Санкт-Петербургский государственный университет
Прикладная математика и информатика
Вычислительная стохастика и статистические модели

Куликов Даниил Владимирович

Редукция размерности категориальных данных на основе точного критерия Фишера

Бакалаврская работа

Научный руководитель:
к. ф.-м. н., доцент Н. П. Алексеева

Рецензент:
к. т. н., ст. науч. сотр. Л. А. Белякова

Санкт-Петербург
2016

Saint Petersburg State University
Department of Statistical Modelling

Kulikov Daniil Vladimirovich

Dimension reduction of categorical data based on Fisher's exact test

Bachelor's Thesis

Scientific Supervisor:
Candidate of Physico-Mathematical Sciences,
Associate Professor N. P. Alekseeva

Reviewer:
Candidate of Engineering Sciences,
Senior Researcher L. A. Belyakova

Saint Petersburg
2016

Содержание

Введение

В современном мире часто приходится сталкиваться с большим объемом данных и, в связи с этим, с возникающим вопросом об их влиянии на какой-то интересующий нас объект из этих данных. Так, например, любую историю болезни пациента можно представить в виде большого набора категориальных признаков, каждый из которых может означать симптомы болезни, применяемое лечение или последующие осложнения, тогда хочется выяснить, какое лечение при известных начальных симптомах может привести к наименее тяжелым осложнениям или даже к их полному отсутствию. Для решения этой проблемы применяется редукция размерности этих категориальных данных и выявление в них наиболее информативных признаков. В данной работе в качестве меры зависимости признаков будет рассмотрен точный критерий Фишера для данных большой размерности и его применение для решения поставленной задачи.

Цель работы. Реализация программы вычисления точного критерия Фишера и редукции размерности категориальных данных, сравнение с известными аналогами, реализация нового алгоритма перечисления точек грассманиана, вывод о важности и взаимодействии факторов на практическом примере.

Методы исследования. Для выявления зависимостей использовался точный критерий Фишера для таблиц сопряженности размером $r \times c$. Проведена редукция размерности категориальных данных, написаны соответствующие программы на языках R и Matlab и применены на конкретных примерах.

Структура работы. Работа состоит из введения, 5 глав, заключения и библиографии.

В первой главе рассматриваются основные понятия и различные виды p-значений, которые были применены при составлении соответствующей программы.

Во второй главе представляются методы реализации точного критерия Фишера для таблиц большой размерности, приводится принцип действия собственной программы, сравнивается результат её действия с известными аналогами и производится вывод о факторах, значимо влияющих на рецидивы болезни.

В третьей главе описывается алгоритм быстрого перечисления точек грассманиана и программа, составленная на базе этого алгоритма. Программа тестируется на реальных данных и, основываясь на результатах выполнения программы, делается вывод о факторах послеоперационных осложнений. Также реализуется собственный алгоритм

перечисления точек грассманиана, основанный на составлении диаграмм Юнга и их сопоставлении матрицам клеточной формы.

В четвертой главе вводятся понятия симптома, синдрома и грассманиана, выводится их взаимосвязь. Представляется изучение возможности использования способа выращивания конечных подпространств на основе рекуррентных соотношений типа Фибоначчи с помощью интегрирования дизайнов. Проверяется согласованность с флагом для такого типа параметризации грассманиана.

В пятой главе представлены и обоснованы результаты применения собственных программ, основанных на методах, исследованных в бакалаврской работе.

Глава 1

Критерий Фишера и виды p-значений

1.1. Основные понятия

Определение 1.1.1. Таблица сопряжённости — таблица,в которой представляется совместное распределение двух и более переменных, используемое для исследования связи между ними.

Определение 1.1.2. Точный критерий Фишера — тест статистической значимости, обычно используемый в анализе таблиц сопряжённости для исследования значимости взаимосвязи между переменными.

Для вычисления точного p-значения рассматриваемой $r \times c$ таблицы сопряженности нужно выполнить следующее:

1. Определим множество элементарных исходов в виде $r \times c$ таблиц, в которой таблицы имеют известную вероятность относительно нулевой гипотезы о независимости.

2. Упорядочим таблицы из множества элементарных исходов согласно мере отклонения (или статистическому критерию), определяющему степень отклонения каждой таблицы от нулевой гипотезы.

3. Суммируем вероятности таблиц из множества элементарных исходов, которые отклоняются не больше, чем данная.

Сложность его реализации состоит в том, что размер множества элементарных исходов растет экспоненциально с ростом размерности таблицы. Одним из решений этой проблемы был предложен алгоритм, описанный в статье [1].

Будем рассматривать статистические данные, которые записываются в таблицу сопряженности размером $r \times c$, где r — количество строк, а c — количество столбцов. Нас интересуют категориальные признаки, то есть признаки, которые не могут быть представлены в некотором интуитивном порядке или записаны числами. Таким образом, получим таблицу сопряженности, в которой записаны категориальные данные как

по строкам, так и по столбцам. Рассмотрим таблицу, в которой x_{ij} будет обозначать количество данных исследования, попадающих под категорию строки i и столбца j.

Таблица 1.1. Общий вид

Rows/Cols	Col_1	Col_2	$\cdots$	Col_c	RowTotal
Row_1	x_{11}	x_{12}	$\cdots$	x_{1c}	m_1
Row_2	x_{21}	x_{22}	$\cdots$	x_{2c}	m_2
.	.	.	.	.	.
Row_r	x_{r1}	x_{r2}	$\cdots$	x_{rc}	m_r
ColTotal	n_1	n_2	$\cdots$	n_c	N

Основная задача — узнать, верна ли нулевая гипотеза о независимости признаков в данной таблице сопряженности, для этого будем вычислять p-значение. Exact Tests в SPSS Statistics обеспечивают тремя видами p-значения для каждого теста. "Золотым стандартом"является точное p-значение. К сожалению, его вычисление не всегда оптимально, поскольку для большого объема данных оно будет долгим. В таком случае, используют p-значение Monte Carlo, оно является очень близким приближением к точному p-значению и задает границы, в котором оно лежит. А для больших, хорошо сбалансированных объемов данных асимптотическое p-значение не сильно отличается от точного, но для проверки этого факта придется вычислить p-значение другого типа.

Для вычисления точного p-значения рассматриваемой $r \times c$ таблицы сопряженности, нужно выполнить следующие пункты:

1. Определим множество элементарных исходов в виде $r \times c$ таблиц, в которой таблицы имеют известную вероятность относительно нулевой гипотезы о независимости.

2. Упорядочим таблицы из множества элементарных исходов согласно мере отклонения (или статистическому критерию), определяющему степень отклонения каждой таблицы от нулевой гипотезы.

3. Суммируем вероятности таблиц из множества элементарных исходов, которые отклоняются не больше, чем данная.

   Рассмотрим каждый пункт по отдельности.

1.2. Множество элементарных исходов и статистический критерий

Будем обозначать за x рассматриваемую таблицу сопряженности $r \times c$ и за x — любую сгенерированную таблицу из множества элементарных исходов, которая может быть рассмотрена. Точная вероятность наблюдения такой сгенерированной таблицы x зависит от используемой модели распределения. Когда и строки, и столбцы содержат только категориальные признаки, в руководстве [2] приводятся следующие три распределения: полное полиномиальное распределение, ординальное полиномиальное распределение и распределение Пуассона. Со всеми этими моделями вероятность получения x зависит от неизвестных параметров, принадлежащих клеткам $r \times c$ таблицы. Ключом к точному непараметрическому выводу является устранение всех неподходящих параметров из распределения x. Это достигается простым ограничением множества: берем в него только те таблицы, у которых маргинальные суммы совпадают с данной. В частности, определим множество элементарных исходов как

$$\Gamma = \left\{ x : x \; is \; r \times c; \sum_{j=1}^{c} x_{ij} = m_i; \sum_{i=1}^{r} x_{ij} = n_j \forall i, j \right\}. \tag{1.1}$$

Затем определим, что вероятность получения $x \in \Gamma$ относительно нулевой гипотезы о независимости

$$P(x) = \frac{\prod_{i=1}^{r} m_i! \prod_{j=1}^{c} n_j!}{N! \prod_{j=1}^{c} \prod_{i=1}^{r} x_{ij}!}, \tag{1.2}$$

где $N = \sum_{i=1}^{r} m_i = \sum_{j=1}^{c} n_j$.

Формула (1.2), не содержащая неизвестных параметров, выполняется при всех трех моделях распределения.

Множество элементарных исходов Γ в некотором роде является пространством наших действий по генерации таблиц. Это множество растет с огромной скоростью, поэтому часто предлагаются алгоритмы, решающие проблему его генерации, такие, как в статье [3] и статье [4]. В ординальном полиномиальном распределении суммы по строкам фиксированы, тогда как суммы по столбцам могут варьироваться в разных моделях. В полном полиномиальном распределении и распределении Пуассона могут варьироваться значения сумм как по строкам, так и по столбцам.

Для статистического анализа каждая таблица $x \in \Gamma$ упорядочивается по статистическому критерию или мере отклонения, которая количественно определяет степень от-

клонения таблицы от нулевой гипотезы независимости строк и столбцов. Статистический критерий будет обозначаться за $D(x)$. Большие абсолютные значения $D(x)$ будут опровергать нулевую гипотезу, тогда как малые наоборот будут с ней согласовываться. Функциональная форма $D(x)$ непосредственно для каждого типа критерия будет далее приведена. В течение этого раздела, функция $D(x)$ будет определяться как общий статистический критерий. Конкретные случаи статистического критерия будем обозначать их собственными уникальными символами. Например, для критерия хи-квадрат Пирсона, общий символ $D(x)$ заменим на $CH(x)$. Эту форму легко получить из следующих соображений: критерий хи-квадрат независимости сравнивает наблюдаемые значения в таблице с ожидаемыми значениями этих величин при нулевом распределении. Этот статистический критерий измеряет расхождение между наблюдаемыми и ожидаемыми значениями. Если расхождение больше, чем ожидалось, то считаем доказанным ошибочность нулевой гипотезы о независимости.

Статистический критерий хи-квадрат Пирсона имеет вид:

$$CH(x) = \sum_i \sum_j \frac{(x_{ij} - E_{ij})^2}{E_{ij}},$$

где E_{ij} — ожидаемое значение в строке i и столбце j, то есть $E_{ij} = m_i n_j / N$, а следовательно:

$$CH(x) = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(x_{ij} - m_i n_j / N)^2}{m_i n_j / N}. \tag{1.3}$$

1.3. Точные двусторонние p-значения

Точное двустороннее p-значение определяется как сумма вероятностей всех таблиц в Γ, которые, по крайней мере, так же далеки, как и наблюдаемая таблица x по отношению к D. В частности,

$$p_2 = \sum_{D(y) \geq D(x)} P(y) = pr[D(y) \geq D(x)]. \tag{1.4}$$

Для дальнейшего использования, определим критическую область множества элементарных исходов:

$$\Gamma^* = y \in \Gamma : D(y) \geq D(x). \tag{1.5}$$

Вычисление уравнения (1.4) иногда довольно трудно, так как размер множества элементарных исходов Γ растет экспоненциально. Например, множество всех таблиц 5×6

с суммами по строкам (7, 7, 12, 4, 4) и суммами по столбцам (4, 5, 6, 5, 7, 7) содержит 1,6 миллиардов таблиц. Тем не менее, таблицы этого множества элементарных исходов довольно редки и вряд ли уступают точному p-значению, основанному на теории больших выборок. SPSS Exact Tests используют сетевые алгоритмы, основанные на методах Мехта и Патель (1983, 1986a, 1986b) неявного перечисления таблиц в Γ и, таким образом, быстро идентифицирующие их в Γ^*. Это делает возможным вычисление точных p-значений для многих, кажущихся неразрешимыми, наборов данных, таких, как выше.

Несмотря на наличие сетевых алгоритмов, набор данных иногда слишком большой для возможности вычисления точного p-значения. Но он может быть слишком редким для достаточной надежности асимптотического значения. В этой ситуации Exact Tests используют вариант Монте-Карло, где выбирается только небольшая часть из таблиц в Γ и получается объективная оценка точного p-значения.

1.4. Двусторонние p-значения Monte Carlo

Двустороннее p-значение Монте-Карло является очень хорошим приближением к точному двустороннему p-значению, но его гораздо проще вычислить. Результаты метода Монте-Карло могут быть использованы вместо точных результатов, когда последние слишком сложно вычислить. Метод Монте-Карло является устойчивой, надежной процедурой, которая, в отличие от точного подхода, всегда занимает предсказуемое количество машинного времени. Хоть метод и не дает точного p-значения, но зато приводит довольно малый доверительный интервал, в границах которого, с высокой степенью значимости (обычно 99%), и содержится точное p-значение.

В методе Монте-Карло множество таблиц, мощность которого равна некоторому M, выбирается из множества Γ, где каждая таблица выбирается пропорционально её гипергеометрической вероятности (см. уравнение (1.2)). (Выборка таблиц пропорционально их вероятностям называется сырой выборкой по методу Монте-Карло.)

Для каждой выбранной таблицы $y_j \in \Gamma$ определим бинарный результат $z_j = 1$, если $y_j \in \Gamma^*$, иначе $z_j = 0$. Среднее арифметическое этих z_j берется как точечная оценка Монте-Карло точного двустороннего p-значения.

$$\hat{p_2} = \frac{1}{M} \sum_{j=1}^{M} z_j. \tag{1.6}$$

Вычислить $\alpha\%$ доверительный интервал в данном случае можно по следующей формуле:

$$CI_\alpha = \hat{p_2} \pm Z_{\alpha/2}\hat{\sigma}/(\sqrt{M}), \tag{1.7}$$

где $Z_{\alpha/2}$ — коэффициент доверия, его значение возьмем из таблицы на сайте [5].

Пример 1.4.1. 99% доверительным интервалом для точного p-значения является

$$CI_{0.99} = \hat{p_2} \pm 2.576\hat{\sigma}/(\sqrt{M}).$$

Техническая сложность возникает тогда, когда $\hat{p_2} = 1$ или $\hat{p_2} = 0$. Выборочное среднеквадратическое отклонение тогда равно нулю, а значит возникнет проблема, так как доверительный интервал должен будет быть нулевой ширины. Альтернативный способ вычисления доверительного интервала, не зависящий от $\hat{\sigma}$, основан на обращении точного биномиального критерия для проверки гипотезы, когда встречается крайний исход. Теперь рассмотрим подробнее этот метод, получивший название Clopper-Pearson, нахождения точного доверительного интервала. Точный доверительный интервал — это интервал от p_{lb} до p_{ub}, который удовлетворяет следующим условиям при $x = 1, 2, \ldots, n-1$:

$$\sum_{k=0}^{x} \binom{n}{k} p_{ub}^k (1-p_{ub})^{n-k} = \alpha/2,$$

$$\sum_{k=x}^{n} \binom{n}{k} p_{lb}^k (1-p_{lb})^{n-k} = \alpha/2,$$

где p_{lb} — нижняя граница интервала, p_{ub} — верхняя граница интервала, n — количество измерений, k — количество успехов.

Нижняя граница равна 0, когда $x = 0$, а верхняя равна 1, когда $x = n$.

Уравнения выше основаны на биномиальной функции распределения, для вычисления которой мы можем использовать бета-распределение или F распределение.

Используя формулы выше и опираясь на сайт [6] и руководство [2] получим, что если $\hat{p_2} = 0$, то $\alpha\%$ доверительным интервалом для точного p-значения является следующий промежуток:

$$CI = [0, 1 - (1 - \alpha/100)^{1/M}]. \tag{1.8}$$

Аналогичным образом, когда $\hat{p_2} = 1$, то $\alpha\%$ доверительным интервалом для точного p-значения является:

$$CI = [(1 - \alpha/100)^{1/M}, 1]. \tag{1.9}$$

1.5. Асимптотические двусторонние p-значения

Для всех критериев в этой главе статистический критерий $D(x)$ имеет асимптотическое хи-квадрат распределение. Асимптотическое двустороннее p - значение имеет вид:

$$\tilde{p_2} = Pr(\chi^2 \geq D(x)|df), \tag{1.10}$$

где χ^2 случайная величина с хи-квадрат распределением и df — соответствующие степени свободы. Для тестов на неупорядоченных $r \times c$ таблицах сопряженности количество степеней свободы вычисляется следующим образом: имеем $rc - 1$ свободных параметров относительно гипотезы о зависимости и $(r-1)+(c-1)$ свободных параметров относительно гипотезы о независимости, следовательно степеней свободы будет $rc - 1 - ((r-1)+(c-1)) = rc - r - c + 1 = (r-1) \times (c-1)$.

Глава 2

Неупорядоченные $r \times c$ таблицы сопряженности

В отличие от предыдущей главы, в этой главе будут рассматриваться только неупорядоченные таблицы сопряженности, поэтому, прежде всего, стоит определить отличие неупорядоченных таблиц сопряженности от односторонне-упорядоченных и упорядоченных таблиц. Оно состоит в том, что в таких таблицах и по строкам, и по столбцам содержатся данные, которые интуитивно не могут быть упорядочены, то есть данные, являющиеся категориальными признаками, например, пол пациента. В программе SPSS Statistics Exact Tests доступны три критерия для таких таблиц категориальных данных: критерий Хи-квадрат Пирсона, критерий отношения правдоподобия и точный критерий Фишера. Все они выполняются только для неупорядоченных таблиц сопряженности. Критерий хи-квадрат Пирсона уже был упомянут в разделе 1.2, а точный критерий Фишера будет разобран подробнее ниже, поэтому скажем пару слов о критерии отношения правдоподобия. Этот критерий является альтернативой критерию хи-квадрат Пирсона. Его статистический критерий вычисляется по следующей формуле:

$$LI(x) = 2\sum_{i=1}^{r}\sum_{j=1}^{c} x_{ij} \log \frac{x_{ij}}{m_i n_j / N}.$$

Для данной формулы разница в количестве свободных параметров следующая:

$$(rc - 1) - (r + c - 2) = rc - r - c + 1 = (r - 1)(c - 1).$$

Рассчитав полученный статистический критерий, точное p-значение будет определяться через (1.4), а именно через выражение $Pr\ (LI(y) \geq\ LI(x) \mid y\ \in\ \Gamma)$. Следует отметить, что для приведенной ниже таблицы-примера точные p-значения по критериям Пирсона (0.027) и отношения правдоподобия (0.036) разнятся практически на 0.01, что является довольно большой величиной, учитывая поставленную задачу подтверждения или опровержения нулевой гипотезы о независимости.

Теперь перейдем к рассмотрению точного критерия Фишера.

2.1. Постановка задачи

Предположим, что есть таблица сопряженности размера $r \times c$. За x_{ij} будем обозначать количество данных исследования, попадающих под категорию строки i и столбца

j. Определим маргинальные суммы по строкам и по столбцам:

$$m_i = \sum_{j=1}^{c} x_{ij}, i = 1, 2, \ldots, r,$$

$$n_j = \sum_{i=1}^{r} x_{ij}, j = 1, 2, \ldots, c.$$

Нулевая гипотеза о независимости признаков:

$$H_o : x_{ij} = m_i \times n_j, \forall(i, j).$$

Требуется её подтвердить или же опровергнуть, то есть объявить, что для какой-то пары (i, j) нулевая гипотеза выполняться не будет (без указания какой именно пары).

2.2. Таблица-пример

Для более полного понимания метода поясним его на простом примере. Предположим, что были получены данные о расположении поражений ротовой полости после обследований в трех сельских районах Индии. Эти данные покажем здесь в виде таблицы сопряженности 9×3, как показано далее. Переменные, указанные в таблице: область (указывает конкретную область поражения ротовой полости) и регион (указывает географический положение, конкретный район).

Нас интересует следующее: существенно ли отличается распределение поражений полости рта в трех географических регионах. Строки и столбцы этой таблицы — категориальные данные, а значит можно применить все три критерия, в том числе и точный критерий Фишера. Данные в таблице сопряженности настолько редки, что обычное хи-квадрат асимптотическое распределение с 16 степенями свободы не даст точного p-значения. Теперь перейдем непосредственно к точному критерию Фишера.

2.3. Точный тест Фишера

Обычно критерий Фишера ассоциируется с таблицами сопряженности размера 2×2. Его расширение до неупорядоченных $r \times c$ таблиц было впервые предложено Фрименом и Хальтоном (1951). Вот почему он также известен как критерий Freeman-Halton. Этот критерий стал альтернативой критерия хи-квадрат Пирсона и критерия отношения правдоподобия для доказательства независимости строк и столбцов в неупорядоченной таблице сопряженности большого размера. В программе SPSS Statistics

Таблица 2.1. Пример

		Географический регион		
		Kerala	Gujarat	Andhra
Область поражения	слизистая губы		1	
	слизистая щеки	8	1	8
	спайка губ		1	
	десна		1	
	твердое нёбо		1	
	мягкое нёбо		1	
	язык		1	
	диафрагма полости рта	1		1
	альвеолярный отросток	1		1

асимптотические результаты доступны только для таблиц 2×2, однако точные значения и значения Монте-Карло доступны и для больших таблиц. Для любой исследуемой таблицы сопряженности статистический критерий $D(x)$ обозначим за $FI(x)$. Он вычисляется по следующей формуле из руководства [2]:

$$FI(x) = -2\log(\gamma P(x)), \tag{2.1}$$

где

$$\gamma = (2\pi)^{(r-1)(c-1)/2} N^{-(rc-1)/2} \prod_{i=1}^{r} (m_i)^{(c-1)/2} \prod_{j=1}^{c} (n_j)^{(r-1)/2}.$$

Разберем эту формулу на частном случае таблицы сопряженности 2×2:

$$FI(x) = -2\log\left((2\pi)^{1/2} N^{-3/2} \prod_{i=1}^{r} (m_i)^{1/2} \prod_{j=1}^{c} (n_j)^{1/2} P(x)\right),$$

где

$$P(x) = \frac{\prod_{i=1}^{2} m_i! \prod_{j=1}^{2} n_j!}{N! \prod_{j=1}^{2} \prod_{i=1}^{2} x_{ij}!},$$

где $N = \sum_{i=1}^{2} m_i = \sum_{j=1}^{2} n_j$.

Следовательно получаем следующий вид:

$$FI(x) = -2\log\left((2\pi)^{1/2} N^{-3/2} \prod_{i=1}^{r} (m_i)^{1/2} \prod_{j=1}^{c} (n_j)^{1/2} P(x)\right).$$

Если заменить гипергеометрическое распределение на вероятностное распределение multinominal$(N, \pi_{1,1}, \ldots, \pi_{r,c})$, где каждая из N независимых величин $\pi_{i,j}$ — это вероятность попадания этой величины в ячейку таблицы (i, j). Получим, что вероятность наблюдения таблицы сопряженности из множества элементарных исходов — это:

$$P(X) = c \prod_{i=1}^{r} \prod_{j=1}^{c} \frac{\pi_{i,j}^{t_{i,j}}}{t_{i,j}!},$$

где

$$c = 1 / \sum_{(\tau_{i,j})} \prod_{i=1}^{r} \prod_{j=1}^{c} \frac{\pi_{i,j}^{\tau_{i,j}}}{\tau_{i,j}!},$$

где суммирование проводится по всем таблицам $\tau_{i,j}$ из множества таблиц с теми же маргинальными значениями.

Для вычисления этого значения можно работать с $-2(\log P(X) - \log c)$, чтобы избежать излишних вычислений. После чего домножить полученное на вычисленное значение c.

Вычислим теперь для таблицы-примера значение статистического критерия $FI(x)$. Сначала вычислим $P(x)$ по формуле (1.2), оно будет равно 0,0000053. Далее выясним значение $\gamma = 9,73$. Таким образом получим $FI(x) = 19,72$.

Точное p-значение определяется через (1.4), а именно $pr[FI(y) \geq 19.72 \mid y \in \Gamma]$. То есть генерируем множество таблиц, потом для каждой считаем значение статистического критерия, и, если оно больше 19.72, то добавляем его к текущему p-значению. Точное p-значение равно 0,010, а значит существует значительное взаимодействие между областью поражения и географическим положением.

Иногда набор данных слишком велик для точного анализа, и вместо него должен быть использован метод Монте-Карло. Так, например, мы получим объективную оценку точного p-значения для точного критерия Фишера на основе сырой выборки по методу Монте-Карло при выборке 10000 таблиц из множества элементарных исходов. Генерируем множество мощности 10000, для каждой таблицы из этого множества считаем значение статистического критерия, и, основываясь на нем, получаем множество Γ^*. Далее вычисляем $\hat{p_2}$, используя (1.6) и $\hat{\sigma}$, используя (**??**). Вычислив эти значения, сможем рассчитать доверительный интервал через (1.7) или, в отдельных случаях, через (1.8) и (1.9). В данном примере нижняя граница будет равна 0.007, а верхняя граница — 0.013. Пусть метод Монте-Карло производит 99% доверительный интервал для точного p-значения. Таким образом, хоть эта точечная оценка может слегка измениться, если взять выборку с другой начальной точки или другой генератор случайных чисел,

мы можем быть на 99% уверены, что точное p-значение содержится в интервале от 0,007 до 0,013. Кроме того, мы всегда можем взять большее количество таблиц, чтобы уменьшить ширину этого интервала. Очевидно, что критерий по методу Монте-Карло приводит к такому же выводу, что и при точном критерии, и показывает, что существует значительная взаимосвязь строк и столбцов в этой таблице сопряженности. В данном примере асимптотический результат не сможет продемонстрировать значимости.

2.4. Альтернативная программа

Казалось бы, что алгоритм понятен и прост, но (1.4) сложно вычислить при росте размера таблицы, так как размер множества элементарных исходов Г будет расти экспоненциально. Для облегчения вычисления можно использовать сетевой алгоритм, описанный в статье [7], или алгоритм Pagano and Halvorsen, описанный в статье [8]. Однако я буду использовать алгоритмы, описанные в статьях [9] и [10], основанные на методе Monte Carlo и использовании теста χ^2. Язык написания программы — Matlab R2013a. Для повышения вычислительной точности я составил отдельные программы для простых случаев таблиц, а именно: $2 \times 2, 2 \times 3, 2 \times 4, 3 \times 3$. Позднее действие данной программы сравним с результатами, полученными с помощью SPSS Exact Tests на примере, приведенном ниже.

Для использования программы в файле SetTable.m в единственной строке fisherRxC([...]) внутри квадратных скобок введем таблицу сопряженности по следующему правилу — вводим построчно элементы таблицы через пробел, строки разделяем точкой с запятой. Функция fisherRxC включает в себя частные случаи, имеющие отдельные функции, а именно: fisher2x2, fisher2x3, fisher2x4, fisher3x3. На выходе получим точное p-значение или p-значение Монте Карло.

2.5. Сравнение результатов

Применим мою программу на некоторых сравнениях. Далее сравним полученные в моей программе точные p-значения со значениями в программе SPSS Statistics. Для более точной проверки будем брать таблицы сопряженности разных размеров. Основываясь на данных таблицы 5.7 и убедившись в правильности (то есть абсолютно малом отклонении от значений, полученных с помощью программы SPSS) полученных значе-

ний, можно сделать вывод о том, что программа работает корректно.

Таблица 2.2. Результаты выполнения программы

Точный критерий Фишера	0.7495
Критерий Фишера с МК	0.7455
Хи-квадрат Пирсона	0.8252
Отношение правдоподобия	0.9261

Таблица 2.3. Таблица сопряженности НИС и Т-стадии

НИС/Т-стадия	1	2	3	4	Итого
низкий	8	2	2	2	14
средний	17	2	3	2	24
крупный	1	1	0	0	2
высокий	2	0	1	0	3
Итого	28	5	6	4	43

Таблица 2.4. Сравнение НИС и Т-стадии в SPSS

Точный критерий Фишера	0.749
Хи-квадрат Пирсона	0.825
Отношение правдоподобия	0.926

Программа, реализующая точный критерий Фишера, тестировалась на различных таблицах сопряженности, а также с использованием различных статистических критериев, как показано в примере 2.5.1.

Пример 2.5.1. Пуста дана таблица сопряженности, показанная в таблице 2.3. В результате выполнения собственной программы получатся значения из таблицы 2.2.

Как видно из таблицы 2.4 значение, полученное с помощью статистического критерия точного критерия Фишера, более точно, что имеет немаловажное значение.

2.6. Сравнение с fisher.test()

В языке R уже существует функция, реализующая точный тест Фишера. Это: fisher.test(x, workspace, alternative, conf.int, conf.level, simulate.p.value, B), где x — таблица сопряженности, workspace — размерность пространства в сетевом алгоритме, alternative — отвечает за тип вычисляемого p-значения, conf.int и conf.level — дают возможность вычислить доверительные интервалы с заданным уровнем значимости, а simulate.p.value и B — отвечает за вычисление методом Монте-Карло и количество вычислений соответственно.

Для таблиц размерности 2×2 в языке R можно дополнительно вычислить доверительный интервал, а также левостороннее, правостороннее и двустороннее p-значение по отдельности. В моей программе представлены все типы p-значений одновременно, а также показывается значение, полученное с помощью метода Монте-Карло. Также есть возможность вычисления p-значения с помощью различных статистических критериев.

В языке R для перечисления таблиц сопряженности $r \times c$, где $r, c \geq 2$, используются сетевые алгоритмы. Поскольку точное p-значение трудно вычислимо из-за сложности перебора всех таблиц Γ^* (1.5), в основе этого метода полагается представление этого множества Γ^* в виде сети из узлов и дуг. Введем обозначение $D = (\sum_{j=1}^{r} m_j)!/(\prod_{j=1}^{r} m_j)!$, где m_j взяты из уравнения (1.1). При таком представлении в виде сети можно заменить исходную задачу перебора таблиц на эквивалентную задачу нахождения всех путей в сети, длина которых не будет превосходить значения $D * P(X)$ (1.2), где X—наблюдаемая таблица сопряженности. Построим сеть из $c + 1$ уровня, на каждом уровне определим множество узлов $(k, m_{1k}, \ldots, m_{rk})$, притом от каждого узла будет исходить дуга ровно к одному узлу на каждом нижнем уровне. Таким образом получим набор различных путей узлами разных уровней. Остается сделать отбор подходящих путей. Для этого введем $SP(k, m_{1k}, \ldots, m_{rk})$ и $LP(k, m_{1k}, \ldots, m_{rk})$, равные самому короткому пути из узла k в узел 0 и самому длинному пути соответственно, и $PAST$ — длину пути из текущего узла в узел k. Тогда при выполнении условий $PAST * SP(k) \leq D * P(X)$ и $PAST * LP(k) > D * P(X)$ мы получим подходящие пути и как раз перечислим нужные таблицы. В моей же работе используется непосредственно последовательный перебор относительно свободных ячеек.

Замечание 1. Собственную программу, реализующую точный критерий Фишера для таблиц сопряженности размерности $r \times c$ категориальных признаков, будем применять

в дальнейшем для изучения наборов факторов, влияющих на осложнения у пациентов с болезнями, способными привести к серьезным осложнениям вплоть до летального исхода.

Глава 3

Определение информативных признаков

С помощью изученного в предыдущей главе точного критерия Фишера можно было сделать вывод о значимости взаимосвязи между признаками, поэтому попытаемся расширить его применение для определения наиболее важных со статистической точки зрения симптомов, влияющих на определенный исход болезни пациента. Для этого мы введем понятие грассманиана и изучим метод быстрого перечисления его точек.

3.1. Алгоритм быстрого перечисления точек грассманиана

Определение 3.1.1. [11] Всевозможные k-мерные подпространства данного пространства $V_m = (\mathbb{F}_q)^m$ образуют грассманиан $Gr_q(k, m)$, точкой которого является одно k-мерное подпространство.

Рассматриваемое множество подпространств не является естественным образом упорядоченным, а также включения могут иметь место только для подпространств различных размерностей, поэтому для организации перебора представляется естественным вопрос перенумерации всех подпространств фиксированной размерности, т.е. точек некоторого грассманиана. Данный алгоритм основан на предложенной в диссертации [11] его векторной параметризации и ориентирован на сокращение количества операций для построения каждой следующей точки за счет использования кода Грея и соответствующего ему отношения линейного порядка.

Определение 3.1.2. Двоичный код Грея порядка n — набор всевозможных 2^n n-битных кодов, при этом любые два соседних кода в наборе различаются только в одном разряде.

Докажем существование таких кодов Грея по индукции.

Доказательство. Рассмотрим базу индукции: при $n = 0$ кодом Грея будет являться пустая последовательность.
Рассмотри индукционный переход: допустим, что в результате генерации были получены 2 последовательности, а именно G_n и G_n^{inv}, где первая — последовательность кодов Грея порядка n, а вторая — та же последовательность, только записанная в обратном порядке. Набор следующего порядка G_{n+1} можно получить, дописав нуль слева к кодам

из G_n и единицу слева к кодам из G_n^{inv}, после чего составив из полученного один набор, т.е. $G_{n+1} = 0G_n \cup 1G_n^{inv}$. При таком построении два соседних кода будут различаться только в одном разряде, что соответствует свойству кодов Грея. □

Нам требуется максимальное сокращение количества операций для построения каждой следующей точки грассманиана.

Поставим задачу следующим образом: нужно вывести все подмножества множества из n элементов в таком порядке, что каждое следующее подмножество будет отличаться от предыдущего удалением или добавлением одного элемента. Таким образом получим перечисление в порядке минимального изменения. Любому набору элементов из этого множества можно сопоставить n-битный двоичный код и свести задачу к перебору двоичных кодов Грея.

Можно составить алгоритм рекурсивного перебора на основе полученной рекуррентной формулы $G_{n+1} = 0G_n \cup 1G_n^{inv}$, то есть будем проходить от нулевого двоичного кода порядка n, на каждой итерации получая код увеличенного на 1 порядка.

Пример 3.1.1. Запишем алгоритмически способ генерации кодов Грея на языке *C♯*:

```
const n = Console.ReadLine(); //считали порядок кода Грея
int[] gray = new int[n]; //создание массива под коды
void GrayCodeGen(int step){
        if (step == n+1) Console.WriteLine(gray); //вывод кодов
        else{
                gray[step] = 0;
                GrayCode(step + 1); //шаг с прибавлением 0 слева
                gray[step] = 1;
                InvGrayCode(step + 1); //с прибавлением 1 слева
        }
}
void InvGrayCodeGen(int step){
        if (step == n+1) Console.WriteLine(gray);
        else{
                gray[step] = 1;
                GrayCode(step + 1); //шаг с прибавлением 1 слева
```

```
                gray[step] = 0;
                InvGrayCode(step + 1); //с прибавлением 0 слева
        }
}
void Main(string[] args){
        for (int i=1; i <= n; i++) gray[i] = 0;
        //создание нулевого двоичного кода порядка n
        GrayCodeGen(1); //запуск рекурсивного алгоритма
}
```

Пример 3.1.2. Пример построенного по приведенному алгоритму двоичного кода Грея длины три:
1 итерация: берем код 0 1, отражаем зеркально — 1 0, добавляем 0 перед взятым кодом и 1 перед отраженным, получаем:

$$00\ 01$$
$$11\ 10$$

2 итерация: берем полученный на предыдущем шаге код, зеркально его отражаем и получаем последовательность:

$$000\ 001\ 011\ 010$$
$$110\ 111\ 101\ 100,$$

являющуюся кодом Грея.

Определение 3.1.3. N-арным кодом Грея порядка k называется последовательность всех N^k кодов длины k, в которой любые два соседних кода различаются ровно в одном разряде.

Рассмотрим 2 частных случая, когда N — четное и когда нечетное, и алгоритмы для генерации кодов Грея в этих случаях:
Случай 1: Пусть N — четное, тогда можно генерировать коды Грея так же, как и в бинарном случае, то есть с помощью рекурсивной процедуры, с единственным отличием в том, что числа кода будут варьироваться уже не от 0 до 1, а от 0 до $N-1$.

Пример 3.1.3. Пример построенного по приведенному алгоритму четверичного кода Грея длины два:
Берем код 0 1 2 3, отражаем зеркально — 3 2 1 0, добавляем 0 перед взятым кодом и 1 перед отраженным, повторяем эти действия вплоть до 3, получаем:

00 01 02 03

13 12 11 10

20 21 22 23

33 32 31 30

Случай 2: Пусть N — нечетное, тогда генерировать коды Грея с помощью рекурсивной процедуры не получится, так как в итоге получим разницу во всех разрядах первого и последнего элемента кода.

Пример 3.1.4. Пример построенного по рекурсивному алгоритму троичного кода Грея длины два:

00 01 02

12 11 10

20 21 22,

как видно из примера коды 22 и 00 отличаются больше, чем в 1 разряде.

Для построения кода Грея в случае нечетного N можно использовать итеративный алгоритм:
Нам требуется построить N^k кодов длины k, для этого выпишем все числа x от 0 до $N-1$ — эти числа будут соответствовать месту нашего кода в последовательности.
Далее выполним следующий алгоритм:
1)$i = 0, x_0 = x$
2)$x_{i+1} = \lfloor x/N^i \rfloor$
3)$d_i = (x_i - x_{i+1}) \pmod N$, где d_i — цифры кода, начиная с младшего разряда
4)$x_i \neq 0 \to i = i + 1$ и возвращаемся к пункту 2, иначе добавляем недостающие нули и заканчиваем алгоритм.

Пример 3.1.5. Вычислим 7-ое число последовательности в случае, когда $N = 3, k = 3$.
$x_0 = N - 1 = 6, x_1 = 2 \to d_0 = 4 \pmod 3 = 1$
$x_2 = 0, \to d_1 = 2 \pmod 3 = 2$

$x_2 = 0, k = 3 \to d_2 = 0$

Выпишем полученное число в порядке от d_{k-1} до d_0 — это код 021.

Аналогично выпишем все остальные коды в последовательности.

Пример 3.1.6. Пример построенного по приведенному итеративному алгоритму троичного кода Грея длины три:

000 001 002

012 010 011

021 022 020

120 121 122

102 100 101

111 112 110

210 211 212

222 220 221

201 202 200,

как видно из примера коды 200 и 000 отличаются ровно в одном разряде. Данный код уже не будет рефлексивным, но сохранит свою цикличность и не избыточность.

Далее введем соответствующее двоичному коду Грея отношение линейного порядка. Отметим, что введенное отношение линейного порядка обязано согласовываться с флагом.

Определение 3.1.4. [11] Отношение линейного порядка $\succ_g$ называется обобщенным порядком Грея, если $(a_1, \ldots, a_m) \succ_g (a_1^{'}, \ldots, a_m^{'})$ тогда и только тогда, когда

$$(a_1 \oplus \cdots \oplus a_m, a_2 \oplus \cdots \oplus a_m, \ldots, a_m) \succ_l (a_1^{'} \oplus \cdots \oplus a_m^{'}, a_2^{'} \oplus \cdots \oplus a_m^{'}, \ldots, a_m^{'}),$$

где $\oplus$ — суммирование по модулю q, а $\succ_l$ — лексикографический порядок:
$(a_1, \ldots, a_m) \succ_l (a_1^{'}, \ldots, a_m^{'}) \leftrightarrow \sum_{i=1}^{m} a_i q^{i-1} \geq \sum_{i=1}^{m} a_i^{'} q^{i-1}$.

Упорядочивание множества векторов пространства V_m над полем $\mathbb{F}_q$, соответствующее обобщенному порядку Грея с ограничением на старший разряд, что для $X_{\tau_i} \in V_m$ и $X_{\tau_i} = a_{i_1} X_1 + \cdots + a_{i_m} X_m$ выполняется равенство $(a_{i_1}, \ldots, a_{i_m}) = (a_{i_1}, \ldots, a_{i_s}, 1, 0, \ldots, 0)$ при $s \leq m$, реализует перестановку, минимизирующую количество используемой памяти и гарантирующую одинаковое количество операций для формирования любого нового вектора.

Пример 3.1.7. Коды, соответствующие обобщенному порядку Грея над $\mathbb{F}_2$:

$$abc = a * 2^2 + (a+b) \pmod 2 * 2^1 + (a+b+c) \pmod 2 * 2^0$$
$$000 = 0 * 2^2 + 0 * 2^1 + 0 * 2^0 = 0$$
$$001 = 0 * 2^2 + 0 * 2^1 + 1 * 2^0 = 1$$
$$011 = 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 2$$
$$010 = 0 * 2^2 + 1 * 2^1 + 1 * 2^0 = 3$$
$$110 = 1 * 2^2 + 0 * 2^1 + 0 * 2^0 = 4$$
$$111 = 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 5$$
$$101 = 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = 6$$
$$100 = 1 * 2^2 + 1 * 2^1 + 1 * 2^0 = 7$$

Пример 3.1.8. Коды, соответствующие обобщенному порядку Грея над $\mathbb{F}_3$:

$$abc = a * 3^2 + (a+b) \pmod 3 * 3^1 + (a+b+c) \pmod 3 * 3^0$$
$$000 = 0 * 3^2 + 0 * 3^1 + 0 * 3^0 = 0$$
$$001 = 0 * 3^2 + 0 * 3^1 + 1 * 3^0 = 1$$
$$002 = 0 * 3^2 + 0 * 3^1 + 2 * 3^0 = 2$$
$$012 = 0 * 3^2 + 1 * 3^1 + 0 * 3^0 = 3$$
$$010 = 0 * 3^2 + 1 * 3^1 + 1 * 3^0 = 4$$
$$011 = 0 * 3^2 + 1 * 3^1 + 2 * 3^0 = 5$$
$$021 = 0 * 3^2 + 2 * 3^1 + 0 * 3^0 = 6$$
$$022 = 0 * 3^2 + 2 * 3^1 + 1 * 3^0 = 7$$
$$022 = 0 * 3^2 + 2 * 3^1 + 2 * 3^0 = 8$$

и т.д.

Для того, чтобы перечислить всевозможные k-мерные векторные подпространства пространства, порожденного набором линейно независимых векторов $(X_1, \ldots, X_m)$, достаточно перебрать базисы этих подпространств, а именно всевозможные наборы линейно независимых векторов $X_{\tau_1}, \ldots, X_{\tau_k}$, при этом нужно избежать повторений учитывания одинаковых подпространств.

Векторы X_{τ_i} являются линейно независимыми комбинациями векторов из известного нам набора $(X_1, \ldots, X_m)$, а коэффициенты этой линейной комбинации будут задавать i-ую строку матрицы. Таким образом получим матрицу из коэффициентов размером

$k \times m$. Для составления такой матрицы будем формировать вектора X_{τ_i}.

Алгоритм:

Цикл 1 Для набора $X^{(1)} = (X_1, \ldots, X_m)$ перебираем наборы $(a_1, \ldots, a_m)$в порядке Грея. Формируем на каждой итерации $(X_{\tau_1})_{iter_1} = (X_{\tau_1})_{iter_i - 1} + a_{1_t} X_t^{(1)}$, где a_{1_t}— отличающийся элемент, $X_t^{(1)} = X_t$. Для вектора $(X_{\tau_1})_{iter_1}$ и соответствующего набора $(a_{11}, \ldots, a_{1s_1}, 1, 0, \ldots, 0)$ определяем номер $j_1 = s_1 + 1$ последней единицы из набора.

Цикл 2 Для набора $X^{(2)} = (X_1, \ldots, X_{j_1 - 1}, X_{j_1 + 1}, \ldots, X_m)$ перебираем все оставшиеся наборы $(a_{21}, \ldots, a_{2(m-1)})$ в порядке Грея, начиная с набора $(0, \ldots, 0, 1, 0, \ldots, 0)$ с единицей на j_1 месте. После чего формируем на каждой итерации значение $(X_{\tau_2})_{iter_2} = (X_{\tau_2})_{iter_2 - 1} + a_{2_t} X_t^{(2)}$, где a_{2_t}— отличающийся элемент, $X_t^{(2)}$— вектор на месте t в $X^{(2)}$. Для вектора $(X_{\tau_2})_{iter_2}$ и соответствующего набора коэффициентов $(a_{21}, \ldots, a_{2s_2}, 1, 0, \ldots, 0)$ определяем номер $j_2 = s_2 + 1$ последней единицы из набора.

...

Цикл i Для набора $X^{(i)}$ — набора $X^{(1)}$ без векторов $X_{j_1}, \ldots, X_{j_{(i-1)}}$ перебираем наборы $(a_{i1}, \ldots, a_{i(m-(i-2))})$ в порядке Грея, начиная с набора $(0, \ldots, 1, \ldots, 0)$ с единицей на $j_{i-1} + 2 - i$ месте. Формируем на каждой итерации значение $(X_{\tau_i})_{iter_i} = (X_{\tau_i})_{iter_i - 1} + a_{i_t} X_t^{(i)}$, где a_{i_t}— отличающийся элемент, $X_t^{(i)}$— вектор на месте t в $X^{(i)}$. Для вектора $(X_{\tau_i})_{iter_i}$ и соответствующего набора $(a_{i1}, \ldots, a_{is_i}, 1, 0, \ldots, 0)$ определяем номер $j_i = s_i + 1$ последней единицы из набора.

...

Цикл k Для набора $X^{(k)}$ — набора $X^{(1)}$ без векторов $X_{j_1}, \ldots, X_{j_{(k-1)}}$ перебираем наборы коэффициентов $(a_{k1}, \ldots, a_{k(m-(k-2))})$ в порядке Грея, начиная с такого набора $(0, \ldots, 1, \ldots, 0)$, где единица стоит на $j_{k-1} + 2 - k$ месте. Формируем на каждой итерации $(X_{\tau_k})_{iter_k} = (X_{\tau_k})_{iter_k - 1} + a_{k_t} X_t^{(k)}$, где a_{k_t}— отличающийся элемент, $X_t^{(k)}$— вектор на месте t в $X^{(k)}$.

Составляем базис подпространства $V_k^{(iter)} = \langle (X_{\tau_1})_{iter_1}, \ldots, (X_{\tau_k})_{iter_k} \rangle$.

Конец цикла k

...

Конец цикла i

...

Конец цикла 2

Конец цикла 1

В результате работы алгоритма получим матрицы коэффициентов со следующими свойствами:

1) в каждой строке найдется коэффициент $a_{i,s} = 1$, где s - максимальный номер столбца, удовлетворяющий этому условию, после которого идут все нули.

2) в каждой строке найдется коэффициент $a_{j,l} = 1$, такой, что $i < j$ и $s < l$, то есть последняя единица в каждой строке будет расположена правее, чем в предыдущих.

3) под каждой последней единицей, описанной в первом пункте, в последующих строках идут все нули.

Данные матрицы гарантируют перечисление всех точек грассманиана без повторений и в порядке минимального изменения[11] и имеют общий вид:

Таблица 3.1. Общий вид матрицы

$$
\begin{array}{c}
 & X_1 & \dots & & & & \dots & & X_j & & \dots & X_m \\
X_{\tau_1} & * & \dots & * & 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
X_{\tau_i} & * & \dots & * & 0 & * & \dots & * & 1 & 0 & \dots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
X_{\tau_k} & * & \dots & * & 0 & * & \dots & * & 0 & * & \dots & 0
\end{array}
$$

Пример 3.1.9. Применим этот алгоритм для перечисления точек грассманиана $Gr_4(3,5)$:

В первом цикле начнем перебор всех кодов в порядке Грея:

$$00000 = 0*4^4 + 0*4^3 + 0*4^2 + 0*4^1 + 0*4^0 = 0$$
$$00001 = 0*4^4 + 0*4^3 + 0*4^2 + 0*4^1 + 1*4^0 = 1$$
$$00002 = 0*4^4 + 0*4^3 + 0*4^2 + 0*4^1 + 2*4^0 = 2$$
$$00003 = 0*4^4 + 0*4^3 + 0*4^2 + 0*4^1 + 3*4^0 = 3$$
$$00013 = 0*4^4 + 0*4^3 + 0*4^2 + 1*4^1 + 0*4^0 = 4$$
$$00010 = 0*4^4 + 0*4^3 + 0*4^2 + 1*4^1 + 1*4^0 = 5$$
$$00011 = 0*4^4 + 0*4^3 + 0*4^2 + 1*4^1 + 2*4^0 = 6$$
$$00012 = 0*4^4 + 0*4^3 + 0*4^2 + 1*4^1 + 3*4^0 = 7$$
$$00022 = 0*4^4 + 0*4^3 + 0*4^2 + 2*4^1 + 0*4^0 = 8$$

и т.д.

Под единицей, имеющей максимальный индекс в строке $a_{i,s}$, стоят в последующих строках нули. При этом последняя единица в строках ниже расположена правее последней единицы в предыдущей строке. Поэтому выполняем последующий перебор без учета столбца с индексом $a_{i,s}$ и начиная с варианта, где у последней единицы индекс столбца имеет большее значение, чем индекс s, что соответствует алгоритму.

Так, если в результате действия алгоритма для X_{τ_1} будет получена строка коэффициентов с индексом столбца последней единицы равным 3, а именно $(*, *, 1, 0, 0)$, где под $*$ обозначен некоторый полученный элемент из поля $\mathbb{F}_4 = [0, 1, 2, 3]$, то для последующих X_{τ_2} и X_{τ_3} индекс столбца последней единицы однозначно будет равен соответственно 4 и 5, исходя из условий вида матрицы.

При таких условиях получаемая матрица коэффициентов будет иметь вид:

$$\begin{array}{c} \\ X_{\tau_1} \\ X_{\tau_2} \\ X_{\tau_3} \end{array} \begin{array}{c} \begin{array}{ccccc} X_1 & X_2 & X_3 & X_4 & X_5 \end{array} \\ \begin{pmatrix} * & * & 1 & 0 & 0 \\ * & * & 0 & 1 & 0 \\ * & * & 0 & 0 & 1 \end{pmatrix} \end{array}$$

Далее, если в результате действия алгоритма для X_{τ_1} будет получена строка коэффициентов с индексом столбца последней единицы равным 2, а именно $(*, 1, 0, 0, 0)$, где под $*$ обозначен некоторый полученный элемент из поля $\mathbb{F}_4 = [0, 1, 2, 3]$, то для последующих X_{τ_2} и X_{τ_3} индекс столбца последней единицы уже может варьироваться, при этом удовлетворяя условиям вида матрицы. Максимальный индекс столбца с единицей для $X_{\tau_2} = [3, 4]$, и соответственно для $X_{\tau_3} = [(4, 5), 5]$.

В качестве примера рассмотрим частный случай, когда индексы столбцов с максимальной единицей равны $(2, 4, 5)$, и получаемая матрица коэффициентов будет иметь вид:

$$\begin{array}{c} \\ X_{\tau_1} \\ X_{\tau_2} \\ X_{\tau_3} \end{array} \begin{array}{c} \begin{array}{ccccc} X_1 & X_2 & X_3 & X_4 & X_5 \end{array} \\ \begin{pmatrix} * & 1 & 0 & 0 & 0 \\ * & 0 & * & 1 & 0 \\ * & 0 & * & 0 & 1 \end{pmatrix} \end{array}$$

Особое внимание стоит уделить варианту, когда в результате действия алгоритма для X_{τ_1} будет получена строка коэффициентов с индексом столбца последней единицы равным 1, а именно $(1, 0, 0, 0, 0)$, то для последующих X_{τ_2} и X_{τ_3} индекс столбца последней

единицы варьируется очень сильно, при этом удовлетворяя условиям вида матрицы, так максимальный индекс столбца с единицей для $X_{\tau_2} = [2, 3, 4]$, и соответственно для $X_{\tau_3} = [(3, 4, 5), (4, 5), 5]$.

В общем виде можно выразить максимальный индекс столбца с единицей для 2-ой и 3-ей строки следующим образом: обозначим индекс столбца последней единицы для X_{τ_1} за $i_1 = [1, \ldots, m + r - k]$, где r — индекс рассматриваемой строки, тогда индексы последней единицы для X_{τ_2} могут принимать значения $i_2 = [i_1 + 1, \ldots, m + r - k]$, а для X_{τ_3} — значения $i_3 = [i_2 + 1, \ldots, m + r - k]$. В качестве примера рассмотрим частный случай, когда индексы столбцов с максимальной единицей равны $(1, 2, 3)$, и получаемая матрица коэффициентов будет иметь вид:

$$\begin{array}{c} \\ X_{\tau_1} \\ X_{\tau_2} \\ X_{\tau_3} \end{array} \begin{array}{c} \begin{array}{ccccc} X_1 & X_2 & X_3 & X_4 & X_5 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array}$$

Таким образом, алгоритм перечисляет всевозможные 3-мерные векторные подпространства пространства $(\mathbb{F}_4)^5$.

3.2. Алгоритм перечисления точек грассманиана с использованием диаграмм Юнга

В разделе 3.1 рассматривалось перечисление точек грассманиана $Gr_q(k, m)$ с использованием k вложенных циклов и переборов кодов в порядке Грея, в этом разделе рассмотрим альтернативный способ перечисления его точек. Как уже говорилось ранее, точки грассманиана представимы в виде треугольных матриц размера $k \times m$, имеющих вид 3.1.

Определение 3.2.1. [14] Клеткой Шуберта называется подмножество $S_I \subset Gr_q(k, m)$, которое состоит из всевозможных подпространств V_I определенного комбинаторного типа I, где $I = (i_1, \ldots, i_k)$ — вектор длины k, состоящий из строго возрастающих номеров $1 \leq i_1 < i_2 < \cdots < i_k \leq m$.

Определение 3.2.2. Флаг F — строго возрастающая последовательность подпространств $V_0 \subset V_1 \subset \cdots \subset V_m$ пространства V.

Теорема 3.2.1. *Выбор флага F подпространств в $(\mathbb{F}_q)^m$ определяет клеточное разбиение грассманиана, при этом матрицы коэффициентов такого разбиения будут иметь вид 3.1.*

Доказательство. Возьмем U — некоторую k-плоскость в $Gr_q(k,m)$, определим флаг и $V_i = (e_1, \ldots, e_i) \subset (\mathbb{F}_q)^m$. Для каждой плоскости рассмотрим возрастающую последовательность подпространств:

$$0 \subset U \cap V_1 \subset U \cap V_2 \subset \cdots \subset U \cap V_{n-1} \subset U \cap V_n = U$$

Отметим, что $U \cap V_i$ при $i \leq n-k$ нулевое, а другие имеют размерность $(i+k-m)$. Возьмем некоторую последовательность целых чисел $(a_1, \ldots, a_k)$ и для неё определим множество следующего вида:

$$Q_{a_1,\ldots,a_k} = \{U : dim(U \cap V_{m-k+i-a_i}) = i\}$$

Размерность подмножества $(U \cap V_{m-k+i-a_i}) = m - a_i$, а значит $Q_{a_1,\ldots,a_k}$ не пусто только при условии того, что $(a_1, \ldots, a_k)$ — невозрастающая последовательность, при этом $\forall a_i \leq m-k$. Тогда в $U \in Q_{a_1,\ldots,a_k}$ выберем базис $(v_1, \ldots, v_k)$ таким образом, что вектор v_i при $i \in (1, \ldots, k)$, будет порождать пространство $(U \cap V_{m-k+i-a_i})$ с такой нормировкой, что $\langle v_i, e_{m-k+j-a_j} \rangle$ равен 0 при $j < i$ и 1 при $j = i$. Следует отметить, что получается итеративная процедура выбора вектора v_i, который на каждом шаге будет определен однозначно. Составив матрицу из векторов $(v_1, \ldots, v_k)$ получим матрицу вида 3.1, причём матрице такого вида будет отвечать $U \in Q_{a_1,\ldots,a_k}$.
В результате получили, что выбор флага определяет множество $Q_{a_1,\ldots,a_k}$, которое, в свою очередь, задает клеточное разбиение грассманиана $Gr_q(k,m)$ на клетки, имеющие вид 3.1. □

Следует обратить внимание, что построенное в дказательстве множество является клеткой Шуберта, а следовательно грассманиан разбивается в дизъюнктное объединение подмножеств S_I [12].
Пусть получен возрастающий набор чисел $I = (i_1, \ldots, i_k)$, где $i_x < i_y$ при $x < y$, где при подставлении в матрицу размера $k \times m$ построчно единиц на места i_j из набора I будут выполняться следующие условия, согласные построению в доказательстве:

1. Правее последней единицы в строке стоят только нули.

2. Под этими единицами в последующих строках стоят всюду нули.

3. Оставшиеся неопределенными числа являются произвольными из поля $\mathbb{F}_q$.

Таким образом, с помощью такого набора I можно задавать матрицы коэффициентов вида 3.1, являющиеся описанием конкретных точек грассманиана. Разобьем грассманиан $Gr_q(k,m)$ в дизъюнктное объединение подмножеств S_I, где S_I — клетка Шуберта. Чтобы перечислить все точки грассманиана можно было бы перечислить всевозможные наборы I, и на их основе построить матрицы коэффициентов, но, с практической точки зрения, эта задача трудновыполнима, поэтому чаще применяют диаграммы Юнга.

Определение 3.2.3. [13] Диаграмма Юнга λ — это конечный невозрастающий набор чисел, чаще представимый в виде набора клеток, выровненных по левой границе, количество которых в каждой строке соответствует числу из набора.

Диаграмму Юнга λ можно задать в виде вектора $(\lambda_1, \ldots, \lambda_k)$, при этом стоит обратить внимание на то, что наборы $I = (i_1, \ldots, i_k)$ и $\lambda = (\lambda_1, \ldots, \lambda_k)$ находятся в биективном соответствии, и $\lambda_{k+1-j} = i_j - j$ при $j = (1, \ldots, k)$. Обычно диаграммы Юнга описывают в виде таблиц из клеток, где в строке j будет находиться λ_j клеток. Стоит обратить внимание, что набор λ является невозрастающим и $0 \leq \lambda_j \leq (m-k)$.

Пример 3.2.1. Аналогично примеру 3.1.9 рассмотрим грассманиан $Gr_4(3,5), k = 3, m = 5$. Рассмотрим набор $I = (1,2,5)$, построим соответствующую ему диаграмму Юнга:

$$\lambda_1 = i_3 - 3 = 5 - 3 = 2; \lambda_2 = i_2 - 2 = 2 - 2 = 0; \lambda_3 = i_1 - 1 = 1 - 1 = 0;$$

то есть $\lambda = (2,0,0)$.

Принцип построения матрицы коэффициентов: при известной диаграммме Юнга нужно, начиная с самой последней нижней строки и двигаясь вверх к первой, сдвигать индекс последней единицы в строке с крайнего левого возможного положения, равного номеру строки, на число, стоящее в λ_j, начиная отсчёт с первого элемента из набора λ.
Так диаграмме Юнга $(2,0,0)$ будет сопоставляться матрица вида:

$$\begin{pmatrix} 1 & 0 & * & * & 0 \\ 0 & 1 & * & * & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Таким образом, существует однозначное соответствие между заданной диаграммой Юнга и видом матрицы, задающей клетку Шуберта.

Пример 3.2.2. Ниже аналогично приведены примеры некоторых соответствий диаграмм Юнга и клеток Шуберта грассманиана $Gr_4(3,5)$, где под $*$ обозначен произвольный элемент из поля $\mathbb{F}_4$:

Таблица 3.2. Примеры сопоставления диаграмм Юнга и клеток Шуберта

Диаграмма Юнга λ	Клетка Шуберта S_λ	$dim S_\lambda$
$(0,0,0)$	$\begin{pmatrix} 1 & 0 & 0 & * & * \\ 0 & 1 & 0 & * & * \\ 0 & 0 & 1 & * & * \end{pmatrix}$	$6-0=6$
$(1,0,0)$	$\begin{pmatrix} 1 & 0 & * & 0 & * \\ 0 & 1 & * & 0 & * \\ 0 & 0 & 0 & 1 & * \end{pmatrix}$	$6-1=5$
$(1,1,0)$	$\begin{pmatrix} 1 & * & 0 & 0 & * \\ 0 & 0 & 1 & 0 & * \\ 0 & 0 & 0 & 1 & * \end{pmatrix}$	$6-2=4$
$(1,1,1)$	$\begin{pmatrix} 0 & 1 & 0 & 0 & * \\ 0 & 0 & 1 & 0 & * \\ 0 & 0 & 0 & 1 & * \end{pmatrix}$	$6-3=3$
$(2,1,1)$	$\begin{pmatrix} 0 & 1 & 0 & * & 0 \\ 0 & 0 & 1 & * & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$	$6-4=2$
$(2,2,1)$	$\begin{pmatrix} 0 & 1 & * & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$	$6-5=1$
$(2,2,2)$	$\begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}$	$6-6=0$

Размерность S_λ можно получить, вычислив значение $k(m-k)-\sum_{i=1}^{k}\lambda_i$.

Для преобразования полученных клеток Шуберта к виду матрицы коэффициентов 3.1 нужно полученную матрицу для клетки Шуберта S_λ преобразовать в матрицу коэффициентов $A = C_{m\times m}S_\lambda C_{k\times k}$, где $C_{n\times n}$ — нулевая квадратная матрица размера $n \times n$ с единицами на побочной диагонали.

На основе выводов, приведенных выше, можно построить алгоритм перечисления точек грассманиана $Gr_q(k, m)$.

Алгоритм:

1. Генерируем в порядке Грея коды $(\lambda_1, \ldots, \lambda_k)$ длины k в поле F_{m-k} согласно алгоритму 3.1.1.

2. Рассматриваем код, как k-значное число, выполняем проверку с помощью попарного сравнения двух соседних разрядов. Отбираем из полученных кодов только удовлетворяющие условию невозрастания.

3. Каждому отобранному коду сопоставляем матрицу коэффициентов клетки Шуберта:

   а. Генерируем матрицу размера $k \times m$ из элементов, равных -1.

   б. В каждую строку i таблицы записываем 1 в столбец с индексом $i + \lambda_{k+1-i}$.

   в. Слева от полученных единиц в каждой строке записываем всюду нули.

   г. В строках выше над полученными единицами записать всюду нули.

4. Для полученных в предыдущем пункте матриц рассматриваем все оставшиеся неизмененными элементы, равные -1. Подсчитываем их количество t и в t вложенных циклах прогоняем значения этих элементов от 0 до $q - 1$.

5. Для всех полученных в результате действия алгоритма матриц проводим окончательное преобразование к матрицам .

Очевидно, полученный алгоритм приводит к результатам аналогичным алгоритму быстрого перечисления точек грассманиана, следовательно корректен.

В результате работы алгоритма получим требуемые матрицы коэффициентов, с помощью которых осуществляется перебор всех точек грассманиана, то есть всевозможных k-мерных подпространств пространства $(\mathbb{F}_q)^m$.

Преимущества алгоритма: Для приведенного выше алгоритма требуется всего лишь

одна реализация кодов Грея и не требуется использование k вложенных циклов с внутренней генерацией, поэтому данный алгоритм проводит более быструю реализацию перечисления, при этом перечисляемые подпространства будут генерироваться в порядке понижения размерности соответствующих клеток Шуберта, что даёт четкую порядковую структуру перечисления этих подпространств.

3.3. Применение программы

Задача редукции размерности сводится к задаче поиска некоторого случайного вектора заданной размерности, с помощью которого можно было бы описать данный нами эксперимент. То есть нужно отыскать матрицы коэффициентов M, такие, как полученные нами с помощью алгоритма перечисления точек грассманиана, чтобы разница между данным нам вектором $X = (X_1, \ldots, X_m)$ и преобразованным вектором $\hat{X} = MX$ была как можно меньшей. В качестве меры различия будем использовать точный критерий Фишера, исследованный нами в разделе 2.3. С использованием данного алгоритма была составлена программа на языке R, вычисляющая наилучший синдром k-ой размерности. Результат выполнения программы записывается в текстовый файл, в первой строке которого записывается комбинация, с помощью которой получаются симптомы, а под ней строки с соответствующим симптомам значениями данных. Применим её на реальных данных об операциях при тяжелых кишечных заболеваниях. Из всех данных выделим только наиболее интересующий нас набор данных, характеризующих послеоперационные осложнения, а именно их появление или отсутствие.

Используем программу и применим её к этим данным для поиска синдрома первого порядка, задаваемого парой симптомов X_{τ_1} и X_{τ_2}, тем самым выделив два основных фактора тяжести послеоперационных осложнений. В результате действия программы были получены два симптома. Первый из них: $X_{\tau_1} = X1 + X4 + X5$, то есть, согласно таблице 3.3, взаимодействие увеличения диаметра, утолщения и слоистости стенки. Представим его значения в зависимости от исходных данных послеоперационных осложнений в таблице 3.4.

Значение точного критерий Фишера в данном случае равно 0.02593.

Здесь нулевое значение фактора X_{τ_1} получается при отсутствии изменений в стенке кишки и слоистости при увеличении диаметра стенки или утолщении стенки, а также расширении и утолщении, не повлекшем слоистости стенки. Единичное значение фак-

Таблица 3.3. Набор послеоперационных осложнений

Признак	Название	Определение
X1	УДК	Увеличение диаметра кишки (0—нет, 1—да)
X2	ГНК	Гиперпневматоз кишки (0—нет, 1—да)
X3	УЖ	Повышенный уровень жидкости в кишке (0—нет, 1—да)
X4	УСК	Утолщение стенки кишки (0—нет, 1—да)
X5	ССК	Слоистость стенки (0—нет, 1—да)
X6	УБр	Уплотнение брыжейки (0—нет, 1—да)
X7	РГ	Реактивный гидроторакс (0—нет, 1—да)
X8	АЛ	Ателектаз легких (0—нет, 1—да)

Таблица 3.4. Значения первого фактора

X_{τ_1}	УДК	УСК	ССК	Объем
0	0	0	0	16
	0	1	1	5
	1	0	1	5
	1	1	0	5
1	0	0	1	16
	0	1	0	3
	1	0	0	5
	1	1	1	8

тор принимает при слоистости, увеличении и утолщении без других изменений и при полном изменении поведения стенки кишки. Таким образом, фактор X_{τ_1} будет означать тяжесть изменений стенки кишки (0 — при тяжелом, 1 — при облегченном).

Второй симптом $X_{\tau_2} = X4 + X7$, то есть, согласно таблице 3.3, взаимодействие утолщения стенки и реактивного гидроторакса. Представим его значения в зависимости от исходных данных послеоперационных осложнений в таблице 3.5.
Значение точного критерия Фишера в данном случае равно 0.003304.

Таблица 3.5. Значения второго фактора

X_{τ_2}	УСК	РГ	Объем
0	0	0	27
	1	1	5
1	0	1	15
	1	0	16

Нулевое значение фактора X_{τ_2} получается при отсутствии утолщения стенки и гидроторакса или при одновременном их появлении, при этом утолщение стенки может благотворно влиять при появлении гидроторакса, а отсутствие этих осложнений заметно ускорит выздоровление пациента. Единичное значение будет приниматься при раздельном появлении осложнений. Возникновение реактивного гидроторакса при истонченной стенке кишки может вызвать ряд ещё более крупных осложнений. Таким образом, фактор X_{τ_2} будет означать тяжесть возможных последующих осложнений в грудной области (0 — легкие, 1 — тяжелые).

Глава 4

О параметризации грассманиана

4.1. Связь грассманиана с симптомом и синдромом

Пусть существует случайный вектор $X = (X_1, \ldots, X_m)^{\mathrm{T}}$, компоненты которого принимают значения в поле $\mathbb{F}_q$.

Определение 4.1.1. [15] Обозначим за $\tau = (t_1, \ldots, t_k) \subseteq (1, \ldots, m)$ k-подмножество из m натуральных чисел. Зададим вектор-строку $A_\tau = (a_1, \ldots, a_m)$, где $a_i = 1$, если $i \in \tau$, иначе $a_i = 0$. Линейная комбинация вида $X_\tau = A_\tau X \pmod q$ называется симптомом ранга k.

Определение 4.1.2. [15] Пусть имеется $k+1$ симптомов $X_0, \ldots, X_k$. Совокупность $q^{k+1} - 1$ симптомов вида $\beta_0 X_0 + \cdots + \beta_k X_k \pmod q$, где $\beta_i \in \mathbb{F}_q$ не равны нулю одновременно, называется синдромом k-го порядка.

Разберем связь между грассманианом и понятиями симптома и синдрома. Рассмотрим частный случай $q = 2$. Набор из m различных линейно независимых векторов $(X_1, \ldots, X_m)$ над полем $\mathbb{F}_2$ является базисом некоторого m-мерного пространства $V_m = (\mathbb{F}_2)^m$. Любой вектор этого пространства можно разложить по базису в виде линейной комбинации $X_\tau = a_1 X_1 + \cdots + a_m X_m$, где a_i — элементы поля $\mathbb{F}_2$ и $\tau = \{a_i\}_{a_i \neq 0}$. Получаем, что векторы этого m-мерного пространства — симптомы некоторого ранга. Далее рассмотрим набор полученных линейно независимых векторов $(X_{\tau_0}, \ldots, X_{\tau_k})$ как базис, который образует $V_k = \langle X_{\tau_0}, \ldots, X_{\tau_k} \rangle$ $k+1$-мерное подпространство пространства V_m. Аналогично получаем, что V_k — синдром k-го порядка.

4.2. Параметризация на основе рекуррентных соотношений

Согласно определению синдрома 4.1.2, синдром нулевого порядка будет являться единичным симптомом. Синдром первого порядка будет состоять из трех симптомов $X_\tau, X_\nu, X_{\tau\nu}$, где $X_{\tau\nu} = X_\tau + X_\nu$. Эти симптомы можно считать за точки проективной прямой. Синдром второго порядка будет устроен как проективная геометрия P_2^2 и будет получаться из синдрома первого порядка добавлением к нему нового признака X_k и

симптомов, образованных линейной комбинацией синдрома первого порядка и симптома X_k. Аналогично по индукции получаем, что синдром $k+1$-го порядка можно задать следующим образом:

$$S_{k+1} = (S_k, X_i, S_k + X_i \pmod 2), X_i \notin S_k. \tag{4.1}$$

При этом симптом, стоящий на месте 2^i в синдроме S_k будем называть базовым.
Таким образом получили последовательность синдромов, построенных в конечном поле в результате рекуррентных соотношений типа Фибоначчи. На основании этого можно предположить возможность использования способа выращивания конечных подпространств на основе рекуррентных соотношений для параметризации грассманиана. Для проверки этого предположения введем несколько понятий.
Введем понятие согласованности линейного порядка с флагом 3.2.2.

Определение 4.2.1. Отношение линейного порядка $\prec$ на $V_m = (\mathbb{F}_q)^m$ согласовано с флагом F, если для $\forall i = (0, 1, \ldots, m-1)$ и $v \in V_i$, $\omega \in V_m \setminus V_i$ имеет место $v \prec \omega$.

Введем понятие дизайна.

Определение 4.2.2. [15] Дизайн $D(v, b, r, k, \lambda)$ — размещение v элементов по b блокам, каждый элемент встречается r раз, блоки размера k, а каждая пара встречается λ раз. При этом выполняется $vr = bk, r(k-1) = \lambda(v-1)$.
Дизайн называется симметричным, если $v = b, r = k$. Обозначается $D(v, k, \lambda)$.
Дизайн называется каноническим над полем F_q, если в блоках сумма элементов над F_q равна нулю. Обозначается $D^0(v, b, r, k, \lambda | F_q)$.

Синдром S_{n-1} можно описывать как проективную геометрию с точками-симптомами, так и как структуру симметрий n-мерной аффинной геометрии в виде канонического дизайна над F_{2^n}.
Согласно теореме Зингера [15] гиперплоскости геометрии P_q^n, $q = p^r$, взятые в качестве блоков, и точки, взятые в качестве элементов, образуют симметричный дизайн $D_n = D(k_n, k_{n-1}, k_{n-2})$, где $k_n = (q^n - 1)/(q-1)$.
Как получить дизайн через синдромы? Рассмотрим на примере синдромов первого и второго порядков. В качестве множества элементов примем синдром S_2, в качестве блоков — синдром S_1, тогда получится канонический дизайн $D(7, 3, 1 | S_2)$.
Канонические дизайны можно построить, используя метод интегрирования дизайнов [15],

основанный на многократных отражениях конечных геометрий в пространствах большей размерности. Операция интегрирования является индуктивной и начинается с базы индукции, которой является проективная прямая P_q^1, которая представима в виде вырожденного дизайна $D(q+1,1,0)$. Переход будет осуществляться на основе рекуррентного соотношения типа Фибоначчи, то есть $\phi_{j+1} = \phi_j + \alpha\phi_{j-1}$, где ϕ_j принадлежит множеству точек P_q^n как элементов дизайна D_n, которое обозначим за Ω_q^n. Начальными условиям для рекуррентной последовательности будет являться $\phi_0 \in \Omega_q^{n-1}$ и $\phi_1 \in \Omega_q^n$, а параметр рекуррентности α выбирается так, что $\phi_j \in \Omega_q^n \setminus \Omega_q^{n-1}$ при $j = 2, \ldots, n$. Последовательность ϕ_j будет импульсной, а набор $[\phi_1, \ldots, \phi_q]$ будем называть импульсным модулем $\pi(\phi_0|\phi_1)$, и соответственно для блока $B = [x_1, \ldots, x_k]$ $\pi(B|\phi_1)$ — импульсным модулем блока. Тогда для любого блока $B = [x_1, \ldots, x_k], x_i \in \Omega_q^{n-1}$ дизайна D_{n-1} существует разбиение $\Omega_q^n \setminus \Omega_q^{n-1}$ при помощи некоторого набора векторов $\{y\}$, такое, что импульсные модули блока относительно этого набора не будут пересекаться, а в сумме будут давать само множество. Тогда блоки вида $[B, \pi(B_i|y_j)]$ вместе с блоком из элементов Ω_q^{n-1} образуют соответствующий P_q^n дизайн D_{n+1}, а блоки $\pi(B_i|y_j)]$ — канонический дизайн, соответствующий евклидовой геометрии E_q^n.

Теперь остается выбрать базовые симптомы для синдрома S_n. Всевозможные варианты будут исчерпываться группой, изоморфной группе автоморфизмов P_q^n. Для дизайна $D_{n-1}, q = 2$ это будет группа $SL_n^{F_2}$ невырожденных матриц $n \times n$ над полем F_2 порядка $(q^n-1)\ldots(q^n-q^{n-1})$. Матрица этой группы A будет состоять из n ненулевых n-столбцов над F_2, соответствующих элементам поля F_q и ни один из векторов не будет являться линейной комбинацией других. Таким образом параметризацию грассманиана можно будет задать отображением из множества исходных векторов в произведение матрицы A на соответствующий векторам дизайн.

Теорема 4.2.1. *Порядок на основе рекуррентных соотношений несогласован с полным флагом на пространстве* $V_m = (\mathbb{F}_q)^m$.

Доказательство. Зададим базис пространства V_m как набор $\langle X_1, \ldots, X_m\rangle$.

Зададим базис подпространтсва V_k как набор $\langle X_{\tau_1}, \ldots, X_{\tau_k}\rangle$, где $X_{\tau_i}, i = (1, \ldots, k)$ — линейная комбинация над векторами из базиса пространства.

Зададим полный флаг на пространстве V_m:

$$V_0 = \{0\} \subset V_1 = \langle X_1\rangle \subset \cdots \subset V_m = \langle X_1, \ldots, X_m\rangle.$$

Выбор такого полного флага даёт возможность определить точное соответствие между клеточным разбиением Грассманиана и клеточными матрицами вида 3.1. Зафиксируем вектор t, содержащий коэффициенты a_i матрицы 3.1, притом для некоторого индекса k коэффициент $a_k = 1$, после него идут нули, а до него любые элементы из поля $\mathbb{F}_q$. Очевидно, что этот вектор будет удовлетворять виду клеточной матрицы разбиения, а значит $t \in V_k \setminus V_{k-1}$. Однако при рассмотрении других векторов в $V_k \setminus V_{k-1}$ появится проблема в невозможности перестановки в порядке, определенном рекуррентными соотношениями, без нарушения согласованности с флагом, поскольку найдется элемент $v \in V_i$ и $\omega \in V_m \setminus V_i$, при котором будет выполняться $\omega \prec v$, что противоречит определению 4.2.1. □

Таким образом, получаем, что метод неприменим в связи с отсутствием согласованности такого порядка с флагом.

Глава 5

Приложения

5.1. Применение точного критерия Фишера

Рассмотрим группу больных с заболеваниями щитовидной железы. Им проводилась операция по удалению части или всей железы. Требуется рассмотреть зависимость вероятности рецидива от методов лечения и протекания болезни.
Факторы:

- Объем операции — ТЭ (тиреоидэктомия — полное удаление ткани щитовидной железы), либо ТЭ + ЦЛ, либо ТЭ + ЦЛ + БЛ.

- РЙТ — радиойодтерапия. Радиойодтерапия применяется при лечении тиреотоксикоза, сопровождающего диффузный токсический зоб, автономно функционирующие аденомы. Радиоактивный йод также используют при лечении дифференцированного рака щитовидной железы в качестве дополнительного метода и/или при терапии рецидивов заболевания (повторное заболевание), регионарных и отдаленных метастазов после хирургического этапа.

- НИС проточник (в %) — натрий/йод симпортер. Этот белок модифицирует транспорт йода не только в щитовидную, но и в лактирующую молочную железу, а также в некоторые другие ткани. Его вариабельная экспрессия характерна и для рака щитовидной железы, и для рака молочной железы. НИС рассматривается как потенциальный переносчик циторедуктивных препаратов в ткань рака щитовидной железы и может использоваться в диагностике этого заболевания.

- BRAF — это ген, который вместе с рецепторами эпидермального фактора роста (EGFR) отвечает за один из сигнальных путей в клетке. В нормальной клетке сигнальный путь находится в неактивном состоянии, поскольку факторы роста связаны со своими рецепторами EGFR. При развитии мутации патологический ген начинает продуцировать активированный белок. Этот белок запускает механизм избыточного переноса сигнала к факторам роста. Результатом является в несколько раз ускоренное размножение клеток и рост новообразования.

- Стадии. Определение стадии – это способ описания рака, т.е. определение его локализации, распространения и влияния на функции других органов тела. Врачи используют диагностические тесты для определения стадии рака, поэтому до окончания всех тестов определение стадии не может быть завершено. Знание стадии помогает врачу определить наиболее подходящий вид лечения и сделать прогноз выздоровления. Существуют различные описания стадий при различных видах рака. Одним из инструментов определения стадии является система TNM. Эта система использует для оценки стадии рака три критерия: саму опухоль, близлежащие лимфоузлы и распространение опухоли на другие части тела. Для определения стадии рака результаты объединяются. Существуют пять стадий: стадия 0(нулевая стадия) и стадии I – IV. Стадия даёт возможность общего способа описания рака и совместной работы врачей для наилучшего планирования лечения. TNM – аббревиатура от T(опухоль), N(узел) и M(метастазы). Врачи рассматривают три эти фактора для определения стадии рака:

  1. Насколько велика первичная опухоль и где она расположена? (Опухоль, T)
  2. Распространилась ли опухоль на лимфоузлы? (Узел, N)
  3. Есть ли метастазы рака в других частях тела? (Метастазы, M)

- Появление рецидива.

- Время наступления рецидива.

Основные факторы риска: мутация гена BRAF, стадии заболевания и НИС проточник, поэтому необходимо ответить на следующие вопросы:

1. есть ли зависимость между уровнем НИС (в %) и положительным BRAF статусом?
2. есть ли зависимость НИС (в %) от T-стадии?
3. есть ли зависимость НИС (в %) от N-стадии?
4. есть ли зависимость стадии болезни от наличия мутации BRAF?
5. зависит ли время наступления рецидива от уровня НИС (в %)?

6. чаще ли наступает рецидив у больных с положительным BRAF и низким НИС (1% и менее) по сравнению с остальными?

Определять зависимость будем с помощью точного критерия Фишера и соответствующей программы SPSS.

Будем отвечать на данные вопросы по порядку.
Уровень НИС (в %) разобьем на 4 категории: низкий ($\leq$ 1%), средний (1% – 5%), крупный (5% – 10%) и высокий ($\geq$ 10%).

На основании полученных данных составим таблицы сопряженности для ответов на вопросы выше, после чего их проанализируем.

Пусть точное p-значение > 0.05, тогда будем считать факторы независимыми, в противном случае будет доказана существенная зависимость. Основываясь на анализе этих таблиц сопряженности (таблицы 2.3 – 5.7) можем прийти к следствию о том, что факторами, влияющими на наличие рецидива, являются: Т-стадия, уровень НИС и объем проведенной операции, при этом также близка к зависимости мутация BRAF и Т-стадия. Таким образом можно сделать вывод о том, что ключевые факторы риска — это мутация BRAF и уровень НИС, но при корректной операции и правильной оценке начальной стадии рецидива болезни можно избежать в большинстве случаев.
Ниже в таблицах 5.1 – 5.7 представлены результаты, полученные моей программой, и также для сравнения приведены результаты выполнения аналогичной программы в SPSS Statistics.

Таблица 5.1. Таблица сопряженности BRAF и Т-стадии

BRAF/Т-стадия	1	2	3	4	Итого
норма	12	5	3	1	21
мутантный	16	0	3	3	22
Итого	28	5	6	4	43

Таблица 5.2. Сравнение BRAF и T-стадии в SPSS

Точный критерий Фишера	0.088
Хи-квадрат Пирсона	0.096
Отношение правдоподобия	0.085

Таблица 5.3. Таблица сопряженности BRAF и N-стадии

BRAF/N-стадия	0	1	Итого
норма	13	8	21
мутантный	8	14	22
Итого	21	22	43

Таблица 5.4. Сравнение BRAF и N-стадии в SPSS

Точный критерий Фишера	0.131
Хи-квадрат Пирсона	0.131
Отношение правдоподобия	0.131

Таблица 5.5. Таблица сопряженности объема операции и наличия рецидива

Объем/Рецидив	нет	есть	Итого
ТЭ	8	4	12
ТЭ + ЦЛ	19	0	19
ТЭ + ЦЛ + БЛ	6	6	12
Итого	33	10	43

Таблица 5.6. Сравнение объема операции и наличия рецидива в SPSS

Точный критерий Фишера	0.001
Хи-квадрат Пирсона	0.006
Отношение правдоподобия	0.001

Таблица 5.7. Таблица полученных моей программой значений

Сравнение	Значение точного критерия Фишера
НИС и T-стадия	0.7509
BRAF и T-стадия	0.0871
BRAF и N-стадия	0.1308
Объем операции и наличие рецидива	0.0007

Заключение

Результатом выполнения бакалаврской работы стало детальное изучение точного критерия Фишера для различных размерностей, составлены программы для частных случаев $2 \times 2, 2 \times 3, 2 \times 4, 3 \times 3$ размерностей таблиц сопряженности, в дальнейшем обобщенные на общий случай таблиц $r \times c$. Изучено применение грассманиана в задачах статистики, понятия симптома, синдрома, их взаимосвязь и применение. Реализован алгоритм быстрого перечисления точек грассманиана с выбором наилучшего синдрома. Рассмотрен альтернативный способ перечисления точек грассманиана, основанный на перечислении диаграмм Юнга. Обе программы применены на реальных данных и на их основе сделан вывод о значимых факторах, влияющих на исход болезни, а также выявлена взаимосвязь между исходными симптомами болезни, лечением и послеоперационными осложнениями.

Литература

1. Agresti A. Categorical data analysis // New York: Wiley. — 1990.
2. Mehta C. R., Patel N. R. — IBM SPSS Exact tests. — IBM Corporation, 2011.
3. Suzukiyz T., Aokiyx S., Murotay K. Use of primal - dual technique in the network algorithm for two - way contingency tables // Japan Journal of Industrial and Applied Mathematics. — 2004. — Vol. 22. — P. 133–145.
4. Verbeek A. A survey of algorithms for exact distributions of test statistics in r x c contingency tables with fixed margins // Computational Statistics and Data Analysis. — 1985. — Vol. 3. — P. 159–185.
5. URL: http://www.statisticshowto.com/tables/z-table/.
6. URL: http://www.sigmazone.com/binomial_confidence_interval.htm.
7. Mehta C., Patel N. A network algorithm for performing fisher's exact test in $r \times c$ contingency tables // Journal of the American Statistical Association. — 1983. — Vol. 78:382. — P. 427–434.
8. Pagano M., Halvorsen K. An algorithm for finding the exact significance levels of $r \times c$ contingency tables // Journal of the American Statistical Association. — 1981. — Vol. 78. — P. 427–434.
9. Smith P., Forster J., McDonald J. Monte carlo exact tests for square contingency tables // Journal of the Royal Statistical Society A. — 1996. — Vol. 159. — P. 309–321.
10. Yates F. Contingency tables involving small numbers and the χ^2 test // Journal of Royal Statistical Society, Supplementary. — 1934. — Vol. 1. — P. 217–235.
11. Ананьевская П. В. Исследование конечно-линейных статистических моделей, оптимизация и избыточность : дис. на соискание степени к. ф.-м. н. / П. В. Ананьевская ; С. - Петербургский государственный университет. — 2013. — 142 с.
12. Гриффитс Ф., Харрис Д. Принципы алгебраической геометрии. — Мир, 1982.
13. Городенцев А. Л. Алгебра-1. — МЦНМО, 2011. — С. 526.

14. Казарян М. Э. Введение в теорию когомологий. — МИАН, 2006. — Т. 3. — С. 106.

15. Алексеева Н. П. Анализ медико-биологических систем // Издательство С. - Петербургского университета. — 2012. — 185 с.