выраженное антиградиентом.

Приведём формальное описание классического алгоритма градиентного бустинга. Пусть для каждого S_i из множества обучающих объектов задан номер y_i класса K_{y_i}, которому он принадлежит. Обозначим $L(F(S), y)$ — дифференцируемая функция потерь, f — базовый классификатор и $c \in \mathbb{R}$ некоторая константа, $F_{k-1}(S) = \sum_{j=1}^{k-1} c_j \cdot f_j(S)$ — ансамбль первых $k-1$ классификаторов. В обычном алгоритме бустинга на k-ом шаге очередной базовый классификатор f_k подбирается по формуле

$$f_k, c_k = argmin_{f,c} \sum_{i=1}^{m} L(F_{k-1}(S_i) + c \cdot f(S_i), y_i). \tag{2.1}$$

На практике поиск решения задачи 2.1 может быть невозможен или сильно затруднён. В градиентном бустинге очередной классификатор f_k обучается на новой обучающей выборке $\{S_i, z_i\}_{i=1}^{m}$, где $z_i = -\frac{\partial L(r,y_i)}{\partial r}\Big|_{r=F_{k1}(S_i)}$, $i = 1,2,\ldots,m$. То есть каждый следующий потерь с длиной шага c_k.

Параметр $c_k \in \mathbb{R}$ подбирается с помощью методов скалярной оптимизации $c_k = argmin_c \sum_{i=1}^{m} L(F_{k-1}(S_i) + c \cdot f_k(S_i), y_i)$.

Для описания предлагаемого алгоритма градиентного бустинга над процедурами CVP будем рассматривать задачу бинарной классификации. Задача многоклассовой классификации может быть сведена к бинарной классификации различными способами. Например, при помощи оценок принадлежности объекта классу и процедуре голосования.

Поскольку алгоритмы бустинга ориентированы на использование в качестве базового классификатора простых прогнозаторов с низкой дисперсией, будем использовать классификатор, состоящий из одного **ЭК**, и определяемый функцией близости:

$$f(S,(\sigma, H), t_1, t_2) = \begin{cases} t_1, & \text{если } \tilde{B}(\sigma, S, H) = 0, \\ t_2, & \text{если } \tilde{B}(\sigma, S, H) = 1, \end{cases}$$

где (σ, H) — **ЭК** и t_1, $t_2 \in \mathbb{R}$ — параметры модели, $S \in M$ — классифицируемый объект.

В качестве функции потерь используется стандартная логистическая функция:

$$L(y, F_k) = -frac1m \sum_{i=1}^{m} log(sigmoid(y_i \cdot F_k(S_i)),$$
$$sigmoid(r) = \frac{1}{1+e^{-r}},$$

где $F_k(\cdot)$ — ансамбль из классификаторов, полученный после k-го шага, возвращающий оценку принадлежности классу 1.

При этом возникает следующая проблема — базовый прогнозатор $f(S, (\sigma, H), t_1, t_2)$ не дифференцируем по своим параметрам, что не позволяет применять стандартные методы градиентной оптимизации. Для поиска параметров функции $f(S, (\sigma, H), t_1, t_2)$ предлагается два подхода.

В первом подходе настройка f_k производится с помощью генетического алгоритма перебором возможных значений для $(\sigma, H), t_1, t_2$. Поиск параметра σ_j, $j = 1, 2, \ldots, n$, осуществлялся в диапазоне от $[0, k_j]$, где k_j — наибольший элемент множества N_j. Признак $x_j \in H$, если $\sigma_j \prec k_j$. Интервал возможных значений параметров t_1, t_2 выбран $[-\frac{2}{K}, \frac{2}{K}]$, где K — число базовых моделей в алгоритме градиентного бустинга.

Второй подход использует приближение градиента функции потерь L_{base} его несмещенной оценкой, предложенной в [72]. Такая оценка используется для оптимизации по параметру σ с фиксированными t_1, t_2 с помощью стохастического градиентного спуска. Оптимизация по t_1, t_2 производится после с помощью генетического алгоритма.

Также как и в первом подходе будем искать n значений σ_j, $j = 1, 2, \ldots, n$, где $\sigma_j = k_j$ означает, что $x_j \notin H$ (т.е. фактически осуществляется поиск объекта $S(\sigma, H)$). Предполагается, что значения σ принадлежат некоторому распределению с параметрами θ и оптимизируется математическое ожидание функции потерь по этому распределению $\mathbb{E}_p(\sigma|\theta) L_{base}(S, z, \sigma)$. Простейшей несмещённой оценкой на градиент является оценка REINFORCE:

$$g_{REINFORCE}(\mathbb{E}_p(\sigma|\theta) L_{base}(S, z, \sigma)) = L_{base}(S, z, \sigma) \cdot \frac{\partial logp(\sigma|\theta)}{\partial \theta},$$

где $\sigma \sim p(\sigma|\theta)$. Недостатком данной оценки является высокая дисперсия.

Ещё один способ получения несмещённой оценки градиента — репараметризация распределения. С ее помощью можно получить оценку с низкой дисперсией. Пусть для исходного распределения $\sigma \sim p(\sigma|\theta)$ существует такая функция $T(\theta, \epsilon)$, и какое-то распределение $p(\epsilon)$, не зависящее от параметров модели, такое что если $\epsilon \sim p(\epsilon)$, то $\sigma = T(\theta, \epsilon) \sim p(\sigma|\theta)$. Тогда оценка на градиент имеет следующий вид:

$$g_{REINFORCE}(L_{base}(S, z, \sigma)) = L_{base}(S, z, \sigma) \cdot \frac{\partial logp(\sigma|\theta)}{\partial T(\theta, \epsilon)} \cdot \frac{\partial T(\theta, \epsilon)}{\partial \theta}.$$

В [72] теоретически обоснована несмещённость оценки, которая комбинирует в себе оба подхода и обладает низкой дисперсией:

$$g^{LAX} = [L_{base}(S, z, \sigma) - c_w(\sigma)] \frac{\partial logp(\sigma|\theta)}{\partial \theta} + \frac{\partial c_{\phi(\sigma)}}{\partial \theta},$$

где c_w дополнительная нейронная сеть с параметрами w. Нейронная сеть обучается на уменьшение дисперсии оценки градиента. Градиент дисперсии несмещенной оценки по w считается следующим образом:

$$\frac{\partial D(g)}{\partial w} = \mathbb{E} \frac{\partial g^2}{\partial w}$$

Основная проблема подхода с использование репараметризации состоит в том, что не всегда удаётся подобрать распределение для репараметризации. Для построения градиентного бустинга над процедурами CVP было сделано дополнительное предположение, о независимости σ_i от $sigma_j$ при $i \neq j$, $i, j = 1,2, \dots, n$.

Тогда в качестве распределения $p(\sigma|\theta)$ взято полностью факторизованное многомерное нормальное распределение $\prod_{j=1}^{n} \mathcal{N}(\sigma_j|\mu_j, s_j)$, $\theta = \{\mu_j, s_j\}_{j=1}^{n}$, $T(\theta, \epsilon) = \prod_{j=1}^{n}(\mu_j + s_j \cdot \mathcal{N}(\epsilon_j|0, 1))$, $p(\epsilon_j) = \mathcal{N}(\epsilon_j|0, 1))$.

2.4 Результаты экспериментов

Экспериментальное сравнение качества классификации проводилось на пяти наборах данных из www.kaggle.com и archive.ics.uci.edu. Для тестирования применялась методика 3-кратного скользящего контроля, а качество оценивалось мерой AUC ROC. Отдельно сравнивались классические методы CVP с обобщёнными процедурами, а также стохастические композиции обобщённых **ЭК** с классическими композициями над деревьями.

Результаты счёта представлены в таблице 1. В таблице 1 в первом столбце приведено название задачи (датасета). Каждый из следующих четырёх столбцов соответствует одному из вариантов упорядочения значений признаков при голосовании по тупиковым представительным **ЭК** .

Таблица 1 — Сравнение качества классификации модели голосования по представительным эл.кл. при различном упорядочивании признаков

Название датасета	Признаки антицепи	Признаки цепи	Смешанные признаки	Признаки цепи (с обратным порядком)
Car	73%	70%	**84**%	81%
Heart	76%	74%	**81**%	-
Ph	43%	10%	51%	**63**%
Dermatology	**95**%	82%	**95**%	-
Turkey	35%	30%	35%	**39**%

Полученные результаты подтверждают, что классический подход (каждый признак — антицепь) не учитывает зависимости, присутствующие в значениях атрибутов, и поэтому склонен к переобучению. Линейный порядок (цепь) очень чувствителен к выбору вида линейного порядка (возможно $k!$ вариантов цепей для признака значности k). Выбор того, какой признак является цепью, а какой антицепью не является однозначным. Практика показала, что в качестве кандидатов на роль цепей целесообразно рассматривать признаки высокой значности, имеющие естественный линейный порядок (например, возраст). Наличие среди антицепей нескольких цепей (вариант смешанных признаков) в большинстве задач привело к улучшению качества и к ускорению времени работы. Для датасетов Car, Ph и Turkey рассмотрена также эвристика с дублированием признаков с обратным порядком. Нетрудно видеть, что в случае, когда все признаки — цепи, дублирование даёт более хорошие результаты, чем его отсутствие.

В рамках исследования реализован бэггинг над процедурами CVP. Для бэггинга над процедурами CVP использовался подход с дублированием признаков с обратным порядком, а также метод случайных подпространств. Оценка принадлежности распознаваемого объекту S классу K определялась посредством усреднения оценок принадлежности объекта S классу K, полученных каждым алгоритмом голосования по тупиковым представительным **ЭК** в ансамбле.

Сравнение бэггинга проводилось с аналогичными композициями решающих деревьев из библиотеки scikit-learn: бэггинг над деревьями (Bagging trees) и случайный лес (Random Forest).

Градиентный бустинг сравнивался с реализациями градиентного бустинга над деревьями из библиотек CatBoost и XGBoost.

Модели из библиотек применялись с гиперпараметрами, найденными путём случайного поиска.

Результаты сравнения бэггинга над **ЭК** со стохастическими композициями над деревьями представлен в таблице 2.

Таблица 2 — Сравнение качества классификации различных стохастических композиций

Название датасета	Бэггинг над **ЭК**	Бэггинг над деревьями	Случайный лес
Car	**0.86**	0.69	0.8
Ph	0.83	**0.85**	**0.85**
Heart	0.77	0.72	**0.8**
Dermatology	0.88	0.86	**0.89**
Performance	0.71	**0.72**	**0.72**
Phishing	0.81	0.8	**0.85**
Cancer	**0.72**	0.67	0.71

Результаты вычислительных экспериментов показывают, что в целом Бэггинг над **ЭК** работает достаточно стабильно и даже в среднем немного превосходит бэггинг над решающими деревьями. Однако также в среднем проигрывает случайному лесу, хотя и показывает лучшее качество на некоторых наборах.

В таблице 3 приведено сравнение алгоритмов бустинга над деревьями и бустинга над **ЭК**. При проведении экспериментов выяснилось, что метод построения базового классификатора при помощи генетического алгоритма даёт лучшее качество классификации, чем комбинированный метод с использованием несмещённых оценок на градиент. Предполагается, что данный эффект вызван слишком строгим предположением о независимости отдельных значений строящегося **ЭК**. В таблице вынесены результаты, соответствующие полностью генетическому подходу.

Таблица 3 — Сравнение качества классификации различных стохастических композиций (бустинг)

Название датасета	CatBoost	XGBoost	Бустинг над **ЭК**
Car	**0.8**	0.72	0.71
Ph	0.84	0.84	**0.85**
Heart	**0.84**	0.82	0.72
Dermatology	**0.91**	0.89	0.85
Performance	0.73	0.74	0.71
Phishing	0.9	0.87	**0.91**
Cancer	**0.79**	**0.79**	0.76

Результаты вычислительных экспериментов показывают, что бустинг над **ЭК** показывает лучшее качество классификации, чем бэггинг над **ЭК**. Однако также в среднем проигрывает бустингу над решающими деревьями. Это можно объяснить тем, что алгоритмы CatBoost и XGBoost содержат большее число эвристик, полученных в результате многолетнего использования алгоритмов бустинга различными исследовательскими группами. В тоже время алгоритм бустинга над **ЭК** создан недавно, и до конца не реализовал свой потенциал.

Глава 3. О выборе частичных порядков на множествах значений признаков

В данной главе рассматривается задача выбора частичных порядков на множествах значений признаков. Предлагаются три процедуры выбора порядков. Первая процедура основана на рассматриваемом в данной главе критерии корректности классификации. Получаемые в результате первой процедуры частичные порядки обеспечивают корректность классификации. Вторая процедура не обладает свойством корректности, но эффективна по времени работы. Она ориентирована на оценку информативности отдельных значений признаков. Ещё один способ выбора частичных порядков заключается в сведении задачи классификации к проблеме оптимизации, и решению этой проблемы градиентными методами.

Основные результаты, преддставленные в этой главе, опубликованы в [24–26, 59]

3.1 Критерий корректности классификации

Пусть $Q(K, S')$ — множество всех представительных **ЭК** из $\mathcal{P}(K)$, которые содержат объект $S' \in M$. Рассмотрим алгоритм A, который строит множество всех тупиковых представительных **ЭК** класса K. В этом случае справедлива

Теорема 3.1.1. *Алгоритм A классифицирует правильно объект S' из $R(K)$ тогда и только тогда, когда S' является независимым от $R(\overline{K})$ элементом множества M.*

Доказательство. *Необходимость.* Если алгоритм A классифицирует объект S' из $R(K)$ правильно, то $Q(K,S') \neq \varnothing$. Значит, объект S' содержит хотя бы один тупиковый представительный **ЭК** (σ,H). Имеем $S' \preceq S_((\sigma,H))$, $S_((\sigma,H))$ — максимальный независимый от $R(\overline{K})$ элемент множества M. Следовательно, S' — независимый от $R(\overline{K})$ элемент множества M.

Достаточность. Пусть $S' \in R(K)$ и S' — независимый от $R(\overline{K})$ элемент множества M. Рассмотрим **ЭК** (σ, H) такой, что $S' = S_((\sigma,H))$. Ни один объект S'' из $R(\overline{K})$ не содержит **ЭК** (σ,H), иначе $S'' \preceq S'$ и, следовательно, S' не является независимым от $R(\overline{K})$ элементом множества M, что противоречит условию утверждения теоремы. Таким образом, S' содержит представительный **ЭК** (σ,H) класса K. **ЭК** (σ,H) содержит хотя бы один тупиковый представительный **ЭК** класса K. Поэтому $Q(K,S') \neq \varnothing$. ■

Таким образом, если для любого K любой прецедент из $R(K)$ является независимым от $R(\overline{K})$ элементом множества M, то алгоритм голосования по всем тупиковым представительным **ЭК**, является корректным. Аналогичный результат имеет место для алгоритма голосования по всем представительным **ЭК**.

Согласно теореме 2.2.1, при любых частичных порядках $N_1, N_2, \ldots, N_n$ добавление признаков с обратными порядками делает корректной процедуру классификации по (тупиковым) представительным **ЭК**. Однако увеличение числа признаков в два раза значительно увеличивает вычислительные затраты.

Отметим, что этап обучения — это самый сложный в вычислительном плане этап логической классификации из-за необходимости решать задачу дуализации над произведением частичных порядков, которая относится к труднорешаемым перечислительным задачам дискретной математики.

На практике порядки на множествах значений признаков могут быть не заданы или заданы «неудачно», демонстрируя не очень высокое качество классификации и не обеспечивая корректность классификации на обучающей выборке. Поэтому актуальны рассматриваемые ниже вопросы эффективного задания частичных порядков $N_1, \ldots, N_n$.

3.2 Быстрая процедура независимого линейного упорядочения значений признаков

В данном разделе описана процедура выбора линейных порядков на множествах значений признаков, эффективная по времени вычислений и позволяющая повысить качество классификации, но не обеспечивающая корректность классификации.

Предлагаемая процедура основана на оценке информативности отдельных значений признаков по методу, идейно близкому к методике предварительного анализа обучающей выборки, предложенной в [27]. Пусть $\mu_{ij}^{(1)}(a)$, $i \in \{1,2,\ldots,l\}$, $j \in \{1,2,\ldots,n\}$, $a \in N_j$, — доля прецедентов класса K_i, у которых признак x_j принимает значение a; $\mu_{ij}^{2}(a)$, $i \in \{1,2,\ldots,l\}$, $j \in \{1,2,\ldots,n\}$, $a \in N_j$, — доля прецедентов не из класса K_i, у которых признак x_j принимает значение a. Величину $\mu_{ij}(a) = \mu_{ij}^{1}(a) - \mu_{ij}^{2}(a)$ назовем весом значения a признака x_j в классе K_i. Эта величина служит мерой близости значения a признака x_j классу K_i. Для каждого класса $K_i, i \in \{1,2,\ldots,l\}$, и для каждого признака x_j, $j \in \{1,2,\ldots,n\}$, зададим следующий линейный порядок: $\forall y,z \in N_j$, $y \preceq z$ тогда и только тогда, когда $\mu_{ij}(y) \geq \mu_{ij}(z)$.

Рассмотрим $S' = (a_1,\ldots,a_n) \in R(K_i)$, $i \in \{1,2,\ldots,l\}$. Предположим, что после задания порядка данный объект классифицирован неправильно. Тогда, согласно теореме 3.1.1 существует объект $S'' = (b_1,\ldots,b_n) \in R(\overline{K_i})$ такой, что $\mu_{ij}(a_j) \leq \mu_{ij}(b_j)$ при $j = 1,2,\ldots,n$. Следовательно, признаковые значения объекта S' недостаточно характерны для класса K_i, то есть объект S' не является типичным для K_i. Таким образом, задание порядка описанным способом делает модель голосования по представительным **ЭК** менее чувствительной к выбросам. Очевидный недостаток процедуры заключается в том, что порядок на множестве значений каждого признака выбирается независимо от выбора порядков для других признаков.

3.3 Процедура корректного упорядочения значений признаков

Пусть для любого класса K, $K \in \{K_1, \ldots, K_l\}$ алгоритм A строит множество всех тупиковых представительных **ЭК** класса K. Частичный порядок на M назовем (A,K)-корректным, если алгоритм A правильно классифицирует каждый объект из $R(K)$. Согласно теореме 2.2.1, частичный порядок на M является (A,K)-*корректным* тогда и только тогда, когда для любого объекта S', $S' \in R(K)$, и любого объекта S'', $S'' \in R(\overline{K})$, либо S' и S'' несравнимые, либо $S' \prec S''$ (запись $S'' \not\preceq S'$).

Построим булеву матрицу B_K. Каждой паре объектов (S',S''), где $S' \in R(K)$ и $S'' \in R(\overline{K})$, соответствует строка в матрице B_K. Каждому признаку x_j, $j \in \{1,2,\ldots,n\}$, и каждой паре $a,b \in N_j$, $a \neq b$, соответствует столбец матрицы B_K. Элемент матрицы B_K, расположенный на пересечении строки (S', S'') и столбца (x_j,a,b), равен 1, если значение признака x_j равно a и b у объектов S' и S'' соответственно.

Согласно определению из первой главы, набор столбцов H матрицы B_K называется покрытием, если каждая строка матрицы B_K в пересечении хотя бы с одним из столбцов, входящих в H, дает 1 (каждая строка матрицы B_K покрыта хотя бы одним столбцом из H). Покрытие матрицы B_K называется неприводимым, если любое его собственное подмножество покрытием не является. Имеет место

Теорема 3.3.1. *Частичный порядок, заданный на множестве M, является (A,K)-корректным тогда и только тогда, когда существует покрытие H матрицы B_K такое, что $\forall j \in \{1,2,\ldots,n\}$ и $\forall a,b \in N_j$, $b \prec a$, столбец (x_j,a,b) не входит в H.*

Доказательство. *Необходимость.* Предположим противное. Пусть существуют $j \in \{1,2,\ldots,n\}$ и $a,b \in N_j$, $b \prec a$, такие, что столбец (x_j,a,b) входит во все покрытия матрицы B_K. Тогда найдется строка (S', S'') матрицы B_K, которая покрыта столбцом (x_j,a,b) и не покрыта всеми остальными столбцами матрицы B_K, поскольку в противном случае множество $\mathcal{H}$ всех столбцов матрицы B_K кроме столбца (x_j,a,b) является покрытием. При этом из построения матрицы B_K следует, что объекты S' и S'' отличаются только признаком x_j, и этот признак принимает значения a и b у объектов S' и

S'' соответственно. Из условия $b \prec a$ следует $S'' \prec S'$, что противоречит (A,K)-корректности частичного порядка.

Достаточность. Рассмотрим покрытие H матрицы B_K, удовлетворяющее условию теоремы. Тогда для любой строки (S',S'') матрицы B_K в H найдется столбец вида (x_j,a,b), покрывающий строку (S',S''). Это означает, что признак x_j принимает значения a и b у объектов S' и S'' соответственно и $b \not\preceq a$. Следовательно, $S'' \not\preceq S'$. Отсюда в силу произвольности выбора объектов S' и S'' следует (A,K)-корректность частичного порядка. ■

Пусть множество значений каждого признака x_j, $j = 1, 2, \ldots, n$, упорядочено линейно. Тогда получившийся частичный порядок на множестве объектов из M будем называть *линейным* на множестве M.

Рассмотрим (A, K)-корректный линейный порядок на множестве M и покрытие H из утверждения теоремы 3.3.1 Согласно данной теореме, для любого столбца $(x_j, a, b) \in H$, $a,b \in N_j$, не выполнено $b \prec a$. Поскольку множество N_j является цепью, то $a \prec b$. Поэтому справедливо

Следствие 3.3.1. *Линейный порядок на множестве M является (A,K)-корректным тогда и только тогда, когда существует покрытие H матрицы B_K такое, что $a \prec b$ для любого столбца (x_j,a,b) из покрытия H.*

Пусть множество значений каждого признака x_j, $j = 1, 2, \ldots, n$, упорядочено антилинейно. Тогда получившийся частичный порядок на множестве объектов из M будем называть *антилинейным* на множестве M.

Рассмотрим (A, K)-корректный антилинейный порядок на множестве M. Для всех $j = 1, 2, \ldots, n$ и для всех $a,b \in N_j$ не выполнено $b \prec a$, поскольку N_j — антицепь. Следовательно для любого покрытия матрицы B_K выполнены условия теоремы 3.3.1. Поэтому справедливо

Следствие 3.3.2. *Антилинейный порядок на множестве M является (A,K)-корректным для любого класса K из $\{K_1, \ldots, K_l\}$.*

Рассмотрим в качестве конкретного примера, иллюстрирующего приведённую теорему, задачу классификации с двумя классами K_1 и K_2 и двумя прецедентами $S_1 = (0, 1, 0)$ и $S_2 = (1, 0, 1)$, где $S_1 \in K_1$, $S_2 \in K_2$. Построим матрицу B_{K_1}.

Матрица B_{K_1} состоит из одной строки, соответствующей паре объектов S_1, S_2, и шести столбцов. Каждый из столбцов соответствует признаку x_j, $j = 1, 2, 3$, и паре значений $(0, 1)$ или $(1, 0)$. Элементы матрицы B_{K_1} в столбцах $(x_1, 0, 1)$, $(x_2, 1, 0)$ и $(x_3, 0, 1)$ равны 1. Остальные элементы матрицы B_{K_1} равны 0.

$$B_{K_1} = \left(\overbrace{1\ 0}^{x_1} \overbrace{0\ 1}^{x_2} \overbrace{1\ 0}^{x_3} \right)$$

Очевидно, что набор столбцов H матрицы B_{K_1} является покрытием тогда и только тогда, когда он содержит хотя бы один из столбцов $(x_1, 0, 1)$, $(x_2, 1, 0)$ и $(x_3, 0, 1)$. Следовательно, согласно теореме 3.3.1 частичный порядок на множестве M не будет (A, K_1)-корректным в том и только в том случае, если одновременно $1 \prec 0$ для x_1, $0 \prec 1$ для x_2 и $1 \prec 0$ для x_3.

Эти условия задают линейный частичный порядок на множестве M, при котором $S_2 \prec S_1$, и по теореме 3.1.1 алгоритм A не может верно классифицировать объект S_1.

Построим с помощью матрицы B_{K_1} (A, K_1)-корректный частичный порядок на множестве M. Для этого рассмотрим покрытие $H = \{(x_1, 0, 1),\ (x_2, 1, 0),\ (x_3, 0, 1)\}$ матрицы B_{K_1}. Данное покрытие согласно следствию 3.3.1 задаёт частичный порядок, в котором $0 \prec 1$ для x_1, $1 \prec 0$ для x_2 и $0 \prec 1$ для x_3.

Эти условия задают линейный частичный порядок на множестве M, при котором $S_1 \prec S_2$, и по теореме 3.1.1 алгоритм A верно классифицирует объект S_1.

Рассмотрим в качестве ещё одного примера, иллюстрирующего приведённую теорему 3.3.1, задачу классификации с двумя классами K_1 и K_2 и четырьмя прецедентами $S_1 = (1, 0)$, $S_2 = (2, 1)$, $S_3 = (1, 1)$ и $S_4 = (0, 0)$, где $S_1, S_2 \in K_1$, $S_3, S_4 \in K_2$. Построим матрицу B_{K_1}. Она состоит из четырёх строк и восьми столбцов.

В каждой строке (S_1, S_3), (S_1, S_4) и (S_2, S_3) ровно один элемент равен 1. Это элемент в столбце $(x_2, 0, 1)$ для строки (S_1, S_3), элемент в столбце $(x_1, 1, 0)$ для строки (S_1, S_4) и элемент в столбце $(x_1, 2, 1)$ для строки (S_2, S_3). В строке (S_2, S_4) ровно два элемента равны 1: элемент в столбце $(x_1, 2, 0)$ и элемент в столбце $(x_2, 1, 0)$.

$$B_{K_1} = \begin{pmatrix} \overbrace{0\;0\;0\;0\;0\;0}^{x_1} & \overbrace{1\;0}^{x_2} \\ 0\;0\;1\;0\;0\;0 & 0\;0 \\ 0\;0\;0\;0\;0\;1 & 0\;0 \\ 0\;0\;0\;0\;1\;0 & 0\;1 \end{pmatrix}$$

Набор столбцов $H_1 = \{(x_1,1,0),(x_1,2,0),(x_1,2,1),(x_2,0,1)\}$ является покрытием матрицы B_{K_1}. Согласно следствию 3.3.1 линейный на множестве M порядок такой, что $1 \prec 0$, $2 \prec 1$, $2 \prec 0$ для x_1, и $0 \prec 1$ для x_2, будет (A,K_1)-корректным. Если на множестве N_1 задать линейный порядок $2 \prec 1 \prec 0$ и на множестве N_2 задать линейный порядок $0 \prec 1$, то полученный линейный на множестве M порядок удовлетворяет всем условиям следствия 3.3.1 и является (A,K_1)-корректным.

Набор столбцов $H_2 = \{(x_1,1,0),(x_1,2,1),(x_2,0,1),(x_2,1,0)\}$ также является покрытием матрицы B_{K_1}. Он задаёт набор условий, среди которых есть $0 \prec 1$ и $1 \prec 0$ для x_2. Очевидно, не существует удовлетворяющего этим условиям упорядочения множества N_2.

Рассмотренный пример доказывает, что не для каждого набора ограничений существует линейный порядок, который им удовлетворяет. Имеет смысл искать покрытия с небольшим числом столбцов. При этом, если для признака x_j, $j \in \{1,2,\dots,n\}$, не удалось построить линейный порядок, то на множестве N_j может быть установлен антилинейный порядок, который всегда удовлетворяет ограничениям согласно следствию 3.3.2.

Основная проблема при построении (A,K)-корректного частичного порядка заключается в вычислительной сложности перечисления неприводимых покрытий булевой матрицы [1, 8]. Для увеличения числа признаков с линейными порядками можно использовать приближенные методы поиска минимального покрытия.

При проведении экспериментов построение (A,K)-корректного частичного порядка осуществлялось с применением генетического алгоритма, ориентированного на приближенное решение задачи поиска минимального покрытия.

Пусть $\delta(B_K,H) = 0$ если H — покрытие матрицы B_K, и $\delta(B_K,H) = 1$ в противном случае. Функция $\delta_j(H) = 0$ если множество N_j может быть линейно упорядочено, согласно условиям следствия 3.3.1, и $\delta_j(H) = 1$ в противном случае. Тогда

$$f(H) = \delta(B_K, H) + \frac{1}{2n}\left(\sum_{j=1}^{n}(\delta_j(H)) + |H|\right). \qquad (3.1)$$

Функция $f(H)$ принимает минимальное значение, если H является покрытием матрицы B_K, при этом число признаков, которые возможно линейно упорядочить, максимально, число столбцов в H минимально. Использование функции $f(H)$ в генетическом алгоритме для оценки выживаемости позволяет находить покрытия матрицы B_K, по которым возможно построение частичных порядков на множествах значений признаков с высокой долей линейных порядков.

После завершения работы генетического алгоритма последовательно рассматривался каждый признак.

С применением алгоритма Тарьяна [90] осуществлялось упорядочение ориентированного графа, вершинами которого служили значения из N_j, а каждому ребру (a,b), $a,b \in N_j$, соответствовал столбец (x_j,a,b). При этом обход вершин осуществлялся согласно порядку, определяемому процедурой быстрого линейного упорядочения описанной в предыдущем разделе. Если в результате применения алгоритма получался линейный порядок, это означало, что N_i — цепь. В противном случае N_j — антицепь.

Стоит отметить, что из-за сильной разреженности матрицы B_K применение жадного алгоритма для поиска минимального покрытия матрицы B_K оказалось менее эффективным.

3.4 Процедура градиентного выбора частичного порядка

Ещё один подход к выбору частичных порядков основан на сведении классификации к задаче оптимизации некоего дифференцируемого функционала.

Обозначим $\mathcal{A}_\Theta$ — параметрическое семейство алгоритмов классификации со множеством значений параметров Θ. Пусть A_θ — алгоритм из семейства $\mathcal{A}_\Theta$ с параметрами θ. Тогда часто задачу классификации формально представляют в виде минимизации некоторого функционала $\mathcal{L}(S_1, \ldots, S_m, y_1, \ldots, y_m, \theta)$ по всем $\theta \in \Theta$, где $y_1, \ldots, y_m$ номера классов соответствующих прецедентов $S_1, \ldots, S_m$.

Часто в качестве функционала $\mathcal{L}$ выступает функционал эмпирического риска. Пусть $L(A_\theta(S), y)$ — функция близости ответа алгоритма $A_\theta(S)$ и метки класса y. Тогда функционалом эмпирического риска называют $\frac{1}{m}\sum_{i=1}^{m} L(A_\theta(S_i), y_i)$ — среднее значение ошибки на обучающей выборке.

Пусть $\precneqq = \precneqq_1 \times \cdots \times \precneqq_n$, где $\precneqq_i$ — множество всевозможных линейных порядков на множестве N_i, $i \in \{1, \ldots, n\}$. Пусть также $\Omega(M) = \Omega(N_1) \times \cdots \times \Omega(N_n)$, где $\Omega(N_i)$ является множеством всех биекций множества N_i на себя, $i \in \{1, \ldots, n\}$. Будем обозначать $\omega = \omega_1 \times \cdots \times \omega_n \in \Omega(M)$, если $\omega_i \in \Omega(N_i), i \in \{1, \ldots, n\}$, и $\forall S = (a_1, \ldots, a_n) \in M$ $\omega(S) = (\omega_1(a_1), \ldots, \omega_n(a_n))$. Будем также считать, что на множестве M задан некий порядок $\preceq_0 \in \precneqq$. Тогда справедливо следующее

Утверждение 3.4.1. *Для любого частичного порядка $\preceq \in \precneqq$ существует единственная биекция $\omega \in \Omega(M)$ такая, что для всех пар S',$S'' \in M$ имеет место $S' \preceq S'' \Leftrightarrow \omega(S') \preceq_0 \omega(S'')$*

Данное утверждение по сути задаёт взаимно однозначное соответствие между множествами Ω и $\precneqq$. На основании сказанного задачу выбора порядка можно решать в виде:

$$\mathcal{L}(\omega(S_1), \ldots, \omega(S_m), y_1, \ldots, y_m, \theta) \to \min_{\theta,\omega} \tag{3.2}$$

Решением задачи 3.2 будет некий набор параметров $\theta^* \in \Theta$, определяющий алгоритм классификации, и отображение $\omega^* \in \Omega(M)$. Согласно утверждению 3.4.1 найденному отображению ω^* соответствует единственный порядок $\preceq \in \precneqq$. Данный порядок будем называть *оптимальным*. Заметим, что поставленная задача совпадает со стандартной задачей классификации в оптимизационной форме, если принять $\Omega(M) = \{I\}$, где I — тождественный оператор.

Замечание. Формально, для постановки задачи 3.4.1 необходимо задать порядок $\preceq_0 \in \underline{\prec}$, однако нетрудно видеть, что решение задачи **??** не зависит от выбора частичного порядка $\preceq_0$, поэтому его можно выбрать случайным образом либо при помощи одной из описанных выше процедур. Данное свойство может нарушаться, если также рассматривать порядки, не являющиеся произведением цепей.

Наивным подходом в решении задачи 3.2 является решение путём полного перебора всевозможных $\omega \in \Omega(M)$, и решение задачи классификации в классической постановке при каждом фиксированном ω. Данный подход теоретически возможен, поскольку каждое множество N_i является конечным, однако на практике он невозможен из-за большого числа всевозможных упорядочений множества M, которое равно $|N_1|! \times \dots \times |N_n|!$.

Более эффективной стратегией является жадный поиск оптимального решения. Данная стратегия представляет из себя итерационный процесс: на i-ой итерации решается задача минимизации функционала $\mathcal{L}(\omega_{<i}(S_1), \dots, \omega_{<i}(S_m), y_1, \dots, y_m, \theta)$ по всем θ и ω_i, где $\omega_{<i} = \omega_1^* \times \dots \times$ $\times \omega_{i-1}^* \times \omega_i \times I \times \dots \times I$, $\omega_1^*, \dots, \omega_{i-1}^*$ — решения подученные на предыдущих итерациях, I — тождественное отображение. Решается данная задача путём полного перебора $|N_i|!$ вариантов. После n-ой итерации будет получено отображение $\omega^* \in \Omega(M)$. При этом, был осуществляется просмотр $|N_1|! + \dots + |N_n|!$ вариантов, что существенно меньше, чем при наивном подходе. Однако найденное ω^* не всегда будет являться решением задачи 1. Дальнейшее уменьшение трудоёмкости вычислений возможно при наложении дополнительных ограничений на задачу 3.2.

Наложим следующие ограничения на задачу 3.2 Пусть $\mathcal{L}$ является функционалом эмпирического риска. Тогда задача 3.2 может быть представлена в виде:

$$\frac{1}{m} \sum_{i=1}^{m} L(A_\theta(\omega(S_i)), y_i) \to \min_{\theta, \omega} \tag{3.3}$$

Пусть функция близости L и алгоритм A_θ являются непрерывно дифференцируемыми. Например, L — кроссэнропия, A_θ — линейный классификатор. Тогда, в большинстве случаев, наиболее эффективным способом оптимизации функционала эмпирического риска по параметрам θ будет градиентная оптимизация. В рамках данной работы на основе теории Синхорна-Кноппа [89] разработан метод позволяющий использовать градиентный спуск и при оптимизации функционала эмпирического риска по ω.

Пусть $N = \{0, \ldots, k-1\}$ — множество значений произвольного признака $x \in \{x_1, x_2, \ldots, x_n\}$. Известно, что $\Omega(N)$ изоморфно группе перестановок элементов множества N. В свою очередь, группа перестановок множества N изоморфна множеству перестановочных матриц размера $|N| = k$. Данное множество обозначается $\mathcal{P}_k$.

Пусть $a \in N$. Обозначим $arange(k) = (0, 1, \ldots, k-1)^T$ и $OH(a) = \underbrace{(0, \ldots, 0, \overbrace{\;1\;}^{a}, 0, \ldots, 0)}_{k}^T$ — *one-hot* преобразование значения a признака x. Тогда очевидно следующее утверждение:

Утверждение 3.4.2. *Произвольное $\omega \in \Omega(N)$, соответствующее перестановке, заданной матрицей $P \in \mathcal{P}_k$, $\forall x \in N$ представимо в виде:*

$$\omega(x) = \omega_P(x) = arange(k)^T \cdot P \cdot OH(x) \tag{3.4}$$

Следствие 3.4.1. *Пусть $W \in \mathbb{R}^{k \times k}$, $P = P(W) \in \mathcal{P}_k$ и P дифференцируем по W. Тогда $\omega = \omega_P$ дифференцируемо по W и $d\omega = \langle arange(k)^T \cdot OH(x), dP \rangle$.*

Таким образом, было показано, что задача 3.3 допускает градиентную оптимизацию по ω в случае, когда существует параметризация множества перестановочных матриц вещественными матрицами. Далее речь пойдёт о примере такой параметризации.

Пусть $\mathcal{B}_k$ — многогранник Бирхоффа (множество всех дважды стохастических матриц). В [89] был предложен оператор $Sink : \mathbb{R}^{k \times k} \to \mathcal{B}_k$. Оператор Синхорна может рассматриваться как предел бесконечного итерационного процесса, состоящего из поочередного нормирования столбцов и строк матрицы, к каждому элементу которой была применена операция потенциирования. Для оператора $Sink$ и матрицы W из абсолютно непрерывного распределения с вероятностью 1 верно следующее равенство:

$$\lim_{\tau \to 0+} Sink(W/\tau) = arg\max_{P \in \mathcal{P}_k} \|W - P\|_F$$

Здесь $\| \cdot \|_F$ является нормой Фроббениуса, а число τ принято называть температурой.

В [85] предложен усечённый оператор Синхорна, представляющий из себя конечный итерационный процесс обычного оператора Синхорна. Обозначим S^q_τ — усечённый оператор Синхорна с числом итераций q и температурой τ. Тогда, при достаточно больших q и достаточно малых τ, будем считать, что для произвольной матрицы $W \in \mathbb{R}^{k \times k}$ $S^q_\tau(W) = B \approx P \in \mathcal{P}_k$.

Формально $S^q_\tau(W) = B$ нельзя использовать для определения отображения ω_B, поскольку $\exists B \in \mathcal{B}_k$ и $\exists x \in N$ такие, что $\omega_B(x) = arange(k)^T \cdot B \cdot OH(x) \notin N$. Пусть $B[i]$ — i-ая строка матрицы B. По определению усечённого оператора Синхорна, элементы строки $B[i]$ неотрицательны, и их сумма равна единице, $i \in \{1, \dots, k\}$. Обозначим $P \sim Discrete_{row}(B)$, если $P[i] \sim Discrete(B[i]) \; \forall i = \overline{1,k}$, где $Discrete(B[i])$ — дискретное вероятностное распределение с вероятностями $B[i]$. Тогда, если $P \sim Discrete_{row}(B)$, то $\omega_P(x) \in N$, $\forall x \in N$.

Пусть $\overline{P} = sg(P - B) + B$, где $P \sim Discrete_{row}(B)$, и $sg(\cdot)$ — оператор остановки градиента. Тогда значение $\overline{P}$ совпадает со значением P, однако $\overline{P}$ дифференцируема по W, и при подсчёте градиента методом обратного распространения ошибки на этапе прохода назад градиент пройдёт через вершину B засчёт второго слагаемого. Отметим также, что оператор $sg(\cdot)$ реализован в большинстве программных средств для автоматического дифференцирования.

Приведём полное описание полученного алгоритма 1, основанного на методе Синхорна.

Выходом данного алгоритма будет набор параметров θ^* и $W^* = (W_1, \dots, W_n)$, использую которые легко получить отображение ω^*. Важно отметить, что у метода Синхорна не существует гарантий сходимости, поэтому полученные значения θ^*, ω^* строго говоря не обязаны быть мак-

---

Algorithm 1: Метод Синхорна в задаче выбора оптимального порядка

Require: $M = N_1 \times \cdots \times N_n$, $S_1, \ldots, S_m$, $y_1, \ldots, y_n$, q, τ, max_iter

1: Инициализация $\theta^1 \in \Theta, W^1 = (W_1^1, \ldots, W_n^1), W_i^1 \in \mathbb{R}^{k_i \times k_i}$, $i \in \{1, \ldots, n\}$
2: $\prec_0$ — естественный порядок
3: **for** $t = 1, \ldots, max\_iter$ **do**
4:     **for** $i = 1, \ldots, n$ **do**
5:         $B_i = S_\tau^q(W_i^t)$
6:         $P_i \sim Discrete_{row}(B_i)$
7:         $\overline{P}_i = sg(P_i - B_i) + B_i$
8:     $\omega = \omega_{\overline{P}_1} \times \cdots \times \omega_{\overline{P}_n}$
9:     $\mathcal{L} = \frac{1}{m} \sum_{i=1}^m L(A_{\theta^t}(\omega(S_i)), y_i)$
10:     Вычислить $\frac{\partial \mathcal{L}}{\partial \theta}\Big|_{\theta^t}, \frac{\partial \mathcal{L}}{\partial W}\Big|_{W^t}$
11:     Получить θ^{t+1}, W^{t+1}, используя θ^1, W^1, $\frac{\partial \mathcal{L}}{\partial \theta}\Big|_{\theta^1}, \frac{\partial \mathcal{L}}{\partial W}\Big|_{W^1}, \ldots$

симальными. Однако результаты проведённых экспериментов показывают, что получаемые псевдорешения θ^*, ω^* оказываются лучше имеющихся альтернатив.

Важным ограничением на использование метода Синхорна в задаче выбора оптимального порядка является требование дифференцируемости A_θ. Однако, ни алгоритмы классификации основанные на решающих деревьях (лучшие в мире), ни алгоритмы основанные на упорядоченных элементарных классификаторах не являются дифференцируемыми. Это сильно ограничивает применимость метода. Для решения этой проблемы предлагается, решая задачу 3.2 для недифференцируемого алгоритма A_θ, брать $\Omega(M) = \{\omega^*\}$, где ω^* — решение задачи 3.3 для некоторого дифференцируемого алгоритма классификации A'_θ. В качестве A'_θ рекомендуется брать линейный классификатор для предотвращения переобучения.

Для повышения качества порядка, выбираемого методом Синхорна, использовалась следующая идея. Для хорошей классификации прецедентов из класса K в случае линейно упорядоченных данных, выгодно, чтобы признаковые значения характерные для класса K следовали за значениями, не характерными для класса K. Аналогичная идея лежит в основе быстрой процедуры линейного упорядочения.

Для достижения указанного, предлагается использовать следующие две улучшения. Во-первых, брать в качестве $\preceq_0$ порядок, полученный быстрой процедурой выбора линейного порядка. Во-вторых, использовать регуляризованный функционал эмпирического риска:

$$\mathcal{L} = \frac{1}{m}\sum_{i=1}^{m}\left[y_i \cdot \log(sigmoid(\theta^T\omega(S_i))) + (1-y_i)\log(1 - sigmoid(\theta^T S_i))\right] -$$
$$- \alpha \cdot \sum_{i=1}^{n} w_i$$

где y_i равняется нулю, если $S_i \in R(K)$, и единице, если $S_i \in R(\overline{K})$.

3.5 Результаты экспериментов

Целью экспериментов являлось сравнение качества классификации при различных способах упорядочивания значений признаков. Экспериментальное сравнение качества классификации проводилось на семи наборах данных из www.kaggle.com и archive.ics.uci.edu. Для тестирования применялась методика трёхкратного скользящего контроля, а качество оценивалось мерой AUC ROC.

В таблице 4 представлены результаты классификации алгоритмом голосования по тупиковым представительным **ЭК** для следующих случаев: 1) для каждого N_j независимо выбран линейный порядок при помощи быстрой процедуры; 2) частичные порядки выбраны корректной процедурой; 3) каждое множество N_i является антицепью (классический подход). При проведении экспериментов с корректной процедурой использовалась функция 3.1 для оценки выживаемости.

Полученные результаты показывают, что время, затраченное на упорядочение значений признаков в каждой из двух предлагаемых процедур, существенно меньше времени обучения логического классификатора. Обе процедуры выбора линейных порядков улучшают качество классификации по сравнению с классическим подходом. При этом процедура независимого упорядочения показывает лучшее качество. Как и ожидалось, эта процедура тратит в несколько раз меньшее время, чем корректная процедура

Таблица 4 — Сравнение качества классификации при различном выборе частичных порядков

Название датасета	Быстрая процедура	Корректная процедура	Все признаки антицепи
Car	**0.76**	0.74	0.71
Ph	0.78	0.77	**0.79**
Heart	0.69	**0.72**	0.68
Dermatology	**0.81**	0.75	0.76
Performance	**0.65**	0.59	0.64
Phishing	0.76	**0.77**	0.75
Cancer	**0.63**	**0.63**	0.59

согласованного упорядочения. Суммарное время, затраченное на классификацию, для обеих процедур упорядочения значительно меньше времени классификации с использованием признаков с обратными порядками. Таким образом, более высокое качество классификации и меньшее время работы делает процедуру независимого выбора линейного порядка предпочтительней корректной процедуры. С другой стороны, подход с выбором корректного частичного порядка предоставляет больше информации для анализа классифицируемых данных.

Отметим, что в таблицу на занесёны результаты с процедурой выбора линейного порядка при помощи метода Синхорна. При проведении экспериментов, указанная процедура сильно уступала по качеству остальным подходам, поэтому на данный момент подход, ориентированный на использование оператора Синхорна, остаётся сугубо теоретическим.

При проведении экспериментального сравнения, было замечено, что важность признака коррелирует с тем, насколько легко его линейно упорядочить. Поэтому рассматривалась следующая модель. Важность каждого признака оценивалась через сумму информационных весов каждого значения признака. Наиболее важные признаки упорядочивались линейно, при помощи корректной процедуры выбора порядка. На множествах значений остальных признаков устанавливался порядок антицепь. Нетрудно видеть, что предлагаемый подход, который называется *комбинированным*, позволяет существенно улучшить качество классификации.

Открытым на данный момент является вопрос о выборе признаков, которые нужно линейно упорядочить для получения хорошего результата по качеству классификации. При проведении экспериментов этот вопрос решался следующим образом. Признаки упорядочивались в порядке уменьшения их весов. В установленном порядке, начиная с первого признака, последовательно увеличивалось подмножество линейно упорядоченных признаков путем добавления по одному признаку. Процесс останавливается, когда добавление нового признака приводит к снижению качества классификации.

Таблица 5 — Сравнение качества классификации

Название датасета	Быстрая процедура	Корректная процедура	Комбинированная процедура
Car	0.76	0.74	**0.77**
Ph	0.78	0.77	**0.81**
Heart	0.69	0.72	**0.74**
Dermatology	**0.81**	0.75	0.8
Performance	0.65	0.59	**0.69**
Phishing	0.76	**0.77**	**0.77**
Cancer	0.63	0.63	**0.68**

Полученные результаты показывают высокое качество классификации при использовании комбинированной процедуры выбора частичного порядка.

Глава 4. Асимптотически оптимальный алгоритм дуализации над произведением частичных порядков

В данном разделе введены основные понятия логического анализа данных с частичными порядками, и сформулированы две центральные задачи такого анализа, а именно, задача поиска максимальных независимых элементов произведения частичных порядков и задача поиска минимальных независимых элементов произведения частичных порядков. Также найдены асимптотические оценки типичного числа максимальных независимых элементов в случае произведения цепей. На основании этих оценок установлена асимптотическая оптимальность алгоритма RUNC-M+. Приведён способ эффективной реализации алгоритма RUNC-M+ с использованием битовых операций.

Основные результаты этой главы опубликованы в [16–21, 55, 56]

4.1 Задача дуализации над произведением частичных порядков в матричной формулировке

Введём обозначения: $Q_1(x, P)$, $x \in P$, — множество всех элементов в P, непосредственно следующих за x ($Q_1(x, P) = \{y \in P : x \prec y, \forall a \in P : x \prec a \Rightarrow a \not\prec y\}$); $Q_2(x, y, P)$, $x \in P$, $y \in Q_1(x, P)$, — множество всех элементов в P, не предшествующих x и предшествующих y ($Q_2(x, y, P) = \{a \in P : a \not\preceq x, a \preceq y\}$).

Введём понятие упорядоченного тупикового покрытия матрицы L_R, строками которой являются наборы из $R \subseteq P$.

Пусть H — набор столбцов матрицы L_R с номерами $j_1, \ldots, j_r$, $\sigma = (\sigma_1, \ldots, \sigma_r)$, $\sigma_i \in P_{j_i}$ при $i = 1,2, \ldots, r$. Тогда набор столбцов H называется упорядоченным тупиковым σ-покрытием, если выполнены следующие два условия: 1) для любого $i \in \{1,2, \ldots, r\}$ и любого $y \in Q_1(\sigma_i, P_{j_i})$ подматрица L_R^H матрицы L_R, образованная столбцами из H, содержит каждую из строк вида $(\beta_1, \ldots, \beta_{i-1}, \beta_i, \beta_{i+1}, \ldots, \beta_r)$, где $\beta_i \in Q_2(\sigma_i, y, P)$ и $\beta_t \leq \sigma_t$ при $t \neq$

i, $t \in \{1,2,\ldots,r\}$; 2) подматрица L_R^H не содержит строк, предшествующих σ.

Заметим, что если P_i, $i \in \{1,2,\ldots,r\}$, — конечная цепь и $x \in P_i$ не является наибольшим элементом в P_i, то множество $Q_1(x,P)$ состоит из одного элемента, обозначаемого далее через $x+1$, и следовательно, $Q_2(x,x+1,P) = \{x+1\}$. Поэтому в случае произведения конечных цепей условие 1) из определения упорядоченного тупикового σ-покрытия превращается в следующее условие: для любого $i \in \{1,2,\ldots,n\}$ подматрица L_R^H матрицы L_R, образованная столбцами из H, содержит строку $(\beta_1,\ldots,\beta_{i-1},\sigma_{i+1},\sigma_{i+1},\ldots,\sigma_r)$, где $\beta_t \preceq \sigma_t$ при $t \neq i$, $t \in \{1,2,\ldots,r\}$.

Возьмём элемент $x = (x_1,\ldots,x_n) \in P$, в котором компонента $x_{j_i} = \sigma_i, i \in \{1,2,\ldots,r\}$, не является наибольшим элементом в P_{j_i}, а каждая из остальных компонент x_j, $j \in \{1,2,\ldots,r\} \setminus \{j_1,\ldots,j_r\}$ — наибольший элемент в P_j. Положим $\sigma = (\sigma_1,\ldots,\sigma_r)$. Имеет место следующая

Теорема 4.1.1. *Элемент x является максимальным независимым от R тогда и только тогда, когда набор столбцов матрицы L_R с номерами $j_1,\ldots,j_r$ является упорядоченным тупиковым σ-покрытием матрицы L_R.*

Доказательство. Достаточность очевидна. Докажем необходимость.

Пусть элемент x является максимальным независимым от R. Тогда из условия независимости элемента x следует условие 2) из определения упорядоченного тупикового σ-покрытия матрицы L_R. Покажем справедливость условия 1) из того же определения.

Так как при $i \in \{1,2,\ldots,r\}$ компонента x_{j_i} элемента x, не является наибольшим элементом в P_{j_i}, то $Q_1(x_{j_i},P_{j_i}) \neq \varnothing$ для всех $i \in \{1,2,\ldots,r\}$. Пусть $i \in \{1,2,\ldots,r\}$, $\sigma' \in Q_1(x_{j_i},P_{j_i})$. Рассмотрим элемент $x' = (x'_1,\ldots,x'_n)$ множества P, полученный из элемента x путём замены компоненты x_{j_i}, равной σ_i, на компоненту x'_{j_i}, равную σ'_i. Из того, что x — максимальный независимый от R элемент, следует, что $x' \in R^+$. Это означает, что в R существует элемент x'', $x'' = (x''_1,\ldots,x''_n)$, такой что $x'' \preceq x'$.

Положим $\beta_t = x''_{j_t}$, $t \in \{1,2,\ldots,r\}$. Тогда из сказанного следует, что $\beta_t \preceq \sigma'_t$, $\sigma'_t = \sigma_t$, при $t \neq i$, $t \in \{1,2,\ldots,r\}$, и $\beta_i \preceq \sigma'_i$, $\sigma'_i \in Q_1(x_{j_i},P_{j_i})$. Если $\beta_i \preceq \sigma_i$, то $x'' \preceq x$, что противоречит независимости x. Значит, $x'' \preceq x$, т.е.

$x''_{j_i} \in Q_2(x_{j_i}, \sigma'_i, P)$. Таким образом, в силу произвольности выбора компоненты x_{j_i} элемента x, а также произвольности выбора элемента x, условие 1) оказывается выполненным и действительно набор столбцов матрицы L_R с номерами H является упорядоченным тупиковым σ-покрытием матрицы L_R. ∎

Замечание. Введённое в данном разделе понятие упорядоченного тупикового σ-покрытия матрицы L_R обобщает понятие неприводимого покрытия булевой матрицы. Действительно, если каждое $P_i = \{0{,}1\}$ и в P_i задан порядок $0 \prec 1$, то упорядоченное тупиковое $(0, \ldots, 0)$-покрытие матрицы L_R является неприводимым покрытием.

Пример. Рассмотрим множество $P = P_1 \times P_2 \times P_3 \times P_4 \times P_5$, где $P_i = \{0{,}1{,}2{,}3\}$, $0 \prec 1 \prec 2 \prec 3$, $i = 1{,}2{,}3{,}4{,}5$. Пусть $R \subset P$ состоит из четырёх элементов $S_1 = (1{,}0{,}1{,}3{,}2)$, $S_2 = (0{,}2{,}2{,}1{,}2)$, $S_3 = (0{,}3{,}1{,}2{,}3)$, $S_4 = (3{,}0{,}0{,}1{,}0)$. Тогда матрица L_R имеет вид:

$$L_R = \begin{pmatrix} 1 & 0 & 1 & 3 & 2 \\ 0 & 2 & 2 & 1 & 2 \\ 0 & 3 & 1 & 2 & 3 \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Набор столбцов $\{1{,}3{,}5\}$ является упорядоченным тупиковым $(0{,}1{,}2)$-покрытием матрицы L_R. Действительно, подматрица L_R^H содержит все строки $(1,1,2)$, $(0,2,2)$ и $(0,1,3)$ (назовём их покрывающими) и не содержит строк, предшествующих $(0,1,2)$.

Для наглядности выделим покрывающие строки

$$L_R = \begin{pmatrix} \mathbf{1} & 0 & \mathbf{1} & 3 & \mathbf{2} \\ \mathbf{0} & 2 & \mathbf{2} & 1 & \mathbf{2} \\ \mathbf{0} & 3 & \mathbf{1} & 2 & \mathbf{3} \\ 3 & 0 & 0 & 1 & 0 \end{pmatrix}$$

По теореме 4.1.1 элемент $S = (0,3,1,3,2) \in P$ является максимальным независимым от R. В этом нетрудно убедиться. Достаточно заметить, что $S \prec S_1$, $S \prec S_2$, $S \prec S_3$ и S несравним с S_4. А также все непосредственно следующие за S элементы — это S_1, S_2 и S_3.

4.2 Оценка числа упорядоченных тупиковых покрытий целочисленной матрицы

Пусть $P_i = \{0, 1, \dots, k-1\}$, $k \geq 2$, $i = 1, 2, \dots, n$, и элементы в P_i упорядочены в порядке возрастания. Введем обозначения: L — матрица, в которой n столбцов и элементы принадлежат множеству $\{0, 1, \dots, k-1\}$, $k \geq 2$; E_k^r, $r \leq n$, — множество всех наборов вида $(\sigma_1, \dots, \sigma_r)$, в которых $\sigma_i \in \{0, 1, \dots, k-1\}$, $k \geq 2$, при $i = 1, 2, \dots, r$. Рассмотрим $\sigma \in E_k^r$, $\sigma = (\sigma_1, \dots, \sigma_r)$, $\sigma_i < k-1$, $i = 1, 2, \dots, r$. Через $Q_i(\sigma)$, $i \in \{1, 2, \dots, r\}$, обозначим множество наборов $(\beta_1, \dots, \beta_r)$ в E_k^r, таких что $\beta_i = \sigma_i + 1$ и $\beta_j \leq \sigma_j$ при $j \in \{1, 2, \dots, r\} \setminus \{i\}$.

Пусть H — набор из r различных столбцов матрицы L. Множество различных строк подматрицы матрицы L, образованной столбцами набора H, можно рассматривать как некоторое подмножество E^H наборов из E_k^r. Согласно разделу 4.1 набор столбцов H является упорядоченным тупиковым σ-покрытием матрицы L, если выполнены два следующих условия:

1) E^H не содержит набор $(\beta_1, \dots, \beta_r) \in E_k^r$, в котором $\beta_j \leq \sigma_j$ при $j \in \{1, 2, \dots, r\}$;
2) если $i \in \{1, 2, \dots, r\}$, то E^H содержит хотя бы один набор из $Q_i(\sigma)$.

Если выполнено условие 1, то набор столбцов H называется упорядоченным σ-покрытием матрицы L. Если выполнено условие 2, то набор столбцов H называется упорядоченным σ-совместимым набором столбцов матрицы L. Упорядоченное (тупиковое) $(0, 0, \dots, 0)$-покрытие булевой матрицы называется (неприводимым) покрытием.

Квадратную подматрицу порядка r матрицы L назовем упорядоченной σ-подматрицей, если для множества ее различных строк E, рассматриваемого как некоторое подмножество наборов из E_k^r, выполнено $E \cap Q_i(\sigma) \neq \varnothing$ при $i \in \{1, 2, \dots, r\}$.

Обозначим через L_R матрицу, строками которой являются элементы множества R. Напомним, что согласно теореме 4.1.1 задача нахождения максимальных независимых от R элементов множества P (перечисления множества $I(R)$) сводится к задаче нахождения упорядоченных тупиковых покрытий матрицы L_R.

Пусть M_{mn}^k — совокупность всех матриц размера $m \times n$ с элементами из $\{0, 1, \ldots, k-1\}$, $k \geq 2$. Представляют интерес типичные значения числа тупиковых упорядоченных покрытий и длины тупикового упорядоченного покрытия для матрицы из M_{mn}^k. Выявление типичной ситуации связано с высказыванием типа «для почти всех матриц L из M_{mn}^k при $n \to \infty$ выполнено свойство P», причем свойство P может также иметь предельный характер. Например, если на матрицах из M_{mn}^k заданы две функции $F(L)$ и $G(L)$ с положительными значениями, то можно говорить, что для почти всех матриц L из M_{mn}^k выполнено $F(L) \sim G(L)$ ($F(L)$ асимптотически равно $G(L)$), если существуют две положительные бесконечно малые при $n \to \infty$ функции $\alpha(n)$ и $\beta(n)$, такие что для всех достаточно больших n имеет место

$$1 - \frac{|M|}{|M_{mn}^k|} \leq \alpha(n),$$

где M — множество матриц L из M_{mn}^k, для которых

$$1 - \beta(n) \leq \frac{F(L)}{G(L)} \leq 1 + \beta_n$$

Обозначим: ϕ_d, $d > 0$, — интервал $(\frac{1}{2}\log_d mn - \frac{1}{2}\log_d \log_d mn - \log_d \log_d \log_d n, \frac{1}{2}\log_d mn - \frac{1}{2}\log_d \log_d mn + \log_d \log_d \log_d n)$; E_{k-1}^r — множество наборов $(\sigma_1, \ldots, \sigma_r)$ в E_k^r, таких что $\sigma_i < k-1$, $i = 1, 2, \ldots, r$; $\Pi_r(\sigma) = (\sigma_1 + 1)^{r-1} \ldots (\sigma_r + 1)^{r-1}, \sigma \in E_{k-1}^r$.

Пусть $L \in M_{mn}^k$, $\sigma \in E_{k-1}^r$. Положим $B(L, \sigma)$ — множество всех упорядоченных тупиковых σ-покрытий матрицы L; $S(L, \sigma)$ — множество всех упорядоченных σ-подматриц матрицы L;

$$\Sigma_1(L) = \sum_{r=1}^{n} \sum_{\sigma \in E_{k-1}^r} |B(L, \sigma)|;$$

$$\Sigma_2(L) = \sum_{r=1}^{n} \sum_{\sigma \in E_{k-1}^r} |S(L, \sigma)|.$$

Теорема 4.2.1. *Если $m^\alpha \le n \le d^m$, $\alpha > 1$, $d = k/(k-1)$, то для почти всех матриц L из M^k_{mn} при $n \to \infty$ справедливо*

$$\Sigma_1(L) \sim \Sigma_2(L) \sim \sum_{r\in\phi_d} \sum_{\sigma\in E^r_{k-1}} \Pi_r(\sigma) C^r_n C^r_m r! k^{-r^2}$$

и длины почти всех упорядоченных тупиковых покрытий матрицы L принадлежат интервалу ϕ_d.

Доказательство теоремы 4.2.1 опирается на ряд приводимых ниже лемм 4.2.1-4.2.9.

Введем обозначения: W^n_r, $r \le n$, — множество всех наборов вида $\{j_1, \ldots, j_r\}$, где $j_q \in \{1, 2, \ldots, n\}$ при $q = 1, 2, \ldots, r$ и $j_1 < \cdots < j_r$; V^m_r, $r \le m$, — множество всех упорядоченных наборов вида $\{i_1, \ldots, i_r\}$, где $i_u \in \{1, 2, \ldots, m\}$ при $t = 1, 2, \ldots, r$ и $i_{u_1} \ne i_{u_2}$ при $u_1, u_2 = 1, 2, \ldots, r$.

Пусть $\sigma \in E^r_{k-1}, v \in V^m_r$, $v = \{i_1, \ldots, i_r\}$, $w \in W^n_r$, $M(v,w,\sigma)$ — совокупность всех матриц L в M^k_{mn}, таких что в подматрице, образованной столбцами с номерами из w, строка с номером i_t принадлежит $Q_t(\sigma)$ при $t = 1, 2, \ldots, r$.

Лемма 4.2.1. *Если $\sigma \in E^r_{k-1}$, $v \in V^m_r$, $w \in W^n_r$, то*

$$|M(v,w,\sigma)| = \Pi_r(\sigma)k^{mn-r^2}.$$

Доказательство. Действительно, строки с номерами из v можно выбрать $\Pi_r(\sigma)k^{r(n-r)}$ способами, остальные строки — $k^{n(m-r)}$ способами. ■

Пусть $M^*(v,w,\sigma)$ — совокупность всех таких матриц L в $M(v,w,\sigma)$, для которых набор столбцов с номерами из w является упорядоченным σ-покрытием и $L \neq M(v',w,\sigma)$ при $v' \neq v$.

Лемма 4.2.2. *Если $\sigma \in E^r_{k-1}$, $v \in V^m_r$, $w \in W^n_r$, то*

$$|M^*(v,w,\sigma)| \geq \Pi_r(\sigma)k^{mn-r^2}\left(1 - \frac{(r+1)(k-1)^r}{k^r}\right)^{m-r}.$$

Доказательство. Действительно, строки с номерами из v можно выбрать $\Pi_r(\sigma)k^{r(n-r)}$ способами, а остальные строки можно выбрать $(k^n - (r(k-1)^{r-1} + (k-1)^r)k^{n-r})^{m-r}$ способами. ■

Лемма 4.2.3. *Если $\sigma' \in E^r_{k-1}$, $\sigma'' \in E^l_{k-1}$, $v_1 \in V^m_r$, $v_2 \in V^m_l$, $w_1 \in W^n_r$, $w_2 \in W^n_l$ и наборы v_1 и v_2 пересекаются по a ($a > 0$) элементам, а наборы w_1 и w_2 пересекаются по b ($b > 0$) элементам, то*

$$|M(v_1,w_1,\sigma') \cup M(v_2,w_2,\sigma'')| \leq \Pi_r(\sigma')\Pi_l(\sigma'')(k-1)^b d^{ab} k^{mn-r^2-l^2}.$$

Доказательство. Оценим, сколькими способами можно построить матрицу из $M = M(v_1,w_1,\sigma') \cap M(v_2,w_2,\sigma'')$.

Сначала выберем те элементы, которые расположены на пересечении строк с номерами из v_1 и столбцов с номерами из w_1 (не более чем $\Pi_r(\sigma')$ способов). Затем выберем элементы, которые расположены на пересечении строк с номерами из v_2 и столбцов с номерами из w_2, учитывая, что ab из них расположены одновременно на пересечении строк с номерами из v_1 и столбцов с номерами из w_1 (не более чем $\Pi_l(\sigma'')(k-1)^{b(1-a)}$ способов). Произвольным способом доопределим остальные элементы матрицы ($k^{mn-r^2-l^2+ab}$ способов). Из сказанного следует требуемая оценка для M. ■

Лемма 4.2.4. *Если* $m^\alpha \le n \le d^m$, $\alpha > 1$, $d = k/(k-1)$, *то имеет место*

$$\sum_{r=1}^{n} \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2} \sim \sum_{r \in \phi_d} \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2},\ n \to \infty.$$

Доказательство. Положим

$$\begin{aligned}
&p = \frac{1}{2}\log_d mn - \frac{1}{2}\log_d \log_d mn;\\
&q = \log_d \log_d \log_d n;\\
&a_r = \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2};\\
&\Pi_r = \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma).
\end{aligned}$$

1. Пусть $r \ge p + q - 1$. Рассмотрим $(\sigma_1 + 1)^r \cdots (\sigma^{r+1} + 1)^r = (\sigma_1 + 1)^{r-1} \cdots (\sigma^r + 1)^{r-1} \cdot$
$\cdot (\sigma_1 + 1) \cdots (\sigma_r + 1)(\sigma^{r+1} + 1)^r$. Отсюда и из того, что число членов в Π_{r+1} в $k-1$ раз больше числа членов в сумме Π_r, получаем:

$$\frac{\Pi_{r+1}}{\Pi_r} \le (k-1)^{2r+1}.$$

Следовательно,

$$\frac{a^{r+1}}{a^r} = \frac{(n-r)(m-r)}{r+1} d^{-2r-1} \le \frac{mn}{p} d^{-2p-2q+1} \le_n 4d^{-2q+1}.$$

2. Пусть $r \le p - q + 1$ и пусть $r_0(k)$ — наименьшее целое r, $r \ge 2$, при котором

$$\frac{\exp(1) r (k-2)^{r-2}}{(k-1)^{r-2}} \le 1.$$

Покажем, что при $r \ge r_0(k)$ имеет место

$$\Pi_{r-1}(\sigma) \leq \exp(1)(k-1)^{(r-1)(r-2)}.$$

Нетрудно убедиться в справедливости оценки при $k \leq 3$ и любом $r \geq 2$. Пусть $k > 3$ и для меньших значений k при указанном ограничении на r доказываемая оценка справедлива. Представляя сумму $\Pi_{r-1}(\sigma)$ в виде полинома Ньютона, получаем согласно предположению индукции

$$\begin{aligned}\Pi_{r-1}(\sigma) = (1 + 2^{r-2} + \dots + (k-1)^{r-2})^{r-1} \leq \\ \leq (k-1)^{(r-1)(r-2)} \left(1 + \frac{\exp(1)(k-2)^{r-2}}{(k-1)^{r-2}}\right)^{r-1} \leq \\ \leq exp(1)(k-1)^{(r-1)(r-2)}\end{aligned}$$

Очевидно, что при любом $k \geq 2$ и любом $r \geq 1$

$$(k-1)^{r(r-1)} \leq \Pi_r(\sigma).$$

Таким образом, при $r \geq r_0(k)$:

$$\begin{aligned}\frac{a_{r-1}}{a_r} = \frac{\exp(1)(k-1)^{(r-1)(r-2)} r k^{2r-1}}{(k-1)^{r(r-1)}(n-r+1)(m-r+1)} \leq_n \\ \leq_n \frac{\exp(1)(k-1)pd^{2p-2q+1}}{(n-p)(m-p)} \leq_n \\ \leq_n \frac{2}{1-\frac{1}{\alpha}} \exp(1)(k-1)d^{-2q+1}.\end{aligned}$$

При $2 \leq r < r_0(k)$, пользуясь тем, что $\Pi_{r-1}(\sigma)/\Pi_r(\sigma) \leq 1$, имеем:

$$\frac{a_{r-1}}{a_r} = \frac{rk^{2r-1}}{(n-p)(m-p)} \leq_n \frac{c}{mn}, \ c = const, c > 0$$

Следовательно,

$$\sum_{r\in[p+q,n]} a_r = o\left(\sum_{r\in\phi_d} a_r\right);\ \sum_{r\in[1,p-q]} a_r = o\left(\sum_{r\in\phi_d} a_r\right).$$

■

Лемма 4.2.5. *Если $r, l \le c\log_d n$, $c < 1$, то имеет место*

$$\sum_{b=0}^{\min(r,l)} (k-1)^b d^{lb} C_n^r C_r^b C_{n-r}^{l-b} \le C_n^r C_n^l (1+\delta(n)),$$

где $\delta(n) \to 0$ при $n \to \infty$.

Доказательство. Обозначим:

$$\lambda_b = \frac{(k-1)^b d^{lb} C_n^r C_r^b C_{n-r}^{l-b}}{C_n^r C_{n-r}^l}.$$

Так как

$$\frac{C_r^b C_{n-r}^{l-b}}{C_{n-r}^l} \le \left(\frac{rl}{n-r-l}\right)^b$$

и по условию $r, l \le c\log_d n$, $c < 1$, то

$$\lambda_b \le \left(\frac{(k-1)\log_d^2 n}{n^{1-c}(1-2\log_d n/n}\right)^b$$

и оцениваемая сумма не превосходит $C_n^r C_{n-r}^l (1+\delta(n))$, где $\delta(n) \to 0$, $n \to \infty$. Отсюда, пользуясь неравенством $C_{n-r}^l \le C_n^l$, получаем утверждение леммы. ■

Будем считать $M_{mn}^k = \{L\}$ пространством элементарных событий, в котором каждое событие L происходит с вероятностью $1/|M_{mn}^k| = 1/k^{mn}$. Через $MX(L)$ будем обозначать математическое ожидание случайной величины $X(L)$, через $DX(L)$ — дисперсию случайной величины $X(L)$.

Лемма 4.2.6 [42]**.** *Пусть для случайных величин $X_1(L)$ и $X_2(L)$, определенных на M_{mn}^k, выполнено $X_1(L) \geq X_2(L) \geq 0$ и при $n \to \infty$ верно*

$$MX_1(L) \sim MX_2(L); \frac{DX_2(L)}{(MX_2(L))^2} \to 0.$$

Тогда для почти всех матриц L из M_{mn}^k имеет место:

$$X_2(L) \sim X_1(L) \sim MX_2(L), n \to \infty.$$

Пусть $\sigma \in E_{k-1}^r$, $v \in V_r^m$ и $w \in W_r^n$. На множестве $M_{mn}^k = \{L\}$ рассмотрим случайную величину $\eta_{(v,w)}^\sigma(L)$, равную 1, если L принадлежит $M(v,w,\sigma)$, и равную 0 иначе. Оценим вероятность события $\eta_{(v,w)}^\sigma(L) = 1$, обозначаемую далее через $P(\eta_{(v,w)}^\sigma(L) = 1)$. Очевидно, в силу леммы 4.2.1

$$P(\eta_{(v,w)}^\sigma(L) = 1) = \frac{|M_{(v,w,\sigma)}|}{|M_{mn}^k|} = \Pi_r(\sigma)k^{-r^2}. \tag{4.1}$$

Положим

$$\eta_1(L) = \sum_{r=1}^{n} \sum_{v \in V_r^m} \sum_{w \in W_r^n} \sum_{\sigma \in E_{k-1}^r} eta_{(v,w)}^\sigma(L);$$
$$\eta_2(L) = \sum_{r \in \phi_d} \sum_{v \in V_r^m} \sum_{w \in W_r^n} \sum_{\sigma \in E_{k-1}^r} eta_{(v,w)}^\sigma(L)$$

Нетрудно видеть, что $\eta_1(L) = \eta_2(L)$ и $\eta_2(L) \leq \eta_1(L)$. В силу 4.1

$$M\eta_1(L) = \sum_{r=1}^{n} \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2};$$
$$M\eta_2(L) = \sum_{r \in \phi_d} \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2};$$

Из полученных оценок для $M\eta_1(L)$, $M\eta_2(L)$ и леммы 4.2.4 сразу следует

Лемма 4.2.7. *Если $m^\alpha \leq n \leq d^m$, $\alpha > 1$, $d = k/(k-1)$, то имеет место*

$$M\eta_1(L) \sim M\eta_2(L) \sim \sum_{r \in \phi_d} \sum_{\sigma \in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2},\ n \to \infty.$$

Пусть $\sigma \in E_{k-1}^r$, $w \in W_r^n$. На множестве $M_{mn}^k = \{L\}$ рассмотрим случайную величину $\xi_w^\sigma(L)$, равную 1, если L принадлежит $B(L, \sigma)$, и равную 0 иначе. Вероятность события $\xi_w^\sigma(L) = 1$ обозначим через $P(\xi_w^\sigma(L) = 1)$.

Оценим $P(\xi_w^\sigma(L) = 1)$ сверху, пользуясь леммой 4.2.1. Нетрудно видеть, что

$$P(\xi_w^\sigma(L) = 1) \leq \sum_{v \in V_r^m} \frac{|M_{(v,w,\sigma)}|}{|M_{mn}^k|} = \Pi_r(\sigma) C_m^r r! k^{-r^2}. \tag{4.2}$$

С другой стороны, в силу леммы 4.2.2

$$P(\xi_w^\sigma(L) = 1) \geq \sum_{v \in V_r^m} \frac{|M_{(v,w,\sigma)}^*|}{|M_{mn}^k|}, \tag{4.3}$$

где $\frac{|M_{(v,w,\sigma)}^*|}{|M_{mn}^k|} = \Pi_r(\sigma) C_m^r r! k^{-r^2} \times \left(1 - \frac{r(k-1)^{r-1} + (k-1)^r}{k^r}\right)^{m-r}$.

Положим

$$\xi_1(L) = \sum_{r=1}^{n} \sum_{w\in W_r^n} \sum_{\sigma\in E_{k-1}^r} \xi_w^\sigma(L)$$

$$\xi_2(L) = \sum_{r\in\phi_d} \sum_{w\in W_r^n} \sum_{\sigma\in E_{k-1}^r} \xi_w^\sigma(L)$$

Нетрудно видеть, что $\xi_1(L) = \Sigma_1(L)$ и $\xi_2(L) \le \xi_1(L)$. Имеем

$$M\xi_1(L) = \sum_{r=1}^{n} \sum_{w\in W_r^n} \sum_{\sigma\in E_{k-1}^r} P(\xi_w^\sigma(L) = 1);$$

$$M\xi_2(L) = \sum_{r\in\phi_d} \sum_{w\in W_r^n} \sum_{\sigma\in E_{k-1}^r} P(\xi_w^\sigma(L)).$$

Следовательно, в силу 4.2

$$M\xi_2(L) \le M\xi_1(L) \le \sum_{r=1}^{n} \sum_{\sigma\in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2}. \tag{4.4}$$

Пользуясь 4.3 и тем, что $mr(k-1)^r/k^r \le \log_d^2 n/n^c$, $c > 0$, при $r \in \phi_d$, получаем

$$M\xi_1(L) \ge M\xi_2(L) \le F(n) \sum_{r\in\phi_d} \sum_{\sigma\in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2}, \tag{4.5}$$

где $F(n) \to 1$ при $r \in \phi_d$, $n \to \infty$.

Из 4.4, 4.5 и леммы 4.2.4 сразу следует

Лемма 4.2.8. *Если $m^\alpha \le n \le d^m$, $\alpha > 1$, $d = k/(k-1)$, то имеет место*

$$M\xi_1(L) \sim M\xi_2(L) \sim \sum_{r\in\phi_d} \sum_{\sigma\in E_{k-1}^r} \Pi_r(\sigma) C_n^r C_m^r r! k^{-r^2},\ n \to \infty.$$

Лемма 4.2.9. *Если $m^\alpha \le n \le d^m$, $\alpha > 1$, $d = k/(k-1)$, то имеет место*

$$\frac{D\eta_2(L)}{(M\eta_2(L))^2} \to 0; \ \frac{D\xi_2(L)}{(M\xi_2(L))^2} \to 0, \ n \to \infty.$$

Доказательство. Нетрудно видеть, что

$$M(\eta_2(L))^2 = \sum_{r,l\in\phi_d} \sum_{v_1\in V_r^m, v_2\in V_l^m} \sum_{w_1\in W_r^n, w_2\in W_l^n} \sum_{\sigma'\in E_{k-1}^r, \sigma''\in E_{k-1}^l} \frac{|M|}{k^{mn}},$$

где $M = M_{v_1,w_1,\sigma'} \cap M_{v_2,w_2,\sigma''}$. Отсюда, пользуясь леммами 4.2.3 и 4.2.5 получаем:

$$\begin{aligned} M(\eta_2(L))^2 \leq & \sum_{r,l\in\phi_d} \sum_{\sigma'\in E_{k-1}^r, \sigma''\in E_{k-1}^l} \Pi_r(\sigma')\Pi_l(\sigma'')\times \\ & \times \sum_{b=0}^{\min(r,l)} (k-1)^b d^{lb} C_n^r C_r^b C_{n-r}^{l-b} C_m^r r! C_m^l l! k^{-r^2-l^2} \leq \\ & \leq \sum_{r,l\in\phi_d} \sum_{\sigma'\in E_{k-1}^r, \sigma''\in E_{k-1}^l} \Pi_r(\sigma')\Pi_l(\sigma'') \times C_n^r C_n^l C_m^r r! C_m^l l! k^{-r^2-l^2}(1+\delta(n)), \end{aligned}$$

где $\delta(n) \to 0$ при $n \to \infty$.

С другой стороны, в силу леммы 4.2.7 при $n \to \infty$ имеем:

$$(M\eta_2(L))^2 \sim \sum_{r,l\in\phi_d} \sum_{\sigma'\in E_{k-1}^r, \sigma''\in E_{k-1}^l} \Pi_r(\sigma')\Pi_l(\sigma'') \times C_n^r C_n^l C_m^r r! C_m^l l! k^{-r^2-l^2}.$$

Отсюда получаем утверждение доказываемой леммы. ■

Утверждение теоремы 4.2.1 следует непосредственно из лемм 4.2.6-4.2.9. Справедливо

Следствие 4.2.1. *Если $m^\alpha \leq n \leq 2^m$, $\alpha > 1$, то для почти всех матриц L из M_{mn}^2 при $n \to \infty$ справедливо*

$$\Sigma_1(L) \sim \Sigma_2(L) \sim \sum_{r \in \phi_2} C_n^r C_m^r r! 2^{-r^2},$$

и длины почти всех неприводимых покрытий матрицы L принадлежат интервалу ϕ_2.

Приведенные в следствии 4.2.1 оценки типичных значений количественных характеристик множества неприводимых покрытий булевой матрицы первоначально получены в [8] с использованием понятий теории нормальных форм булевых функций. Идейно близки результаты более ранней работы [42], в которой предложен асимптотически оптимальный тестовый алгоритм распознавания и при обосновании этого алгоритма получены асимптотики типичных значений числа неприводимых покрытий и числа неприводимых покрытий типичной длины для матрицы из некоторого специального подмножества множества M_{mn}^2.

Замечание. Пусть N_{mn}^k — подмножество в M_{mn}^k, содержащее все матрицы из M_{mn}^k с попарно различными строками. В утверждении теоремы 4.2.1 можно заменить M_{mn}^k на N_{mn}^k, так как нетрудно показать, что при $m_2 = o(kn)$, $n \to \infty$, почти все матрицы из M_{mn}^k представляют собой матрицы с попарно различными строками.

Приведём результаты, полученные ранее для случая тупиковых покрытий целочисленной матрицы. Пусть $L \in M_{mn}^k$, $\sigma \in E_k^r$. Обозначим $Q_i^1(\sigma)$ — множество всех наборов вида $(\sigma_1, \ldots, sigma_{i-1}, a, \sigma_{i+1}, \ldots, \sigma_r)$, где $a \in \{0,1,\ldots,k-1\}$ и $a \neq \sigma_i$. Набор столбцов H матрицы L называется *тупиковым* σ-покрытием матрицы L, если выполнены два следующих условия:

1) E^H не содержит набора σ;

2) если $i \in \{1,2,\ldots,r\}$, то E^H содержит хотя бы один набор из $Q_i^1(\sigma)$.

Если выполнено условие 1, то набор столбцов H называется σ-покрытием матрицы L. Если выполнено условие 2, то набор столбцов H называется σ-совместимым набором столбцов матрицы L.

Квадратную подматрицу порядка r матрицы L назовем σ-подматрицей, если для множества ее различных строк E, рассматриваемого как некоторое подмножество наборов из E_k^r, выполнено $E \cap Q_i^1(\sigma) \neq \varnothing$ при $i \in \{1, 2, \ldots, r\}$.

Обозначим $\Sigma_3(L)$ — множество тупиковых σ-покрытий матрицы L для всех $r = 1, 2, \ldots, n$ и для всех $\sigma \in E_k^r$; $\Sigma_4(L)$ — множество тупиковых σ-подматриц матрицы L для всех $r = 1, 2, \ldots, n$ и для всех $\sigma \in E_k^r$. Справедлива следующая

Теорема 4.2.2 [13]**.** *Если $m^\alpha \leq n \leq k^m$, $\alpha > 1$, $d = k/(k-1)$, то для почти всех матриц L из M_{mn}^k при $n \to \infty$ справедливо*

$$\Sigma_3(L) \sim \Sigma_4(L) \sim \sum_{r \in \phi_k} C_n^r C_m^r r!(k-1)^r k^{r-r^2}$$

и длины почти всех тупиковых покрытий матрицы L принадлежат интервалу ϕ_k.

4.3 Описание алгоритма RUNC-M+

Введём следующее определение. Набор столбцов $H = (j_1, \ldots, j_r)$ называется σ-совместимым, $\sigma \in E_{k-1}^r$, если подматрица L^H матрицы L, образованная столбцами $j_1, \ldots, j_r$ содержит упорядоченную σ-подматрицу.

Алгоритм RUNC-M+ перечисляет с полиномиальной задержкой все максимальные σ-совместимые наборы столбцов H, а следовательно, согласно теореме 4.2.1, является асимптотически оптимальным.

Алгоритм описывается приведённой ниже процедурой RUNCM. Процедура запускается с параметрами $H = \varnothing$, $\sigma = \varnothing$, $D = \{1, \ldots, m\}$, $C = \{1, \ldots, n\}$, $\forall j \in \{1, \ldots, n\}$, $S[j] = \{k-2, \ldots, 0\}$.

Algorithm 2: Процедура RUNCM

1: $C^{\min} := \{j \in C \mid S[j] \neq \varnothing \ \& \ a_{ij} > 0\}$, где i — номер строки из D с минимальной суммой элементов.
2: **for** $j \in C^{\min}$ **do**
3:     $C := C \setminus \{j\}$
4:     **for** $x = a_{ij} - 1, \ldots, 0$ **do**
5:         **if** $x \in S[j]$, **then**
6:             Исключить из $S[j]$ значение x.
7:             $H := H \cup \{(j, x)\}$.
8:             Исключить из D номера строк t таких, что $a_{tj} > x$.
9:             **if** $D = \varnothing$, **then**
10:                 Найден новый максимальный независимый элемент H.
11:                 Устранить изменения, внесённые на шагах 7 и 8.
12:                 **Выход** из цикла по x.
13:             **else**
14:                 **for** $p \in C$ **do**
15:                     Исключить из $S[p]$ значения, несовместимые с набором H.
16:                 Вызвать RUNCM(L_R, H, D, C, S).
17:             Отменить изменения, внесенные на шагах 7, 8 и 15.
18:     Отменить измения, внесённые на шаге 3.

4.4 Реализация алгоритма RUNC-M+

Пусть $L \in M^k_{mn}$. Элементы матрицы L будем обозначать a_{ij}, $i \in \{1,\ldots,m\}$, $j \in \{1,\ldots,n\}$. Пусть $a \in \{0,1,\ldots,k-1\}$. Положим

$$\delta(a_{ij},a) = \begin{cases} 1, & \text{если } a_{ij} > a; \\ 0, & \text{иначе,} \end{cases}$$

для $i \in \{1,\ldots,m\}$, $j \in \{1,\ldots,n\}$.

Построим булеву матрицу L^*, состоящую из m строк и $k\times n$ столбцов, в которой строка с номером i, $i \in \{1,\ldots,m\}$, имеет вид:

$$(\delta(a_{i1},0),\ldots,\delta(a_{i1},k-1),\delta(a_{i2},0),\ldots$$
$$\ldots,\delta(a_{i2},k-1),\ldots,\delta(a_{in},0),\ldots,\delta(a_{in},k-1)).$$

Нетрудно заметить, что столбцу с номером j, $j \in \{0,1,\ldots,n\}$, исходной матрицы L соответствует группа из k столбцов матрицы L^* с номерами $k(j-1)+1$, $k(j-1)+2,\ldots,kj$. Такие столбцы назовем родственными. Через $P(L^*)$ обозначим множество всех неприводимых покрытий матрицы L^*. Несложно доказывается:

Утверждение 4.4.1. *Пусть $\sigma = (\sigma_1,\ldots,\sigma_r)$, $r \neq n$, $\sigma_i \in \{0,1,\ldots,k-2\}$, при $i = 1,2,\ldots,r$, $k > 2$. Набор из r различных столбцов $H = \{j_1,j_2,\ldots,j_r\}$ матрицы L является упорядоченным тупиковым σ-покрытием тогда и только тогда, когда выполнены два следующих условия:* (1) *набор столбцов матрицы L^* с номерами $t_1,t_2,\ldots,t_r$, где $t_i = (j_i-1)k+\sigma_i+1$, принадлежит $P(L^*)$;* (2) *если $i \in \{1,2,\ldots,r\}$ и $t_i < k_{j_i}$, то в $P(L^*)$ нет набора столбцов с номерами $t_1,\ldots,t_i-1,q_i,t_i+1,\ldots,t_r$, где $q_i \in [t_i+1,k_{j_i}]$.*

Из утверждения 4.4.1 следует, что задача построения множества $S(L)$ сводится к построению подмножества $P(L^*)$ множества $P(L)$, состоящего из всех таких неприводимых покрытий, которые, во-первых, не содержат столбцов с родственными номерами и, во-вторых, удовлетворяют некоторому дополнительному условию (2), которое назовем условием старшинства. Данный факт позволяет использовать при программной реализации алгоритма RUNC-M+ битовых операций, что приводит к многократному уменьшению времени работы и потребляемой памяти.

4.5 Реализация алгоритма RUNC-M++

При решении практических задач наилучшее качество классификации достигалось при линейном упорядочении части признаков, в то время на остальных признаках устанавливался порядок вида антицепь. Ниже приведён алгоритм, объединяющий идеи RUNC-M и RUNC-M+ для решения задачи дуализациии над смешанным произведением цепей и антицепей. Положим

$$\nabla(a_{ij}, a) = \begin{cases} 1, & \text{если } a_{ij} \neq a; \\ 0, & \text{иначе,} \end{cases}$$

для $i \in \{1, \ldots, m\}$, $j \in \{1, \ldots, n\}$.

Для удобства обозначений введём также:

$$\kappa(a_{ij}, a, i) = \begin{cases} \delta(a_{ij}, a), & \text{если множество } P_i \text{ является цепью;} \\ \nabla(a_{ij}, a), & \text{если множество } P_i \text{ является антицепью,} \end{cases}$$

Построим булеву матрицу L^*, состоящую из m строк и $k \times n$ столбцов, в которой строка с номером i, $i \in \{1, \ldots, m\}$, имеет вид:

$$(\kappa(a_{i1}, 0, i), \ldots, \kappa(a_{i1}, k-1, i), \kappa(a_{i2}, 0, i), \ldots$$
$$\ldots, \kappa(a_{i2}, k-1, i), \ldots, \kappa(a_{in}, 0, i), \ldots, \kappa(a_{in}, k-1, i)).$$

Утверждение 4.5.1. *Пусть $\sigma = (\sigma_1, \ldots, \sigma_r)$, $r \neq n$, $\sigma_i \in \{0, 1, \ldots, k-2\}$, при $i = 1, 2, \ldots, r$, $k > 2$. Набор из r различных столбцов*

$H = \{j_1, j_2, \ldots, j_r\}$ матрицы L является упорядоченным тупиковым σ-покрытием тогда и только тогда, когда выполнены два следующих условия: (1) *набор столбцов матрицы L^* с номерами $t_1, t_2, \ldots, t_r$, где $t_i = (j_i - 1)k + \sigma_i + 1$, принадлежит $P(L^*)$;* (2) *если $i \in \{1, 2, \ldots, r\}$, множество P_i является цепью и $t_i < k_{j_i}$, то в $P(L^*)$ нет набора столбцов с номерами $t_1, \ldots, t_i - 1, q_i, t_i + 1, \ldots, t_r$, где $q_i \in [t_i + 1, k_{j_i}]$.*

Из утверждения 4.5.1 следует, что задача построения множества упорядоченных тупиковых покрытий целочисленной матрицы со строками-элементами из декартова произведения цепей и антицепей сводится к построению подмножества $P(L^*)$ множества $P(L)$, состоящего из всех таких неприводимых покрытий, которые, во-первых, не содержат столбцов с родственными номерами и, во-вторых, для цепей, удовлетворяют условию старшинства. Данный факт позволяет использовать битовые операции при программной реализации алгоритма дуализации, что приводит к многократному уменьшению времени работы и потребляемой памяти.

Заключение

Основные результаты работы, выносимые на защиту:

1. Разработана единая схема синтеза алгоритмов логической классификации с использованием терминологии Correct Voting Procedures (CVP).
2. Создана общая схема синтеза логических классификаторов в случае частично упорядоченных данных, которая может быть использована для описания классических логических алгоритмов классификации и предлагаемых в рамках данной работы моделей.
3. Предложена методика повышения качества классификации без потери корректности за счёт выбора частичного порядка на множествах значений признаков.
4. Разработан асимптотически оптимальный алгоритм дуализации над произведением цепей RUNC-M+. Дано теоретическое обоснование предлагаемому алгоритму на основе матричной формулировки задачи дуализации над произведением частичных порядков.
5. Созданы методики повышения качества классификации путём синтеза стохастических композиций логических классификаторов.
6. Экспериментально подтверждено, что предлагаемые в рамках данной работы процедуры результативно применимы для решения проблемы классификации по прецедентам.

Направления дальнейших исследований

Сформулируем открытые проблемы, разрешение которых позволит улучшить существующие методы логического анализа данных применительно к задаче классификации по прецедентам:

1. Обобщить полученные в рамках диссертационной работы оценки на число упорядоченных тупиковых покрытий целочисленной матрицы на случай произведения произвольных частичных порядков с наибольшим элементом. В частности, на случай верхних полурешёток.

2. Исследовать вопрос выбора частичного порядка на множествах значений признаков среди множества (A, K)-корректных порядков.
3. Обобщение методов и понятий, характерных для LAD и FCA на случай частично упорядоченных данных.

Список литературы

[1] Андреев, А. Е. Об асимптотическом поведении числа тупиковых тестов и длины минимального теста для почти всех таблиц / А. Е. Андреев // Пробл. кибернетики. М.: Наука. — 1984. — Вып. 41. — С. 117 – 142.

[2] Бакланова, А. О. Исследование зависимости качества классификации от выбора частичных порядков на множествах значений признаков / А. О. Бакланова, Е. В. Дюкова, Г. О. Масляков // Интеллектуализация обработки информации. Тезисы докладов 13-й Международной конференции. — г. Москва. — 2020. — С. 21 – 25.

[3] Баскакова, Л. В. Модель распознающих алгоритмов с представительными наборами и системами опорных множеств / Л. В. Баскакова, Ю. И. Журавлев // Ж. вычисл. матем. и матем. физ. — 1981. — Т. 21, №5. — С. 1264—1275.

[4] Бонгард, М. М. Проблема узнавания / Бонгард М. М. // М.Ж. Физматгиз. — 1967. — 321 С.

[5] Вайнцвайг, М. Н. Алгоритм обучения распознаванию образов «Кора» / М. Н. Вайнцвайг; Под ред. В. Н. Вапник. — М.: Советское радио, 1973. — С. 110–116.

[6] Дмитриев А.И. О математических принципах классификации предметов или явлений / А. И. Дмитриев, Ю. И. Журавлёв, Ф. П. Кренделев // Дискретный анализ. Новосибирск: ИМ СО АН СССР, 1966. Вып. 7. С. 3–17.

[7] Дюкова, Е. В. Об одном алгоритме построения тупиковых тестов / Е. В. Дюкова // Сборник работ по математической кибернетике, М.:ВЦ АН СССР. — 1976. — Вып. 1. — С. 167–185.

[8] Дюкова, Е. В. Об асимптотически оптимальном алгоритме построения тупиковых тестов / Е. В. Дюкова // Докл. АН СССР. — 1977. — Т. 233., №4. — С. 527–530.

[9] Дюкова, Е. В. Асимптотически оптимальные тестовые алгоритмы в задачах распознавания / Е. В. Дюкова // Пробл. кибернетики. М.: Наука. — 1982. — Вып. 39. — С. 165–199.

[10] Дюкова, Е. В. О сложности реализации некоторых процедур распознавания / Е. В. Дюкова // Ж. вычисл. матем. и матем. физ. — 1987. — Т. 27, №1. — С. 114–127.

[11] Дюкова, Е. В. Асимптотические оценки некоторых характеристик множества представительных наборов и задач об устойчивости / Е. В. Дюкова // Ж. вычисл. матем. и матем. физ. — 1995. — Т. 35, №1. — С. 122–134.

[12] Дюкова, Е. В. О сложности реализации дискретных (логических) процедур распознавания / Е. В. Дюкова // Ж. вычисл. матем. и матем. физ. — 2004. — Т. 44, №3. — С. 562–572.

[13] Дюкова, Е. В. Дискретный анализ признаковых описаний в задачах распознавания большой размерности / Е. В. Дюкова, Ю. И. Журавлёв // Ж. вычисл. матем. и матем. физ. — 2000. — Т. 40, №8. — С. 1264–1278.

[14] Дюкова, Е. В. Методы повышения эффективности логических корректоров / Е. В. Дюкова, Ю. И. Журавлёв, П. А. Прокофьев // Машинное обучение и анализ данных — 2015. — Т. 1, №11. — С. 1555–1583.

[15] Дюкова, Е. В. Об алгебраическом синтезе корректирующих процедур распознавания на базе элементарных алгоритмов / Е. В. Дюкова, Ю. И. Журавлёв, К. В. Рудаков // Ж. вычисл. матем. и матем. физ. — 1996. — Т. 36, №8. — С. 217–225.

[16] Дюкова, Е. В. О дуализации над произведением частичных порядков / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Машинное обучение и анализ данных. — 2017. — т. 3, №4. — 239–249

[17] Дюкова, Е. В. Дуализация над произведением цепей: асимптотичекие оценки числа решений / Е. В. Дюкова, Г. О. Масляков, П. А. Проко-

фьев // Доклады Академии наук — 2018. — Т. 483, №2. — С. 130 – 133.

[18] Дюкова, Е. В. О поиске максимальных независимых элементов частичных порядков (случай цепей) / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Прикладная математика и информатика. — Москва: МАКС Пресс. — 2018. — Т. 58. — С. 12–20.

[19] Дюкова, Е. В. Задача монотонной дуализации и её обобщение — дуализация над произведением цепей / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Труды 10-й Международной конференции «Дискретные модели в теории управляющих систем». — Москва и Подмосковье. — 23–25 мая 2018 г. — Труды / Отв. ред. В. Б. Алексеев, Д. С. Романов, Б. Р. Данилов. — Москва: МАКС Пресс. — 2018. — С. 117–119.

[20] Дюкова, Е. В. Задача дуализации над произведением цепей: асимптотика типичного числа решений / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Тезисы докладов 12-й Международной конференции «Интеллектуализация обработки информации (ИОИ-2018)» (Москва, Россия – Гаэта, Италия). — М.: ТОРУС ПРЕСС. — 2018. — С. 12–159.

[21] Дюкова, Е. В. О числе максимальных независимых элементов частичных порядков (случай цепей) / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Информатика и её применения. — 2019. — Т. 13., №1. — С. 25–32.

[22] Дюкова, Е. В. О логическом анализе данных с частичными порядками в задаче классификации по прецедентам / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Ж. вычисл. матем. и матем. физ. — 2019. — Т. 59., №9. — С. 1605–1616.

[23] Дюкова, Е. В. Классификация над произведением частичных порядков / Е. В. Дюкова, Г. О. Масляков, П. А. Прокофьев // Математические методы распознавания образов: 19-я Всероссийская конференция с международным участием, ММРО-2019. — Москва. — 26—29 декабря 2019 г. — Тезисы докладов. —- С. 29–31.

[24] Дюкова, Е. В. О выборе частичных порядков на множествах значений признаков в задаче классификации / Е. В. Дюкова, Г. О. Масляков // Информатика и её применения. — 2021. — Т. 15., №4. — С. 74–80.

[25] Дюкова, Е. В. О корректной классификации над произведением частичных порядков / Е. В. Дюкова, Г. О. Масляков // Сборник трудов по материалам VII Международной конференции и молодёжной школы (ИТНТ-2021). — Самара. — 20–24 сентября 2021 г. — Т. 3. — С. 34292.

[26] Дюкова, Е. В. Корректная классификация над произведением частичных порядков / Е. В. Дюкова, Г. О. Масляков // Математические методы распознавания образов: 20-я Всероссийская конференция с международным участием, ММРО-2021. — Москва. — 7–10 декабря 2021 г. — Тезисы докладов. —- С. 59–63.

[27] Дюкова, Е. В. Поиск информативных фрагментов описаний объектов в дискретных процедурах распознавания / Е. В. Дюкова, Н. В. Песков // Ж. вычисл. матем. и матем. физ. — 2002. Т. 42., №5. — С. 741–753.

[28] Дюкова, Е. В. Построение распознающих процедур на базе элементарных классификаторов / Дюкова Е. В., Песков Н. В. // Математические вопросы кибернетики. — 2005. — №14. — С. 57–92.

[29] Дюкова, Е. В. Об асимптотически оптимальном перечислении неприводимых покрытий булевой матрицы / Дюкова Е. В., Прокофьев П. А. // Прикладная дискретная математика. — 2014. — Т. 23, №1. — С. 96–105.

[30] Дюкова, Е. В. Об оптимальном корректном перекодировании целочисленных данных в распознавании / Е. В. Дюкова, А. В. Сизов, Р. М. Сотнезов // Информатика и её применения. — 2012. — Т. 6, №4. — С. 61–65.

[31] Дюкова Е. В. Асимптотические оценки числа решений задачи дуализации и ее обобщений / Е. В. Дюкова, Р. М. Сотнезов // Ж. вычисл. матем. и матем. физ. — 2011. — Т. 51, №8. — С. 1431–1440.

[32] Дюкова Е.В. О сложности дискретных задач перечисления / Е. В. Дюкова, Р. М. Сотнезов // Доклады Академии наук. — 2010. — Т. 435, №1. — С. 11–13.

[33] Дюкова Е. В. О сложности задачи дуализации / Е. В. Дюкова, Р. М. Сотнезов // Ж. вычисл. матем. и матем. физ. — 2012. — Т. 52, №10. — С. 1926–1935.

[34] Журавлёв, Ю. И. Об алгебраическом подходе к решению задач распознавания или классификации / Ю. И. Журавлёв // Пробл. кибернетики. М.: Наука. — 1978. — Вып. 33. — С. 5–68.

[35] Журавлёв, Ю. И. Распознавание. Математические методы. Программная система. Практические применения. / Ю. И. Журавлёв, В. В. Рязанов, О. В. Сенько // М.: ФАЗИС. — 2006. — С. 176.

[36] Журавлев, Ю. И. Избранные научные труды / Ю. И. Журавлёв // Москва : Магистр. — 1998. — 416 С.

[37] Кудрявцев, В. Б. Теория тестового распознавания / В. Б. Кудрявцев, А. Е. Андреев, Э. Э. Гасанов // Издательство Физматлит. — 2007. — 321 С.

[38] Кузнецов, В. Е. Об одном стохастическом алгоритме вычисления информационных характеристик таблиц по методу тестов // Дискретный анализ. — 1973. — Т. 23 — С. 8—23.

[39] Кузнецов, С. О. Интерпретация на графах и сложностные характеристики задач поиска закономерностей определенного вида / С. О. Кузнецов // Научная и техническая информация, Сер. 2. – 1989. – №1. – С. 23–28.

[40] Кузнецов С. О. Об одной модели обучения и классификации, основанной на операции сходства / С. О. Кузнецов, В. К. Финн // Обозрение Прикладной и Промышленной Математики. — 1996. — Т. 3, №1. — С. 66–90.

[41] Масич, И. С. Метод оптимальных логических решающих правил для задач распознавания и прогнозирования / И. С. Масич // Системы

управления и информационные технологии. — 2019. — Т. 75, №1. — С. 31–37.

[42] Носков, В. Н. О числе тупиковых тестов для одного класса таблиц / В. Н. Носков, В. А. Слепян // Кибернетика. — Киев. — 1972. — №1. — С. 60–65.

[43] Рязанов, В. В. Логические закономерности в задачах распознавания (параметрический подход) / В. В. Рязанов // Ж. вычисл. матем. и матем. физ. — 2007. — Т. 47., №10. — С. 1793–1808.

[44] Финн, В. К. Базы данных с неполной информацией и новый метод автоматического порождения гипотез / В. К. Финн // В кн.: Диалоговые и фактографические системы информационного обеспечения. — М., 1981. — С. 153–156.

[45] Чегис, И. А. Логические способы контроля электрических схем / И. А. Чегис, С. В. Яблонский // Логические способы контроля электрических схем // Труды математического института имени В. А. Стеклова АН СССР. — 1958. — Т. 51. — С. 270–360

[46] Babin, M. A. Dualization in lattices given by ordered sets of irreducibles / M. A. Babin, S. O. Kuznetsov // Theor. Comput. Sci. — 2017. — Vol. 658. — P. 316–326.

[47] Bengio, S. Fast Decoding in Sequence Models using Discrete Latent Variables / S. Bengio, L. Kaiser, N. Parmar, A. Roy, N. Shazeer, J. Uszkoreit // arXiv preprint arXiv:1803.03382. — 2018.

[48] Blinova, V. G. Toxicology analysis by means of the JSM-method / V. G. Blinova, D. A. Dobrynin, V. K. Finn, S. O. Kuznetsov, E. S. Pankratova // Bioinformatics. — 2003. — Vol. 19, no. 10. — P. 1201–1207.

[49] Bonates, T. O. Maximum patterns in datasets / T. O. Bonates, P. L. Hammer, A. Kogan // Discrete Appl. Math. — 2008. — Vol. 156, no. 6. — P. 846–861.

[50] Boros, E. Dual-bounded generating problems: All minimal integer solutions for a monotone system of linear inequalities / E. Boros, K. Elbassioni, V.

Gurvich, L. Khachiyan, K. Makino // SIAM Journal on Computing. — 2002. — Vol. 31, no 5. — P. 1624–1643.

[51] Boros, E. Logical analysis of numerical data / E. Boros, P. L. Hammer, T. Ibaraki, A. Kogan // Mathematical Programming. — 1997. — Vol. 79. — P. 163–190.

[52] Breiman, L Bagging predictors / L. Breiman // Machine Learning journal. — 1996. — Vol. 24, Iss. 2. — P. 123–140.

[53] Breiman, L Random Forests / L. Breiman // Machine Learning journal. — 2001. — Vol. 45, Iss. 1. — P. 4–32.

[54] Chikalov, I Logical Analysis of Data: Theory, Methodology and Applications / I. Chikalov, V. Lozin, I. Lozina, M. Moshkov, H. S. Ngyen, A. Skowron, B. Zielosko // 10.1007/978-3-642-28667-4_3. — 2013. — Vol. 41. — P. 147–192.

[55] Djukova, E. V. Dualization Problem over the Product of Chains: Asymptotic Estimates for the Number of Solutions / E. V. Djukova, G. O. Maslyakov, P. A. Prokofjev // Doklady Mathematics. — 2018. — Vol. 98, no. 3. — P. 564–567.

[56] Djukova, E. V. / Finding Maximal Independent Elements of Products of Partial Orders (the Case of Chains) / E. V. Dyukova, G. O. Maslyakov, and P. A. Prokofjev // Computational Mathematics and Modeling. — 2019. — Vol. 30, no. 1. — P. 7–12.

[57] Djukova, E. V. Logical Classification of Partially Ordered Data / E. V. Djukova, G. O. Maslyakov, P. A. Prokofyev // 17th Russian Conference, RCAI 2019, Ulyanovsk, Russia. — October 21–25, 2019. — Proceedings. Editors: Kuznetsov, Sergei O., Panov, Aleksandr I. — P. 115–126.

[58] Djukova, E. V. On the Logical Analysis of Partially Ordered Data in the Supervised Classification Problem / E. V. Djukova, G. O. Maslyakov, P. A. Prokofyev // Computational Mathematics and Mathematical Physics. — 2019. — Vol. 59., no. 9. — P. 1542–1552.

[59] Djukova, E. V. Correct classification over a product of partial orders / E. V. Djukova, G. O. Masliakov // 2021 International Conference on Information Technology and Nanotechnology (ITNT). — Samara, Russia. — 2021.

[60] Djukova, E. V. Correct classification over a product of partial orders / E. V. Djukova, G. O. Masliakov // IEEE Proceedings of the IX International Conference on Information Technology and Nanotechnology (ITNT-2023). — Samara, Russia. — 2023. — P. 1–5.

[61] Djukova, E. V. Logical Classification of Partially Ordered Data / E. V. Djukova, G. O. Maslyakov, P. A. Prokofyev // http://arxiv.org/abs/1907.08962. — 2019.

[62] Djukova, E.V. A Classification Algorithm Based on the Complete Decision Tree / E. V. Djukova, N. V. Peskov // J. Pattern Recognition and Image Analysis. — 2007. — Vol. 17, no 3. — P. 363–367.

[63] Djukova, E. V. Asymptotically Optimal Dualization Algorithms / E. V. Djukova, P. A. Prokofyev // Computational Mathematics and Mathematical Physics. — Vol. 55, no 5. — 2015. — P. 891–905.

[64] Djukova, E. V. Logical Correctors in the Problem of Classification by Precedents / E. V. Djukova, Yu. I. Zhuravlev, P. A. Prokofyev // Comput. Math. Math. Phys. — 2017. — Vol 5, no 11. — P. 1866–1886.

[65] Djukova, E. V. Synthesis of Corrector Family with High Recognition Ability / E. V. Djukova Yu. I. Zhuravlev, R. M. Sotnezov // In Book "New Trends in Classification and Data Mining". — ITHEA, Sofia. — 2010. P. 32–39.

[66] Djukova, E. V. Construction of an Ensemble of Logical Correctors on the Basis of Elementary Classifiers / E. V. Djukova, Yu. I. Zhuravlev and R. M. Sotnezov // Pattern Recognition and Image Analysis. — 2011. — Vol. 21, №4. — P. 599–605.

[67] Dorogush, A. V. CatBoost: gradient boosting with categorical features support / A. V. Dorogush, V. Ershov, A. Gulin // arXiv preprint arXiv:1810.11363, 2018

[68] Eiter, T. Computational aspects of monotone dualization: A brief survey / T. Eiter, K. Makino, G. Gottlob // Discrete Applied Mathematics. — 2008. — Vol 156, no 11. — P. 2035–2049.

[69] Elbassioni, K. M. Algorithms for Dualization over Products of Partially Ordered Sets / K. M. Elbassioni // SIAM J. Discrete Math. — 2009. — Vol 23. no. 1. — 487–510.

[70] Fredman, L. On the complexity of dualization of monotone disjunctive normal forms / L. Fredman, L. Khachiyan // Journal of Algorithms. — 1996. — Vol. 21. — P. 618–628.

[71] Ganter, B. Formalizing hypotheses with concepts / B. Ganter, S. O. Kuznetsov // In Ganter, B., Mineau, G., eds.: Conceptual Structures: Logical, Linguistic, and Computational Issues. — Volume 1867 of Lecture Notes in Computer Science. — 2000, Springer Berlin Heidelberg. — P. 342–356

[72] Grathwohl, W. Backpropagation through the Void: Optimizing control variates for black-box gradient estimation / W. Grathwohl, D. Choi, Y. Wu, G. Roeder, D. Duvenaud // ICLR. — 2018.

[73] Gregor, K. Neural Variational Inference and Learning in Belief Networks / K. Gregor, A. Mnih // Proceedings of the 31st International Conference on Machine Learning (ICML), JMLR: WCP. — 2017. — Vol. 32. — P. 1791–1799.

[74] Guolin, K. LightGBM: A Highly Efficient Gradient Boosting Decision Tree / K. Guolin, M. Qi, F. Thomas, W. Taifeng, C. Wei, M. Weidong, Y. Qiwei, L. Tie-Yan // Advances in Neural Information Processing Systems 30. — 2017. — P. 3146–3154.

[75] Hammer, P. L. Partially defined boolean functions and cause-effect relationships / P. L. Hammer // In: Lecture at the International Conference on Multi-Attrubute Decision Making Via ORBased Expert Systems. — University of Passau, Passau, Germany. — 1986.

[76] Ignatov, D. Introduction to Formal Concept Analysis and Its Applications in Information R / D. Ignatov // arXiv preprint arXiv::1703.02819, 2017

[77] Johnson, D. S. On general all maximal independent sets / D. S. Johnson, M. Yannakakis, C. H. Papadimitriou // Information Processing Letters. — 1988. — Vol. 27., no. 3. — P. 119–123.

[78] Kourie, D.G. An incremental algorithm to construct a lattice of set intersections / D. G. Kourie, S. A. Obiedkov, B. W. Watson, D. Merwe // Sci. Comput. Program. — 2009. — Vol. 74, no 3. — P. 128–142.

[79] Krajca, P. Distributed algorithm for computing formal concepts using map-reduce framework / P. Krajca, V. Vychodil // In: N. Adams et al. (Eds.): IDA. — 2009. — Vol. 5772. — P. 333–344.

[80] Kuznetsov, S. O. Comparing performance of algorithms for generating concept lattices / S. O. Kuznetsov, S. A. Obiedkov // J. Exp. Theor. Artif. Intell. — Vol. 14, no 2. — 2002. — P. 189–216.

[81] Kuznetsov, S. O. A Fast Algorithm for Computing All Intersections of Objects in a Finite Semi-Lattice / S. O. Kuznetsov // Automatic Documentation and Mathematical Linguistics. — 1993. — Vol. 27, no 5. — P. 11–21.

[82] Kuznetsov, S. O. Mathematical aspects of concept analysis / S. O. Kuznetsov // Journal of Mathematical Science. — Vol. 80. no. 2. — 1996. — 1654–1698.

[83] Kuznetsov, S. O. Galois connections in data analysis: Contributions from the soviet era and modern russian research / Kuznetsov S. O. // In: Formal Concept Analysis, Foundations and Applications. — 2005. — P. 196–225.

[84] Murakami, K. Efficient algorithms for dualizing large-scale hypergraphs / K. Murakami, T. Uno // Discrete Applied Mathematics. — 2014. — Vol. 170.

[85] Prescott R. A. Ranking via sinkhorn propagation / R. A. Prescott, R. S. Zemel // arXiv preprint arXiv:1106.1925. — 2011.

[86] Ryazanov, V.V. Mathematical Methods for Pattern Recognition : Logical, Optimization, Algebraic Approaches / V. V. Ryazanov, O. V. Sen'ko, Yu. I. Zhuravlev // Proceedings of the 14th International Conference on Pattern Recognition. — 1998. — P. 831–834.

[87] Ryoo, H.S. Milp approach to pattern generation in logical analysis of data / H. S. Ryoo, I. Y. Jang // Discrete Appl. Math. — 2009. — Vol. 157. — P. 749–761.

[88] Schapire, R. E. The Strength of Weak Learnability / R. E. Schapire // Machine Learning. — Boston. MA: Kluwer Academic Publishers. — 1990.

[89] Sinkhorn, R. A relationship between arbitrary positive matrices and doubly stochastic matrices / R. Sinkhorn // The annals of mathematical statistics. — 1964. — vol. 35, no 2. — P. 876–879.

[90] Tarjan, R. E. Depth-first search and linear graph algorithms / R. E. Tarjan // SIAM Journal on Computing. — 1972. — Vol. 1, no. 2. — P. 146–160.

[91] Vinogradov, D. V. A Markov chain approach to random generation of formal concepts / D. V. Vinogradov // Proceedings of the Workshop Formal Concept Analysis Meets Information Retrieval (FCAIR 2013): CEUR Workshop Proceedings. — 2013. — Vol. 977. — P. 127—133.

[92] Vinogradov, D. V. VKF-method of hypotheses generation / D. V. Vinogradov // In: Ignatov, D., Khachay, M., Panchenko, A., Konstantinova, N., Yavorsky, R. (eds) Analysis of Images, Social Networks and Texts (AIST 2014). — Communications in Computer and Information Science. — 2014. — Vol 436. — P. 237–248.

[93] Wille, R. Restructuring lattice theory: An approach based on hierarchies of concepts / R. Wille // In Rival, I., ed.: Ordered Sets, NATO Advanced Study Institutes Series, Springer Netherlands — Vol. 83. — 1982. — P. 445–470.