Министерство образования и науки Российской Федерации
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

В.М. Неделько

Основы статистических методов машинного обучения

Учебное пособие

Новосибирск
2010

УДК 519.25 004.852

В.М. Неделько. Основы статистических методов машинного обучения. Учебное пособие.

В учебном пособии излагаются основы теории и методов машинного обучения в вероятностной постановке. Под машинным обучением понимается анализ данных, при котором выявляются закономерности или строятся модели, описывающие данные. Дисциплина базируется на методах математической статистики.

Пособие предназначено для студентов ФПМИ НГТУ.

Рецензенты: д.т.н., проф. Г.С. Лбов
к.ф.-м.н. Т.А. Ступина

Работа подготовлена кафедрой ПС и БД.

Содержание

Введение

Пособие предназначено для первоначального знакомства со статистическими методами машинного обучения и ориентировано на студентов, изучающих данную дисциплину в рамках соответствующего учебного курса, а также ведущих научно-исследовательскую работу в области анализа данных.

Для дальнейшего изучения предмета рекомендуются учебные пособия [1, 2, 3], а также монографии [4, 5, 6], написанные доступным языком.

Задачи машинного обучения в вероятностной или статистической постановке являются непосредственным обобщением и продолжением задач прикладной математической статистики, поэтому для лучшего понимания материала желательно знакомство с этой дисциплиной [7–11].

В первой главе пособия даются наглядные примеры различных задач машинного обучения: классификации, регрессионного анализа, прогнозирования временных рядов. Задачи иллюстрируют применения базовых методов. Для понимания большей части изложения этой главы достаточно знаний средней школы.

Во второй главе производится систематизация материала, представленного в предыдущей главе, и даётся формальное изложение постановок задач и методов машинного обучения.

Глава 1. Некоторые задачи и методы машинного обучения

В данной главе будут описаны некоторые наиболее распространённые задачи и методы машинного обучения.

§1.1. Метод прецедентов

Метод прецедентов является, пожалуй, наиболее простым методом машинного обучения и при этом одним из наиболее эффективных.

1.1.1. Задача классификации

Для иллюстрации методов машинного обучения будем использовать упрощённые варианты известных прикладных задач. Наиболее известной коллекцией задач машинного обучения является UCI (University of California, Irvine) Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets.html), содержащий на данный момент 174 набора данных из различных прикладных задач.

Для знакомства с методом прецедентов рассмотрим одну простую практическую задачу машинного обучения, а именно задачу «Iris» из репозитория UCI.

Задача состоит в том, чтобы построить правило, позволяющее по внешним признакам различать три вида (сорта) ирисов (цветы):

1 – Iris Setosa,
2 – Iris Versicolour,
3 – Iris Virginica.

В качестве измеряемых характеристик используются:

X_1 – длина чашелистика (sepal length),
X_2 – длина лепестка (petal length).

Таким образом, требуется по измеренным значениям переменных X_1 и X_2 определить вид ириса.

Вид ириса обозначим переменной Y, которая будет принимать значения 1, 2, 3.

Результаты измерений указанных характеристик для 36 экземпляров растений трёх видов приведены в таблице 1. Здесь i обозначает номер объекта (экземпляра) и в верхней позиции обозначает не степень, а верхний индекс, x_1^i – значения переменной X_1, x_2^i – значения переменной X_2, y^i – значения переменной Y.

Заметим, что в исходной таблице данных, доступной в репозитории, используются четыре характеристики, а число объектов составляет 150, однако, для наглядности мы рассматриваем сокращённую таблицу.

Таблица 1. Данные по ирисам.

i	x_1^i	x_2^i	y^i	i	x_1^i	x_2^i	y^i	i	x_1^i	x_2^i	y^i
1	5,1	1,4	1	*13*	6	4	2	*25*	6,7	5,7	3
2	4,9	1,4	1	*14*	6,1	4,7	2	*26*	7,2	6	3
3	4,7	1,3	1	*15*	5,6	3,6	2	*27*	6,2	4,8	3
4	4,6	1,5	1	*16*	6,7	4,4	2	*28*	6,1	4,9	3
5	5	1,4	1	*17*	5,6	4,5	2	*29*	6,4	5,6	3
6	5,4	1,7	1	*18*	5,8	4,1	2	*30*	7,2	5,8	3
7	4,6	1,4	1	*19*	6,2	4,5	2	*31*	7,4	6,1	3
8	5	1,5	1	*20*	5,6	3,9	2	*32*	7,9	6,4	3
9	4,4	1,4	1	*21*	5,9	4,8	2	*33*	6,4	5,6	3
10	4,9	1,5	1	*22*	6,1	4	2	*34*	6,3	5,1	3
11	5,4	1,5	1	*23*	6,3	4,9	2	*35*	6,1	5,6	3
12	4,8	1,6	1	*24*	6,1	4,7	2	*36*	7,7	6,1	3

Для наглядности изобразим имеющиеся данные на диаграмме (см. рис. 1).

Требуется, проанализировав представленные данные (будем называть их выборкой), научиться распознавать сорт ириса по внешним признакам (X_1 и X_2). Математически это означает, что необходимо построить функцию f, сопоставляющую любой паре (x_1, x_2) значений переменных X_1, X_2 некоторое значение y переменной Y.

Такая функция называется решающей функцией или решающим правилом:

$$f : X_1 \times X_2 \to Y \quad \text{или} \quad y = f(x_1, x_2),\ y \in Y,\ x_1 \in X_1,\ x_2 \in X_2 .$$

Здесь и далее мы отождествляем переменную и множество значений, которые она принимает.

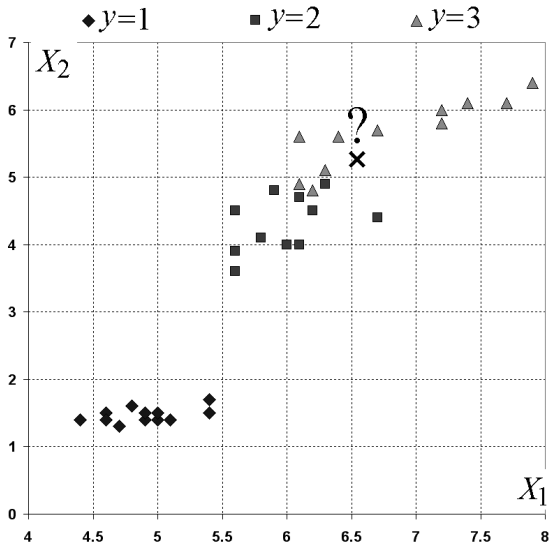

Рис. 1. Визуализация данных из таблицы 1.

Идея метод прецедентов заключается в следующем. Для объекта, который нужно классифицировать, находятся наиболее похожие на него объекты, для которых целевой признак известен.

Простейшая модификация метода заключается в том, что распознаваемый объект относится к тому же классу, к которому принадлежит объект, на который он более всего похож. Такой метод носит название метода ближайшего соседа.

В качестве «меры похожести» можно использовать расстояние ρ между точками, представляющими объекты в признаковом пространстве.

Так например для точки (6,6, 5,2), которая обозначена на рис. 1 крестиком, ближайшей точкой выборки будет точка (6,4, 5,6), которая принадлежит третьему классу (y = 3), поэтому точке (6,6, 5,2) также припишем третий класс, т.е. $f(6{,}6, 5{,}2) = 3$.

Введём формальное обозначение для выборки

$$\nu = \left((x_1^i, x_2^i, y^i) \mid i = 1,\ldots, N \right) = \left((x^i, y^i) \mid i = 1,\ldots, N \right),$$

где $x^i = (x_1^i, x_2^i)$, N – количество объектов.

Здесь мы просто ввели буквенные обозначения для таблицы данных и её содержимого.

Теперь решающую функцию, получаемую методом ближайшего соседа, можно записать формально

$$f(x) = y^{i_x}, \text{ где } i_x = \arg\min_i \rho(x, x^i),$$

где arg min – аргумент минимума, т.е. в данном случае – индекс i, при котором достигается минимум расстояния.

Существует модификация метода, когда вместо одной точки находятся несколько ближайших точек выборки и приписывается класс, к которому принадлежит большинство из них. Это метод k ближайших соседей.

1.1.2. Задача восстановления зависимостей

В предыдущем разделе переменная Y принимала конечный набор значений, на которых не был определён порядок (так как на сортах ирисов не определено отношение «больше-меньше»). В таких случаях задача построения решающей функции называется задачей распознавания образов или задачей классификации (с «учителем»).

Если целевая переменная Y является вещественной, то говорят о задаче восстановления зависимостей. Задачу построения решающей функции при вещественной Y можно также называть задачей регрессионного анализа, хотя это будет не совсем точно, поскольку регрессионный анализ предполагает, что значения переменных, кроме целевой, не являются случайными.

В качестве примера задачи восстановления зависимостей рассмотрим задачу «Wine Quality» из репозитория UCI.

База данных для этой задачи описывает несколько тысяч образцов вин, которые характеризуются одиннадцатью переменными, соответствующими измерениям различных химических показателей. Целевая переменная Y принимает целые значения от 3 до 8 и является экспертной оценкой качества вина.

Поскольку дробные оценки качества не противоречат содержательному смыслу задачи, будем считать переменную Y вещественной.

Как и в предыдущем случае, уменьшим размерность задачи. Выберем две переменных:

X_1 – концентрация летучих кислот (volatile acidity),

X_2 – концентрация лимонной кислоты (citric acid).

Таблица 2. Данные по задаче оценивания качества вин.

i	x_1^i	x_2^i	y^i	i	x_1^i	x_2^i	y^i	i	x_1^i	x_2^i	y^i
1	1,02	0,02	3	*11*	0,7	0,13	5	*21*	0,36	0,34	7
2	0,44	0,42	3	*12*	0,59	0,01	5	*22*	0,3	0,41	7
3	0,61	0,49	3	*13*	0,49	0,49	5	*23*	0,59	0	7
4	0,76	0,02	3	*14*	0,82	0,29	5	*24*	0,34	0,4	7
5	1,19	0	3	*15*	0,69	0,49	5	*25*	0,59	0,06	7
6	0,33	0,32	4	*16*	0,58	0,56	6	*26*	0,35	0,53	8
7	0,58	0	4	*17*	0,56	0,24	6	*27*	0,62	0,67	8
8	0,92	0,27	4	*18*	0,39	0,47	6	*28*	0,49	0,03	8
9	0,48	0,2	4	*19*	0,49	0,1	6	*29*	0,32	0,45	8
10	1,13	0,09	4	*20*	0,51	0,13	6	*30*	0,3	0,56	8

Из всего множества объектов выберем по пять образцов для каждого значения качества. Заметим, что после такого сокращения данных решение, которое будет получено, нельзя переносить на исходную задачу, фактически мы сформулировали новую учебную задачу на основе данных UCI.

Сформированная выборка приведена в таблице 2 и визуализирована на рис. 2, где цифры рядом с точками отражают значения целевой переменной.

Задача заключается в построении решающей функции, которая по значениям химических показателей позволяет оценить качество вина.

Пусть, например, мы получили образец, для которого $x_1 = 0{,}5$, $x_2 = 0{,}35$. Он изображён точкой А на рис. 2. Что можно сказать о его качестве? Наиболее похожим из выборки является объект $i = 2$ со значениями (0,44, 0,42), для которого $y = 3$. Таким образом, по методу ближайшего соседа оценка качества для объекта А также составит 3.

Однако заметим, что недалеко от А находятся и точки, для которых $y = 7$. И при незначительном смещении точки А оценка по методу ближайшего соседа может сильно измениться.

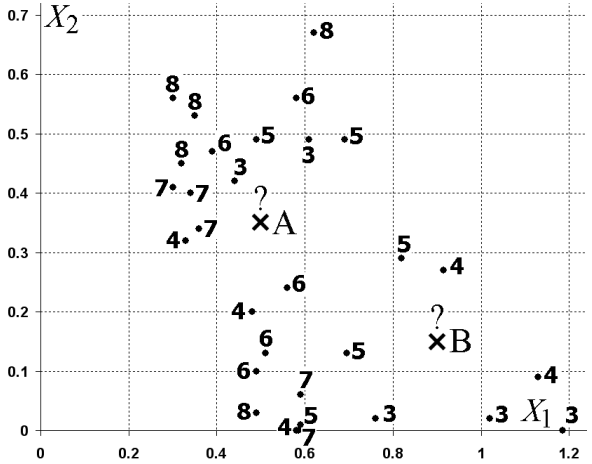

Рис. 2. Визуализация данных из таблицы 2.

Подобное скачкообразное изменение решающей функции представляется неоправданным, поскольку разумно считать, что

качество продукта в данном случае является непрерывной функцией характеристик.

Для построения гладкой решающей функции применим так называемый ядровой (kernel-based) метод, также известный как метод потенциальных функций.

В этом методе решающая функция строится в виде

$$f(x)=\frac{1}{w}\sum_{i=1}^{N} y^i w_i\ ,\ \ w=\sum_{i=1}^{N} w_i\ ,$$

где $w_i=\varphi(\rho(x,x^i))$, $\rho(x,x^i)$ – расстояние между заданной точкой и объектом выборки, а $\varphi(\cdot)$ – потенциальная или ядровая функция.

В качестве потенциальной функции можно использовать любую монотонно убывающую функцию, например, ядро Коши

$$\varphi(z)=\frac{1}{1+\left(\frac{z}{r}\right)^2}\ ,$$

где r – параметр, задающий скорость убывания функции.

На содержательном уровне идея метода в том, что решение в каждой точке есть взвешенное среднее по выборочным значением, причём, чем ближе выборочная точка к данной, тем в большим весом она учитывается.

Для примера вычислим оценки качества в точках А и В. При $r=0{,}1$ в точке А имеем $f(0{,}5,\,0{,}35)\approx 5{,}49$, для точки В получаем $f(0{,}9,\,0{,}15)\approx 4{,}7$.

Данный результат выглядит достаточно разумным. Действительно, согласно выборке, качество вина выше при меньшей концентрации летучих кислот (в т.ч. уксусной), и большей концентрации лимонной кислоты.

§ 1.2. Классификация в пространстве бинарных переменных

Бинарными называются переменные, принимающие одно из двух значений: «истина» и «ложь», которые обозначаются соот-

ветственно 1 и 0. На бинарных переменных мы проиллюстрируем метод классификации, основанный на оценивании распределений вероятностей.

1.2.1. Случай известных распределений

Рассмотрим задачу «Congressional Voting Records» из репозитория UCI.

Представлены результаты голосований 168 конгрессменов-республиканцев и 267 конгрессменов-демократов по 16 законопроектам.

Задача заключается в том, чтобы определить партийную принадлежность депутата по результатам его голосований.

Целевая переменная Y отражает партийную принадлежность:

0 – демократ,

1 – республиканец.

Из 16 исходных выберем 4 переменных, отражающих решение, принятое конгрессменом по соответствующему законопроекту:

X_1 – handicapped-infants,

X_2 – religious-groups-in-schools,

X_3 – anti-satellite-test-ban,

X_4 – immigration.

Значение 1 переменной означает, что сенатор голосовал «за», значение 0 – «против». Названия законопроектов оставлены без перевода, поскольку они несущественны.

В представленных данных имеются пропуски, т.е. неизвестные значения переменных (возможно, что депутат отсутствовал на голосовании, либо воздержался, либо результат не был опубликован).

Поскольку работа с пропусками требует специальных модификаций методов, все объекты с пропуском значения для любой из четырёх выбранных переменных, были исключены из рассмотрения. В результате осталось 160 объектов с $y = 1$ и 243 с $y = 0$.

При случайном равновозможном выборе объекта из оставшихся в рассмотрении вероятность выбора демократа составляет

$P(y=0)=\frac{243}{160+243}\approx 0{,}6$, вероятность выбора республиканца есть $P(y=1)=1-P(y=0)\approx 0{,}4$.

Таблица 3. Распределение вероятностей и решающая функция.

X_1	X_2	X_3	X_4	$P_1(x)$	$P_0(x)$	$g_1(x)$	$g_0(x)$	$f(x)$	$R(x)$
0	0	0	0	0,025	0,004	0,81	0,19	1	0,19
0	0	0	1	0,006	0,004	0,5	0,5	1	0,5
0	0	1	0	0	0,054	0	1	0	0
0	0	1	1	0,031	0,07	0,23	0,77	0	0,23
0	1	0	0	0,288	0,074	0,72	0,28	1	0,28
0	1	0	1	0,343	0,049	0,82	0,18	1	0,18
0	1	1	0	0,05	0,062	0,35	0,65	0	0,35
0	1	1	1	0,063	0,087	0,33	0,67	0	0,33
1	0	0	0	0	0,012	0	1	0	0
1	0	0	1	0,006	0,004	0,5	0,5	1	0,5
1	0	1	0	0,013	0,243	0,03	0,97	0	0,03
1	0	1	1	0,025	0,136	0,11	0,89	0	0,11
1	1	0	0	0,038	0,049	0,34	0,66	0	0,34
1	1	0	1	0,056	0,037	0,50	0,50	1	0,50
1	1	1	0	0,025	0,033	0,34	0,66	0	0,34
1	1	1	1	0,031	0,082	0,20	0,80	0	0,20

В таблице 3 приведены вероятности того, что выбранный наугад конгрессмен окажется проголосовавшим определённым образом, то есть вероятности выбора объекта с заданными значениями переменных $X_1, \ldots, X_4$. Здесь $P_1(x)=P(x/y=1)$ есть вероятность того, что переменные $X_1, \ldots, X_4$ примут заданный набор значений $x=(x_1,x_2,x_3,x_4)$ при условии $y=1$, а $P_0(x)=P(x/y=0)$. Эти вероятности вычислены по таблице исходных данных, объекты которой выступают в роли генеральной совокупности.

Пользуясь формулой Байеса, вычислим вероятности, что выбранный объект принадлежит республиканцам, если известно, как он проголосовал:

$$g_1(x) = P(y=1/x) = P(x/y=1)\frac{P(y=1)}{P(x)} = P_1(x)\frac{P(y=1)}{P(x)},$$

где $P(x) = P_1(x)P(y=1) + P_0(x)P(y=0)$. Очевидно, что

$$g_0(x) = P(y=0/x) = 1 - g_1(x).$$

Глядя на таблицу, видим, что например, сенатор, проголосовавший против всех законопроектов, с вероятностью 0,81 окажется республиканцем и с вероятностью 0,19 — демократом.

Если теперь потребуется ответить, кем является сенатор, проголосовавший против всех законопроектов, то естественно ответить, что республиканцем, т.е. $f(0,0,0,0)=1$. Давая такой ответ, мы с вероятностью 0,19 ошибёмся, однако, давая противоположный ответ, мы ошибёмся с вероятностью 0,81. Таким образом, принятое нами решение, минимизирует вероятность ошибочной классификации.

Выпишем формально решающую функцию, которая минимизирует вероятность ошибочной классификации:

$$f(x)=1, \text{ при } g_1(x) > g_0(x);\ f(x)=0, \text{ при } g_1(x) < g_0(x).$$

При $g_1(x) = g_0(x)$ вероятность ошибочной классификации составляет 0,5 при любом решении, поэтому решающая функция в этом случае может прогнозировать любой из классов.

Вероятность ошибочной классификации для приведённой решающей функции при заданном x составляет

$$R(x) = \min(g_1(x), g_0(x)).$$

Общая вероятность ошибочной классификации есть

$$R = \sum_x R(x)P(x).$$

В рассматриваемой задаче $R \approx 0{,}21$.

Рассмотренная решающая функция называется байесовской, а величина R называется байесовским уровнем ошибки. Такое на-

звание связано, очевидно, с использованием формулы Байеса при записи этой функции.

Термин байесовская решающая функция используется как синоним оптимальной решающей функции. Данный термин общеупотребителен, однако его следует признать неудачным, поскольку во-первых, оптимальную решающую функцию можно сформулировать через совместные распределения $P(x,y)$ без использования формулы Байеса, во-вторых, может возникнуть ложная ассоциация с байесовскими методами построения решающих функций, которые не имеют ничего общего с обсуждаемым понятием.

1.2.2. Выборочная оценка

Предположим теперь, что нам недоступна полная таблица данных, а имеется лишь небольшая выборка из неё. Предположим также, что эта выборка является случайной и независимой (последнее, в частности, означает, что произведён выбор «с возвращением»).

Пусть выбрано 40 конгрессменов, из которых 15 оказались республиканцами (таблица 4), а 25 — демократами.

Задача состоит в том, чтобы по имеющейся выборке построить решающую функцию.

Ранее мы нашли оптимальную решающую функцию при известных распределениях. Когда распределения неизвестны, мы можем оценить их по выборке и подставить оценки вместо вероятностей в выражение для решающей функции. Для этого нужно оценить вероятности $P(x,y)$ или, что эквивалентно, $P(y)$ и $P(x/y)$.

Вероятность $P(y=1)$ того, что сенатор окажется республиканцем, может быть оценена как доля республиканцев среди выбранных объектов: $\widetilde{P}(y=1)=\frac{15}{40}=0{,}375$. Соответственно $\widetilde{P}(y=0)=\frac{25}{40}=0{,}625$. Остаётся оценить $P_1(x)=P(x/y=1)$ и $P_0(x)=P(x/y=0)$.

Указанные два условных распределения оцениваются аналогично друг другу, поэтому рассмотрим только оценивание первого из них.

Поскольку с этого момента мы будем работать только с $P_1(x)$, индекс 1 будет только мешать, поэтому далее мы его будем опускать и писать просто $P(x)$.

На самом деле, при оценивании распределения нам неважно, является оно условным или безусловным — но важно, чтобы выборка была сформирована при том же условии, при котором рассматривается распределение.

Таблица 4. Выборка из сенаторов-республиканцев.

i	x_1^i	x_2^i	x_3^i	x_4^i	i	x_1^i	x_2^i	x_3^i	x_4^i
1	0	0	0	0	*9*	0	1	0	1
2	0	0	0	1	*10*	0	1	1	1
3	0	1	0	0	*11*	1	0	1	1
4	0	1	0	0	*12*	1	1	0	0
5	0	1	0	1	*13*	1	1	0	0
6	0	1	0	1	*14*	1	1	1	0
7	0	1	0	1	*15*	1	1	1	1
8	0	1	0	1					

Итак, имеем выборку из 15 сенаторов-республиканцев, приведённую в таблице 4.

Подсчитаем $N(x)$ – количество раз, которые встречается каждая комбинация $x=(x_1,x_2,x_3,x_4)$ переменных X_1, …, X_4 .

Простейшей оценкой вероятности $P(x)$ будет $\tilde{P}(x)=\frac{N(x)}{N}$ – доля объектов со значением *x*. Значения приведены в таблице 5.

Чтобы понять, насколько хороша полученная оценка, сравним её с истинным распределением $P(x)$, которое приведено в таблице 3, столбец $P_1(x)$, а также в таблице 6.

Таблица 5. Выборочные оценки распределения.

X_1	X_2	X_3	X_4	$N(x)$	$\widetilde{P}(x)$	$\widetilde{P}^1(x)$	$\widetilde{P}^*(x)$	$\widetilde{P}^2(x)$	$\widetilde{P}^3(x)$
0	0	0	0	1	0,067	0,039	0,039	0,033	0,075
0	0	0	1	1	0,067	0,059	0,036	0,078	0,058
0	0	1	0	0	0	0,014	0,020	–0,002	–0,008
0	0	1	1	0	0	0,021	0,019	0,025	0,008
0	1	0	0	2	0,133	0,156	0,110	0,184	0,125
0	1	0	1	5	0,333	0,235	0,332	0,305	0,342
0	1	1	0	0	0	0,057	0,058	–0,015	0,008
0	1	1	1	1	0,067	0,085	0,054	0,060	0,058
1	0	0	0	0	0	0,020	0,029	0,018	–0,008
1	0	0	1	0	0	0,029	0,027	0,004	0,008
1	0	1	0	0	0	0,007	0,016	0,017	0,008
1	0	1	1	1	0,067	0,011	0,014	0,027	0,058
1	1	0	0	2	0,133	0,078	0,084	0,097	0,142
1	1	0	1	0	0	0,117	0,078	0,014	–0,008
1	1	1	0	1	0,067	0,028	0,044	0,067	0,058
1	1	1	1	1	0,067	0,043	0,041	0,089	0,075

Введём следующую меру расхождения распределений

$$\Delta(P', P'') = \sum_{x \in X} \left| P'(x) - P''(x) \right|.$$

Теперь можем вычислить отличие $\widetilde{P}(x)$ от $P(x)$. Просуммировав модули разностей значений в шестом столбце таблицы 5 и пятом столбце таблицы 6, получим $\Delta\left(\widetilde{P}(x), P(x)\right) = 0{,}64$.

Полученное значение весьма велико (максимально возможное значение Δ равно 2), т.е. оценка $\widetilde{P}(x)$ оказалась очень неточной, что неудивительно, поскольку объем выборки очень мал по сравнению с числом оцениваемых параметров распределения. Действительно, при объеме выборки 15 мы оцениваем 16 вероятностей, связанных одним соотношением (сумма равна 1), что даёт 15 свободных параметров.

1.2.3. Гипотеза независимости переменных

Проверим, являются ли переменные $x_1,...x_4$ независимыми. Для этого вычислим вероятности

$$p_j = P(x_j = 1),\ \ j = 1,...,4\,.$$

Суммируя значения в столбце $P(x)$ таблицы 6 по строкам, в которых $x_1 = 1$, получаем

$$p_1 = P(1,0,0,0) + P(1,0,0,1) + P(1,0,1,0) + P(1,0,1,1) +$$
$$+ P(1,1,0,0) + P(1,1,0,1) + P(1,1,1,0) + P(1,1,1,1) =$$
$$= 0 + 0{,}006 + 0{,}013 + 0{,}025 + 0{,}038 + 0{,}056 + 0{,}025 + 0{,}031 = 0{,}194.$$

Таблица 6. Аппроксимация распределения рядом Бахадура.

X_1	X_2	X_3	X_4	$P(x)$	$P^1(x)$	$P^2(x)$	$P^3(x)$
0	0	0	0	0,025	0,029	0,004	0,024
0	0	0	1	0,006	0,037	0,013	0,007
0	0	1	0	0	0,009	0,017	0,001
0	0	1	1	0,031	0,011	0,028	0,030
0	1	0	0	0,288	0,241	0,307	0,289
0	1	0	1	0,343	0,308	0,338	0,342
0	1	1	0	0,05	0,075	0,035	0,049
0	1	1	1	0,063	0,096	0,064	0,064
1	0	0	0	0	0,007	0,007	0,001
1	0	0	1	0,006	0,009	0,013	0,005
1	0	1	0	0,013	0,002	0,010	0,012
1	0	1	1	0,025	0,003	0,014	0,026
1	1	0	0	0,038	0,058	0,033	0,037
1	1	0	1	0,056	0,074	0,047	0,057
1	1	1	0	0,025	0,018	0,026	0,026
1	1	1	1	0,031	0,023	0,044	0,030

Суммируя значения по строкам, в которых $x_2 = 1$, получаем

$$p_2 = P(0,1,0,0) + P(0,1,0,1) + P(0,1,1,0) + P(0,1,1,1) +$$
$$+ P(1,1,0,0) + P(1,1,0,1) + P(1,1,1,0) + P(1,1,1,1) = 0{,}894\,.$$

Аналогичным образом вычисляем

$$p_3 = 0{,}238, \qquad p_4 = 0{,}561.$$

Заметим, что

$$P(x_j = 0) = 1 - P(x_j = 1) = 1 - p_j.$$

Условие независимости переменных выглядит следующим образом

$$P(x_1, x_2, x_3, x_4) = P(x_1)P(x_2)P(x_3)P(x_4).$$

Таким образом, чтобы проверить независимость переменных, достаточно вычислить правую часть выражения и сравнить её с фактическим распределением в левой части.

Для удобства введём обозначение

$$P^1(x) = P^1(x_1, x_2, x_3, x_4) = P(x_1)P(x_2)P(x_3)P(x_4).$$

Переменные независимы, если $P(x) = P^1(x)$.

Вычислим $P^1(x)$ по формуле

$$P^1(x) = \prod_{j=1}^{4} \left((1 - x_j)(1 - p_j) + x_j p_j \right).$$

Выражение выглядит не вполне очевидным, однако смысл его простой: в зависимости о того, x_j равна 0 или 1, перемножаем соответственно $1 - p_j$ или p_j. Например,

$$P^1(0,1,1,0) = (1 - p_1)p_2 p_3(1 - p_4) \approx 0{,}075.$$

Значения $P^1(x)$ приведены в таблице 6.

Видим, что $P^1(x)$ не совпадает с $P(x)$, значит, переменные зависимы.

Величина

$$\Delta\left(P(x), P^1(x)\right) \approx 0{,}3$$

может служить характеристикой степени зависимости.

Хотя $P^1(x)$ не совпадает с $P(x)$, мы можем использовать $P^1(x)$ в качестве приближённых значений $P(x)$. Такое приближение имеет погрешность 0,3, которая также называется мерой адекватности модели, однако оно может быть полезно при оценивании распределения по выборке.

Если для оценивания $P(x)$ нам потребовалось оценить 15 свободных параметров, то при оценивании $P^1(x)$ нужно оценить лишь 4 параметра p_j.

Вычислим эти оценки.

В качестве оценки вероятности p_j возьмём $\tilde{p}_j = \frac{N(x_j=1)}{N}$ – долю объектов выборки, для которых $x_j = 1$.

Имеем

$$\tilde{p}_1 = 0{,}333\,, \quad \tilde{p}_2 = 0{,}8 \quad \tilde{p}_3 = 0{,}267 \quad \tilde{p}_4 = 0{,}6\,.$$

Вычислим оценку

$$\tilde{P}^1(x) = \prod_{j=1}^{4} \left((1 - x_j)(1 - \tilde{p}_j) + x_j \tilde{p}_j \right).$$

Полученные оценки приведены в таблице 5. Заметим, что вычисление $\tilde{P}^1(x)$ по таблице 5 полностью аналогично вычислению $P^1(x)$ по таблице 6.

Определим теперь, насколько хорошо $\tilde{P}^1(x)$ оценивает $P(x)$

$$\Delta\left(P(x), \tilde{P}^1(x)\right) \approx 0{,}54\,.$$

Видим, что хотя $\tilde{P}^1(x)$ получена в предположении независимости переменных, которое на самом деле в данном примере не выполняется, эта оценка оказалась лучше, чем $\tilde{P}(x)$.

Здесь мы имеем, на первый взгляд, парадоксальную ситуацию: сделав ложное предположение, мы получили лучший результат, чем не делая такого предположения. Это объясняется тем, что, оценивая распределение как для независимых переменных, мы

вносим определённую погрешность, если переменные на самом деле зависимы, однако эта погрешность меньше погрешности, с которой мы можем непосредственно оценить совместное распределение.

Этот пример иллюстрирует очень важное правило, которое следует иметь в виду при решении задач анализа данных. Если известно, что истинная модель (в нашем примере распределение) сложная, но объем выборки мал, то следует подбирать более простую (то есть заведомо неточную) модель, поскольку потеря точности (адекватности) модели будет компенсирована меньшей статистической погрешностью при оценивании параметров модели.

Построив оценки $\widetilde{P}^1(x)$ для обоих классов, можно найти оптимальное выборочное решающее правило, которое в некоторых источниках называется наивным байесовским классификатором. Достоинство данного термина — запоминаемость. В остальном термин не вполне удачен. Во-первых, выборочное решающее правило, вообще говоря, не является байесовским (т.е. оптимальным при заданном распределении). Во-вторых, использование такого классификатора не свидетельствует о наивности того, кто его использует. Хотя предположение о независимости переменных и можно назвать наивным, поскольку оно относительно редко выполняется на практике, в рассмотренном методе вовсе не предполагается, что переменные на самом деле независимы. Предполагается лишь то, что погрешность, внесённая моделью независимых переменных, будет скомпенсирована уменьшением статистической погрешности.

1.2.4. Ряд Бахадура

В предыдущем разделе мы рассмотрели две вероятностных модели: модель независимых переменных требовала оценивания n параметров, где n – число переменных, а модель совместного распределения имела $2^n - 1$ параметров. Если объём выборки существенно больше 2^n, то оправдано оценивать совместное распределение, если объём выборки сравним с n, то имеет смысл

использовать модель независимых переменных, даже если нет оснований ожидать, что переменные действительно независимы.

В данном разделе рассмотрим модель, которую уместно использовать, когда объём выборки много больше n, но меньше 2^n. Такая модель связана с разложением совместного распределения в ряд Бахадура.

Идея разложения в ряд Бахадура состоит в том, чтобы, начиная с приближения независимых переменных, последовательно учитывать парные зависимости переменных, зависимости в тройках, четвёрках и т.д.

Разложение в ряд Бахадура выглядит следующим образом

$$P(x) = P^1(x) \sum_{j=1}^{n} q_j(x).$$

Здесь

$$q_1(x) \equiv 1, \qquad q_2(x) = \sum_{j=1}^{n} \sum_{k=j+1}^{n} \rho_{jk} z_j z_k ,$$

$$q_3(x) = \sum_{j=1}^{n} \sum_{k=j+1}^{n} \sum_{l=k+1}^{n} \rho_{jkl} z_j z_k z_l , \ \ldots .$$

В этих выражениях использованы обозначения

$$z_j = \frac{x_j - p_j}{\sqrt{p_j(1-p_j)}},$$

$$\rho_{ik} = \sum_{x \in X} z_j z_k P(x), \ \ \rho_{ikl} = \sum_{x \in X} z_j z_k z_l P(x), \ \ldots .$$

Фактически здесь введены переменные z_j путём центрирования и нормирования на стандартное отклонение переменных x_j, при этом ρ_{jk} представляет собой коэффициент корреляции переменных z_j и z_k.

Для таблицы 6 имеем:

$$\rho_{12} = -0{,}193, \ \rho_{13} = 0{,}284, \ \rho_{14} = 0{,}047,$$
$$\rho_{23} = -0{,}334, \ \rho_{24} = -0{,}056, \ \rho_{34} = 0{,}078,$$

$$\rho_{123} = -0{,}269\,,\ \rho_{124} = -0{,}061\,,\ \rho_{134} = -0{,}025\,,\ \rho_{234} = -0{,}208\,,$$
$$\rho_{1234} = 0{,}022\,.$$

Вычислим теперь (значения приведены в таблице 6)

$$P^2(x) = P^1(x)\,(1 + q_2(x)) \quad \text{и} \quad P^3(x) = P^1(x)\,(1 + q_2(x) + q_3(x))\,.$$

Распределения $P^2(x)$ и $P^3(x)$ представляют собой частичные суммы ряда Бахадура, при этом $P^2(x)$ учитывает только парные зависимости переменных, $P^3(x)$ учитывает парные зависимости и зависимости в тройках. Полная сумма ряда $P^4(x)$ совпадает с исходным распределением $P(x)$.

Частичные суммы ряда Бахадура могут использоваться в качестве приближений для $P(x)$. Вычислим погрешности таких приближений для распределения в таблице 6

$$\Delta\left(P(x), P^2(x)\right) \approx 0{,}14\,, \quad \Delta\left(P(x), P^3(x)\right) \approx 0{,}01\,.$$

Как и следовало ожидать, точность приближения растёт с увеличением числа слагаемых.

Чтобы оценить частичные суммы ряда Бахадура по выборке, достаточно все фигурирующие в выражениях вероятности заменить частотами, т.е. фактически те же вычисления, что проводились для таблицы 6, проделать для значений в таблице 5.

Погрешности выборочных оценок оказываются следующими:

$$\Delta\left(P(x), \widetilde{P}^2(x)\right) \approx 0{,}53\,, \quad \Delta\left(P(x), \widetilde{P}^3(x)\right) \approx 0{,}64\,.$$

Видим, что при учёте парных корреляций оценка распределения оказалась чуть более точной, чем в модели независимых переменных, а при учёте зависимостей в тройках результат такой же, как и при непосредственном оценивании совместного распределения.

Такой результат можно объяснить следующим. Модель независимых переменных требует оценки всего четырёх параметров, но мера её неадекватности высока ($\Delta \approx 0{,}3$). Учёт парных корреляций требует оценивания ещё шести параметров (корреляций

для всех пар переменных), но мера неадекватности модели существенно меньше ($\Delta \approx 0{,}14$), и рост статистической погрешности оказался скомпенсированным повышением адекватности. Учёт зависимостей в тройках потребовал оценивания ещё четырёх параметров и, хотя модель стала практически полностью адекватной, общая погрешность возросла.

Следует заметить, что рассмотренный пример сам по себе, конечно, недостаточен для обоснования тех выводов, которые мы сделали на его основе. Действительно, мы рассмотрели всего одну генеральную совокупность объектов, всего одну выборку из неё, причём очень малого объёма. Однако как показывает практика, эти выводы достаточно универсальны, а именно, всегда следует выбирать сложность модели адекватной объёму выборки. Грубой рекомендацией здесь будет выбирать такую модель, чтобы число оцениваемых параметров было хотя бы в несколько раз меньше объёма выборки. Более точно выбрать сложность модели можно с помощью метода статистического моделирования, который обсуждается в разделе 1.8.2.

Также заметим, что рассмотренная задача, когда полная таблица данных рассматривается как генеральная совокупность и делается случайная выборка из неё, является искусственной. Более содержательной была бы постановка задачи, когда имеющаяся таблица данных является выборкой из некоторой генеральной совокупности. Однако такая постановка была бы слишком сложной и объёмной для рассмотрения здесь, в частности, из-за того, что генеральная совокупность была бы абстрактным понятием, а выборка зависимой (конгрессмены из одной фракции, как правило, согласовывают свои голоса).

§ 1.3. Дискриминантная функция для нормальных распределений

Нормальное распределение является, пожалуй, наиболее часто используемым в теории вероятностей и математической статистике. В задачах анализа данных зачастую нет оснований предполагать нормальность распределений, однако, в ряде случаев такую модель оправдано использовать.

1.3.1. Случай известных распределений

Для простоты будем рассматривать случай двух классов, т.е. $Y = \{1,2\}$, и пусть имеется одна непрерывная переменная X.

Предположим, что заданы распределение $P(y)$ и условные плотности вероятности $\varphi_y(x)$, $y \in \{1,2\}$, причём плотности являются нормальными

$$\varphi_y(x) = \frac{1}{\sqrt{2\pi}\sigma_y} e^{-\frac{(x-\mu_y)^2}{2(\sigma_y)^2}}.$$

Через заданные величины можно выразить совместную плотность вероятности

$$\varphi(x,y) = \varphi_1(x)P(1) + \varphi_2(x)P(2).$$

В качестве примера на рис. 3 приведён график совместной плотности при $P(1) = 0{,}4$, $P(2) = 0{,}6$, $\mu_1 = 1$, $\sigma_1 = 1$, $\mu_2 = 3$, $\sigma_2 = 2$.

Оптимальная (или байесовская) решающая функция будет выглядеть следующим образом

$$f(x) = \begin{cases} 1, & \varphi(x,1) \geq \varphi(x,2) \\ 2, & \varphi(x,1) < \varphi(x,2) \end{cases}.$$

Заметим, что при $\varphi(x,1) = \varphi(x,2)$ значение байесовской решающей функции может выбираться произвольно, как 1 так и 2,

причём практической реализации часто имеет смысл делать случайный выбор между этими значениями.

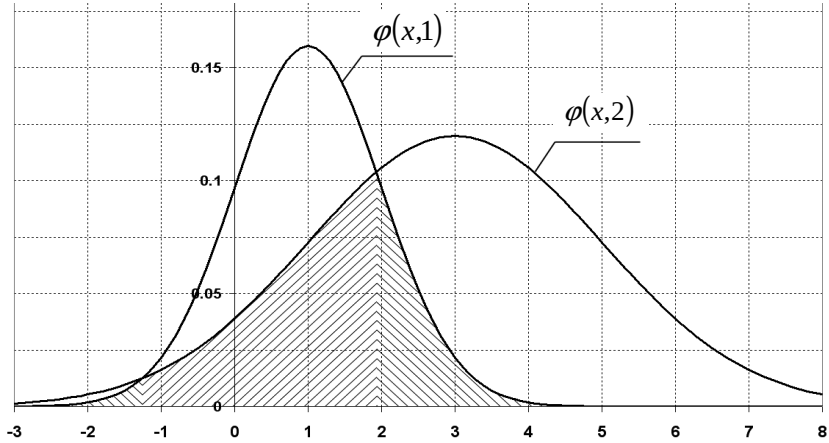

Рис. 3. Совместная плотность вероятности

На рис. 3 компоненты совместной плотности имеют две точки пересечения: при $x = a \approx -1{,}26$ и при $x = b \approx 1{,}93$. Оптимальная решающая функция

$$f(x) = \begin{cases} 1, & a < x < b \\ 2, & (x \le a) \vee (x \ge b) \end{cases}.$$

Вероятность ошибочной классификации есть

$$R = P(f(x) \neq y) = \int\limits_{f(x)=2} \varphi(x,1)\,dx + \int\limits_{f(x)=1} \varphi(x,2)\,dx.$$

Для приведённого примера

$$R = \int\limits_{-\infty}^{a} \varphi(x,1)\,dx + \int\limits_{a}^{b} \varphi(x,2)\,dx + \int\limits_{b}^{+\infty} \varphi(x,1)\,dx =$$

$$= \tfrac{1}{2} P(1)\left(1 + \Phi\left(\frac{a-\mu_1}{\sigma_1}\right) + 1 - \Phi\left(\frac{b-\mu_1}{\sigma_1}\right)\right) +$$

$$+\frac{1}{2}P(2)\left(\Phi\left(\frac{b-\mu_2}{\sigma_2}\right)-\Phi\left(\frac{a-\mu_2}{\sigma_2}\right)\right)\approx 0{,}24$$

– значение есть площадь заштрихованной области на рис. 3.

Здесь интегралы от нормальной плотности выражены через так называемую функцию Лапласа

$$\Phi(z)=\frac{2}{\sqrt{2\pi}}\int_0^z e^{-\frac{t^2}{2}}\,dt\,,$$

значения которой могут быть найдены в справочных таблицах либо вычислены на компьютере.

В разделе 1.2.1 понятие байесовской решающей функции вводилось через условные вероятности, которые в данном случае выглядят (рис. 4) следующим образом:

$$g_1(x)=P(y=1/x)=\frac{\varphi(x,1)}{\varphi(x)}=\frac{\varphi(x,1)}{\varphi(x,1)+\varphi(x,2)},$$

$$g_2(x)=P(y=2/x)=1-g_1(x).$$

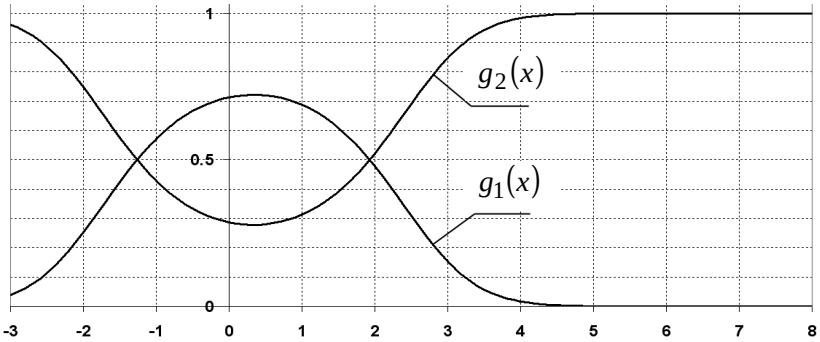

Рис. 4. Условные вероятности

Байесовская решающая функция может быть записана как

$f(x)=1$, при $g_1(x)\geq g_2(x)$; $f(x)=2$, при $g_1(x)<g_2(x)$.

Поскольку функции $g_1(x)$ и $g_2(x)$ в рассматриваемом случае положительны, их сравнение друг с другом эквивалентно сравнению их отношения с единицей, что, в свою очередь, эквивалентно сравнению логарифма отношения с нулём, поэтому байесовскую решающую функцию можно представить в виде

$$f(x)=\begin{cases}1, & l(x)\geq 0\\ 2, & l(x)<0\end{cases},$$

где

$$l(x)=\ln\frac{g_1(x)}{g_2(x)}=\ln\varphi_1(x)-\ln\varphi_2(x)+\ln\frac{P(1)}{P(2)}.$$

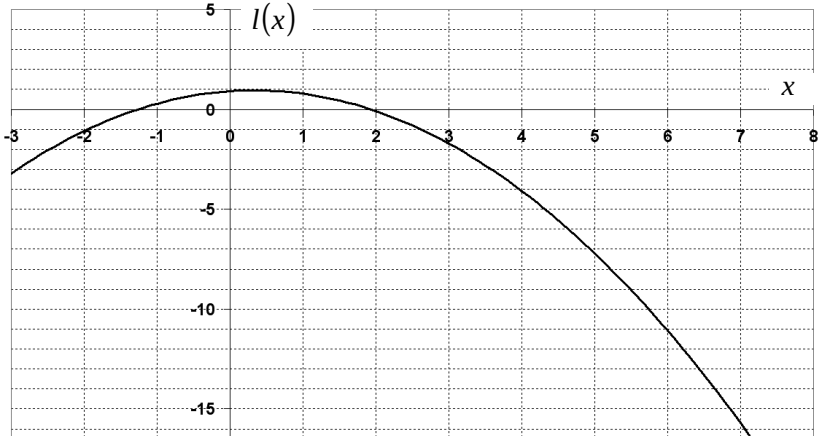

Рис. 5. Разделяющая функция

Функцию $l(x)$ будем называть разделяющей функцией. Для рассматриваемого примера её график приведён на рис. 5.

Подставляя нормальные плотности, получаем

$$l(x)=\frac{(x-\mu_2)^2}{(\sigma_2)^2}-\frac{(x-\mu_1)^2}{(\sigma_1)^2}+\ln\frac{\sigma_2}{\sigma_1}+\ln\frac{P(1)}{P(2)}.$$

Видим, что $l(x)=0$ представляет собой квадратное уравнение, которое имеет либо два корня, либо один корень — в случае $\sigma_1=\sigma_2$, либо не имеет корней — если одна из компонент совместной плотности целиком лежит под другой.

Рассмотрим теперь случай нескольких переменных $X_1,...,X_n$.

Условные плотности вероятности предполагаются нормальными и задаются параметрами μ_y и λ_y, где μ_y — вектор средних (математических ожиданий) для класса y, а λ_y — ковариационная матрица.

Оптимальная решающая функция записывается, так же как и в одномерном случае, через разделяющую функцию

$$l(x)=\ln\varphi_1(x)-\ln\varphi_2(x)+\ln\frac{P(1)}{P(2)}.$$

Решение уравнения $l(x)=0$ задает границу между классами и называется разделяющей поверхностью.

Подставим в разделяющую функцию аналитические выражения для нормальных плотностей

$$\varphi_k=\frac{1}{(2\pi)^{n/2}|\lambda_k|^{n/2}}e^{-\frac{1}{2}Q_k(x)},$$

где $Q_k(x)=(x-\mu_k)'(\lambda_k)^{-1}(x-\mu_k)$.

Штрих в записи квадратичной формы обозначает транспонирование вектора. В дальнейшем будем его опускать, считая левый вектор в произведении всегда транспонированным.

После преобразований имеем

$$2l(x)=Q_2(x)-Q_1(x)+\ln|\lambda_2|-\ln|\lambda_1|+2\ln P_1-2\ln P_2.$$

Далее преобразуем

$$\begin{aligned}Q_2(x)-Q_1(x)&=(x-\mu_2)(\lambda_2)^{-1}(x-\mu_2)-(x-\mu_1)(\lambda_1)^{-1}(x-\mu_1)=\\&=x\left((\lambda_2)^{-1}-(\lambda_1)^{-1}\right)x-\left(\mu_2(\lambda_2)^{-1}-\mu_1(\lambda_1)^{-1}\right)x-\\&x\left((\lambda_2)^{-1}\mu_2-(\lambda_1)^{-1}\mu_1\right)+\mu_2(\lambda_2)^{-1}\mu_2-\mu_1(\lambda_1)^{-1}\mu_1.\end{aligned}$$

Наконец получаем

$$2l(x) = xAx + bx + c\,, \text{ где}$$

$$A = (\lambda_2)^{-1} - (\lambda_1)^{-1}, \qquad b = 2\mu_1 (\lambda_1)^{-1} - 2\mu_2 (\lambda_2)^{-1},$$

$$c = \mu_2 (\lambda_2)^{-1}\mu_2 - \mu_1 (\lambda_1)^{-1}\mu_1 + \ln|\lambda_2| - \ln|\lambda_1| + 2\ln P_1 - 2\ln P_2\ .$$

При получении выражения для b использована коммутативность скалярного произведения векторов и коммутативность произведения вектора на симметричную матрицу (напомним, что ковариационная матрица симметрична).

Уравнение $l(x) = 0$ в общем случае задает поверхность второго порядка, которая при $\lambda_1 = \lambda_2$, то есть при равенстве ковариационных матриц, вырождается в гиперплоскость.

Заметим, что ковариационные матрицы являются положительно определенными, ввиду чего линии (поверхности) уровня нормального распределения представляют собой эллипсоиды. При этом разделяющая поверхность может оказаться любой поверхностью второго порядка: эллипсоидом, гиперболоидом, параболоидом, конусом, цилиндром.

1.3.2. Оценивание параметров

Пусть имеется выборка, представленная прямоугольной матрицей (x_{ij}, y_i), $j = \overline{1,n}$, $i = \overline{1,N}$. Параметр N представляет собой число независимых реализаций системы случайных величин и называется объемом выборки.

Сформируем множество индексов (номеров) объектов (строк матрицы), принадлежащих первому классу: $I_1 = \{i \mid y_i = 1\}$. Аналогично, $I_2 = \{i \mid y_i = 2\}$. Тогда $N_1 = |I_1|$ есть число объектов первого, а $N_2 = |I_2|$ — второго классов в выборке, $N_1 + N_2 = N$.

Проще всего оценить априорные вероятности классов, а именно: $\tilde{P}_k = \dfrac{N_k}{N}$, где $\tilde{P}_k$ – оценка для P_k, а k – номер класса (1 или 2).

Обозначим через $\tilde{\mu}^k = \left(\tilde{\mu}_1^k,...,\tilde{\mu}_n^k\right)$ оценку вектора μ_k. В этом обозначении k рядом с $\tilde{\mu}$ означает не степень, а верхний индекс. За счет переноса номера класса в верхний индекс мы освободили нижний индекс для обозначения компонент вектора.

Компоненты вектора средних для каждого класса оцениваются по формуле

$$\tilde{\mu}_j^k = \frac{1}{N_k} \sum_{i \in I_k} x_{ij} \ .$$

Оценку ковариационной матрицы λ_k обозначим через $\tilde{\lambda}^k = \left(\tilde{\lambda}_{jl}^k\right),\ j = \overline{1,n}\ ,\ l = \overline{1,n}$.

Компоненты этой матрицы вычисляются как

$$\tilde{\lambda}_{jl}^k = \frac{1}{N_k} \sum_{i \in I_k} \left(x_{ij} - \tilde{\mu}_j^k\right)\left(x_{il} - \tilde{\mu}_l^k\right) .$$

Для построения выборочной разделяющей поверхности достаточно теперь в уравнение $l(x) = 0$ подставить вместо параметров распределений найденные оценки.

Случай равных матриц ковариации.

Если известно, что распределения обоих классов имеют одинаковые матрицы ковариации, то нужно всю выборку использовать для вычисления оценки $\tilde{\lambda} = \left(\tilde{\lambda}_{jl}\right)$ матрицы $\lambda = \lambda_1 = \lambda_2$. Оценка находится следующим образом

$$\tilde{\lambda}_{jl} = \frac{1}{N} \left(\sum_{i \in I_1} \left(x_{ij} - \tilde{\mu}_j^1\right)\left(x_{il} - \tilde{\mu}_l^1\right) + \sum_{i \in I_2} \left(x_{ij} - \tilde{\mu}_j^2\right)\left(x_{il} - \tilde{\mu}_l^2\right) \right) .$$

Остальные параметры оцениваются так же, как в общем случае.

В качестве примера рассмотрим задачу об ирисах, изложенную в разделе 1.1.1.

Поскольку мы разобрали метод применительно к случаю двух классов, выберем для классификации два вида ирисов:

1 – Iris Versicolour,

2 – Iris Virginica.

Данные возьмём из таблицы 1, где нас будут интересовать строки с $y=2$ и $y=3$, при этом значения целевой переменной мы переобозначим (2 заменим на 1, а 3 на 2).

Находим оценки безусловных вероятностей классов:

$$\tilde{P}_1=\frac{N_1}{N}=\frac{12}{24}=0{,}5,\ \tilde{P}_2=\frac{N_2}{N}=\frac{12}{24}=0{,}5.$$

Далее вычисляем оценки векторов средних:

$$\tilde{\mu}^1=(6{,}0;4{,}34),\ \tilde{\mu}^2=(6{,}8;5{,}64).$$

Оценки ковариационных матриц:

$$\tilde{\lambda}^1=\begin{pmatrix}0{,}098 & 0{,}058\\ 0{,}058 & 0{,}156\end{pmatrix},\ \tilde{\lambda}^2=\begin{pmatrix}0{,}385 & 0{,}257\\ 0{,}257 & 0{,}226\end{pmatrix}.$$

Параметры разделяющей кривой:

$$A=\begin{pmatrix}-2{,}24 & -7{,}41\\ -7{,}41 & 10{,}11\end{pmatrix},\ b=(105{,}77;-26{,}92),\ c=-228{,}64.$$

Уравнение кривой:

$$-2{,}24x_1^2+10{,}11x_2^2-14{,}82x_1x_2+105{,}77x_1-26{,}92x_2-228{,}64=0.$$

Полученная кривая является гиперболой, график которой приведён на левой диаграмме рис. 6.

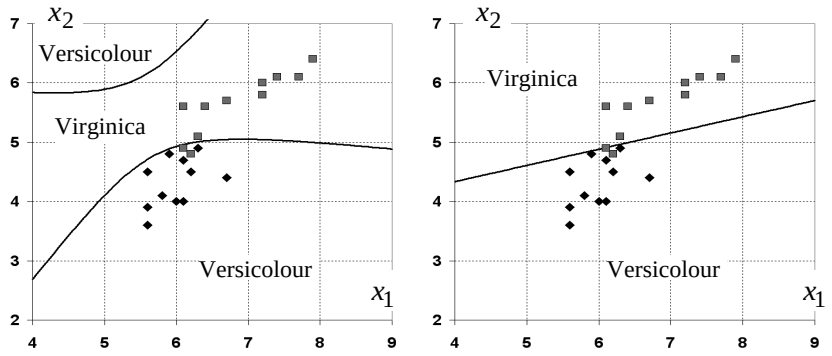

Рис. 6. Выборочные разделяющие кривые

Если предположить равенство ковариационных матриц, то оценка матрицы будет

$$\tilde{\lambda}^1 = \begin{pmatrix} 0{,}24 & 0{,}16 \\ 0{,}16 & 0{,}19 \end{pmatrix}.$$

Параметры разделяющей прямой:

$$b = (4{,}82; -17{,}6),\ c = 57{,}0.$$

Уравнение прямой:

$$4{,}82x_1 - 17{,}6x_2 + 57 = 0.$$

График прямой приведён на правой диаграмме рис. 6.

На практике обычно нет информации о том, являются ли ковариационные матрицы классов одинаковыми, поэтому решение о способе их оценивания приходится принимать эмпирически. Можно, например, применить статистический критерий проверки гипотезы о равенстве ковариационных матриц. Другой вариант: при большом объеме выборки оценивать матрицы для каждого класса отдельно, при малых выборках — оценивать одну матрицу и строить линейную разделяющую поверхность. Здесь мы опять сталкиваемся с эффектом, что при малых выборках следует строить более простое решающее правило, даже если это упрощение вносит заведомую погрешность.

§ 1.4. Деревья решений

Решающие деревья являются одним из наиболее часто используемых инструментов для анализа данных. К достоинствам решающих деревьев относятся их наглядность, применимость к широкому кругу задач, возможность варьировать сложность решений, подстраиваясь под различный объём выборки, способность автоматически выбирать информативные переменные.

1.4.1. Задача классификации

Методы, основанные на деревьях решений, применяются почти во всех задачах анализа данных. В первую очередь, рассмотрим задачу классификации.

В таблице 7 приведена модельная (т. е. созданная искусственно, а не взятая из реальной задачи) выборка, представленная 24-я точками двух классов в двумерном пространстве. Обе переменные количественные и область значений каждой представляет собой интервал [0, 1].

Эта выборка изображена на рис. 7, где треугольные маркеры соответствуют первому, круглые — второму классу.

Таблица 7. Обучающая выборка

i	x_1^i	x_2^i	y^i	i	x_1^i	x_2^i	y^i
1	0,2	0,78	2	13	0,94	0,59	2
2	0,9	0,92	2	14	0,15	0,43	2
3	0,57	0,33	1	15	0,3	0,1	1
4	0,92	0,89	2	16	0,19	0,08	1
5	0,29	0,66	2	17	0,48	0,56	2
6	0,28	0,16	2	18	0,37	0,65	1
7	0,52	0,14	2	19	0,86	0,3	2
8	0,43	0,02	2	20	0,72	0,71	1
9	0,92	0,18	1	21	0,81	0,2	1
10	0,62	0,69	1	22	0,68	0,32	1
11	0,95	0,8	1	23	0,79	0,59	1
12	0,7	0,96	2	24	0,75	0,59	1

Идея метода, основанного на решающих деревьях, заключается в последовательном разбиении пространства значений переменных на области $E_1,...,E_L$ и приписывании каждой полученной области решения в виде номера класса.

Под последовательным разбиением понимается то, что сначала всё пространство $X = X_1 \times X_2$ разбивается на две области $E_1^{(1)}$ и $E_2^{(1)}$, из которых затем выбирается область, которая, в свою очередь, разбивается на две, в результате чего получается три об-

ласти $E_1^{(2)}$, $E_2^{(2)}$, $E_3^{(2)}$, и так далее, пока через $L-1$ таких шагов не получится L областей.

Заметим, что способов подобного разбиения пространства X на L областей очень много. Естественно выбрать разбиение, минимизирующее число ошибочно классифицируемых объектов выборки. Однако число различимых на выборке объёма N разбиений в случае n количественных переменных составляет порядка $(nN)^L$, поэтому полный перебор всех разбиений в реальных задачах обычно невозможен.

При этом задача поиска разбиения, точно классифицирующего обучающую выборку и имеющего минимальное число областей, является NP-полной. Это значит, что не следует рассчитывать на появление имеющих низкую трудоёмкость точных алгоритмов нахождения наилучшего разбиения. В этой ситуации используют эвристические алгоритмы, простейшим вариантом которых является так называемый «жадный» алгоритм.

Таблица 8. Варианты разбиения по переменной X_1.

s	$N^1(s)$	$N^2(s)$	$\tilde{N}(s)$	s	$N^1(s)$	$N^2(s)$	$\tilde{N}(s)$
0,17	0	1	11	0,69	6	7	11
0,195	1	1	12	0,71	6	8	10
0,24	1	2	11	0,735	7	8	11
0,285	1	3	10	0,77	8	8	12
0,295	1	4	9	0,8	9	8	11
0,335	2	4	10	0,835	10	8	10
0,4	3	4	11	0,88	10	9	11
0,455	3	5	10	0,91	10	10	12
0,5	3	6	9	0,92	11	10	11
0,545	3	7	8	0,93	11	11	12
0,595	4	7	9	0,945	11	12	11
0,65	5	7	10	—	12	12	12

Рассмотрим работу «жадного» алгоритма на примере построения разбиения выборки из таблицы 7.

На первом шаге алгоритм перебирает все варианты разбиения пространства X на две области. Разбиение можно проводить как по переменной X_1, так и по X_2.

Переберём сначала варианты разбиения по переменной X_1.

Хотя число вариантов разбиения области значений количественной переменной на два интервала бесконечно (континуально), нас интересуют только варианты, различимые на обучающей выборке, поэтому будем рассматривать только границы, расположенные посредине между проекциями выборочных точек. Все такие варианты границ s по переменной X_1 приведены в таблице 8.

Каждая граница s разбивает пространство X на области $E_1^{(1)} = [0, s) \times [0,1]$ и $E_2^{(1)} = [s,1] \times [0,1]$.

Обозначим за $N^1(s)$ число точек первого, а $N^2(s)$ – число точек второго класса в области $E_1^{(1)}$.

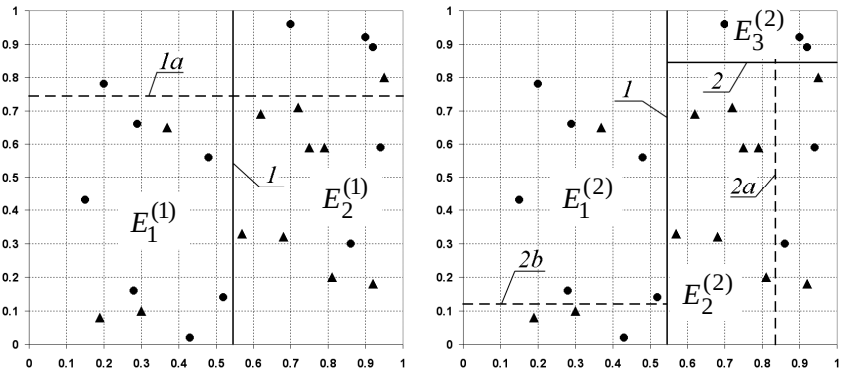

Рис. 7. Построение разбиения деревом решений

Естественно, что в каждой получившейся области решающая функция припишет тот класс, выборочных объектов которого в

этой области больше. Тогда оставшиеся объекты будут классифицированы ошибочно. Таким образом, число ошибочно классифицированных объектов составит

$$\tilde{N}(s) = \min\left(N^1(s), N^2(s)\right) + \min\left(N - N^1(s), N - N^2(s)\right).$$

В таблице 8 приведены все варианты разбиения пространства по переменной X_1. Видим, что наименьшее число ошибок (восемь) получается при $s = 0{,}545$. Соответствующая граница показана на рис. 7 цифрой *1*.

Аналогичным образом следует перебрать все варианты разбиения по переменной X_2. Наименьшее число ошибок (девять) будет достигнуто для границы, обозначенной на рис. 7 как *1а*.

Поскольку разбиение по переменной X_1 даёт меньшее число ошибок, окончательно выбираем разбиение с границей *1*, т.е. $E_1^{(1)} = [0;\ 0{,}545) \times [0;\ 1]$, $E_2^{(1)} = [0{,}545;\ 1] \times [0;\ 1]$.

На втором шаге алгоритма аналогичным образом перебираем варианты разбиений областей $E_1^{(1)}$ и $E_2^{(1)}$.

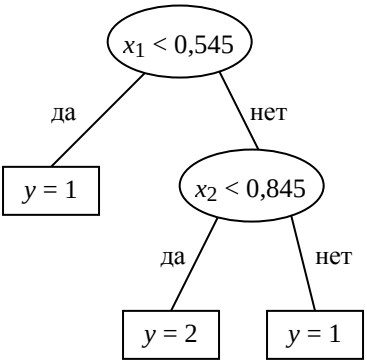

Рис. 8. Дерево решений

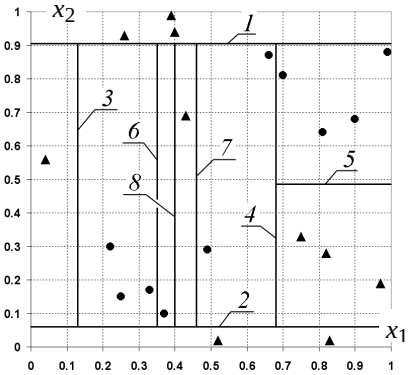

Рис. 9. Разбиение для задачи «исключающего или»

В результате перебора находим (см. рис. 7), что граница *2b* уменьшает общее число ошибок на 1, граница *2a* уменьшает общее число ошибок на 2, а граница *2* уменьшает общее число ошибок на 3. В результате выбираем вариант *2* и получаем разбиение на области

$$E_1^{(2)} = [0;\ 0{,}545) \times [0;\ 1],$$

$$E_2^{(2)} = [0{,}545;\ 1] \times [0;\ 0{,}845),\ E_3^{(2)} = [0{,}545;\ 1] \times [0{,}845;\ 1].$$

Решающая функция $f(x)$ принимает значение 1 при $x \in E_1^{(2)} \cup E_3^{(2)}$ и значение 2 при $x \in E_2^{(2)}$.

На рис. 8 эта же решающая функция представлена в форме дерева решений.

```
[x1 < 0.905] : (20 : 10)
├─[x1 < 0.06] : (17 : 7)
│  ├─(2 : 0)
│  └─[x0 < 0.13] : (15 : 5)
│     ├─(1 : 0)
│     └─[x0 < 0.68] : (14 : 4)
│        ├─[x0 < 0.35] : (7 : 1)
│        │  ├─(3 : 0)
│        │  └─[x0 < 0.455] : (4 : 1)
│        │     ├─[x0 < 0.405] : (2 : 1)
│        │     │  ├─(1 : 0)
│        │     │  └─(1 : 0)
│        │     └─(2 : 0)
│        └─[x1 < 0.485] : (7 : 3)
│           ├─(3 : 0)
│           └─(4 : 0)
└─(3 : 0)
```

Рис. 10. Дерево решений в текстовой форме

На рис. 9 приведён пример другой выборки. Легко заметить, что данная выборка безошибочно классифицируется при разбиении

$$E_1 = [0;\ 0{,}5) \times [0;\ 0{,}5), \quad E_2 = [0{,}5;\ 1] \times [0;\ 0{,}5),$$
$$E_3 = [0;\ 0{,}5) \times [0{,}5;\ 1], \quad E_4 = [0{,}5;\ 1] \times [0{,}5;\ 1].$$

При этом «жадный» алгоритм достигает безошибочной классификации только при разбиении на 9 областей (показаны на рис. 9, цифрами отражён порядок проведения границ). Заметим, что некоторые проведённые границы вообще не уменьшают число ошибок и проводились так, чтобы разделять область с наибольшим числом объектов на подобласти, разделяя объекты примерно поровну.

На рис. 10 приведено это же разбиение в виде дерева решений, в круглых скобках указано общее число объектов в соответствующей области и число ошибок.

Последний пример демонстрирует, что «жадный» алгоритм в общем случае находит далеко не лучшее решение. Существуют многочисленные усовершенствованные варианты алгоритма направленного поиска дерева решений, однако до настоящего времени проблема построения наиболее эффективного алгоритма остаётся открытой.

Одно из важнейших достоинств решающих деревьев заключается в том, что основанные на них методы построения решающих функций очень легко переносятся на случай разнотипного пространства переменных.

Если для вещественных переменных в качестве областей разбиений берутся интервалы, то для номинальных переменных в разбиениях участвуют любые подмножества значений.

1.4.2. Задача восстановления зависимостей.

В случае количественной целевой переменной дерево решений даёт кусочно-постоянную решающую функцию.

Для иллюстрации метода построения решающего дерева в задаче восстановления зависимостей будем использовать данные из таблицы 2 по задаче оценивания качества вин. При этом ограни-

чимся одной переменной X_1, которую для удобства переобозначим как X.

Точечная диаграмма для таблицы исходных данных приведена на рис. 11.

Требуется поставить $L-1$ границ так, чтобы на получившихся L областях кусочно-постоянная оценка зависимости y от x была бы наиболее близка выборочным данным.

Близость функции $f(x)$ выборке будем характеризовать следующей мерой отличия

$$D=\sum_{i=1}^{N}\left(y^i-f\left(x^i\right)\right)^2,$$

которая представляет собой сумму квадратов отклонений выборочных значений целевой переменной от их оценок решающей функцией.

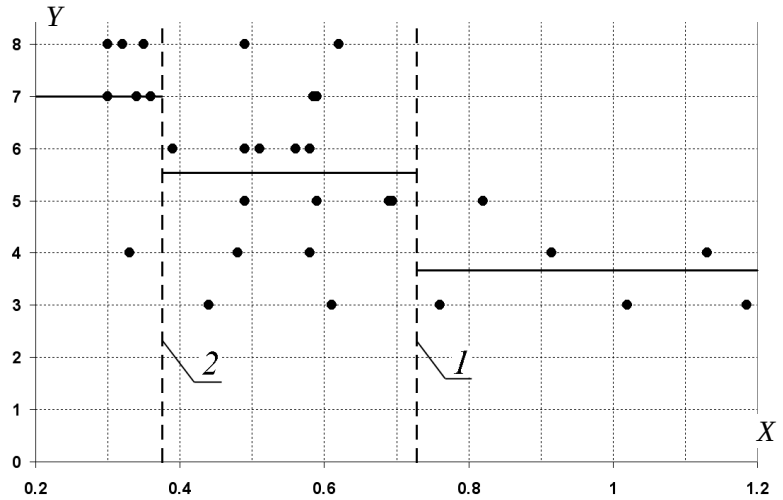

Рис. 11. Кусочно-постоянная решающая функция

Наилучшей константной оценкой будет выборочное среднее

$$\tilde{y}_0 = \frac{1}{N}\sum_{i=1}^{N} y^i \ .$$

Погрешность такой оценки есть

$$D_0 = \sum_{i=1}^{N} \left(y^i - \tilde{y}_0\right)^2 \ .$$

Для рассматриваемой задачи $\tilde{y}_0 = 5{,}5$, $D_0 = 87{,}5$.

Расставлять границы будем «жадным» алгоритмом аналогично тому, как это делалось для задачи классификации.

Первая граница разбивает область значений переменной $X = [0{,}2;\ 1{,}2]$ на два интервала: $E_1^{(1)} = [0{,}2;\ s)$ и $E_2^{(1)} = [s;\ 1{,}2]$.

На первом интервале выборочное среднее и отклонение будут

$$\tilde{y}_1(s) = \frac{1}{N(s)} \sum_{x^i < s} y^i \ , \quad D_1(s) = \sum_{x^i < s} \left(y^i - \tilde{y}_1(s)\right)^2 ,$$

где $N(s)$ – количество точек выборки, расположенных левее s.

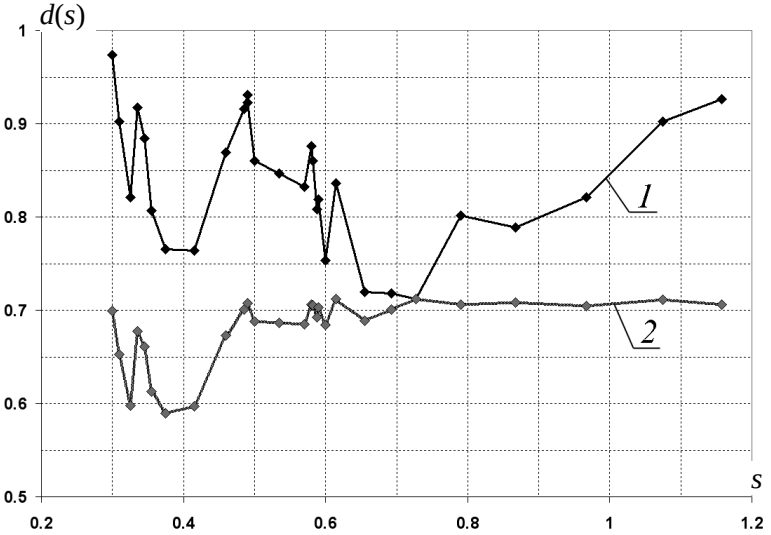

Рис. 12. Остаточное отклонение при различных разбиениях

Аналогично для второго интервала

$$\tilde{y}_2(s)=\frac{1}{N-N(s)}\sum_{x^i\ge s} y^i\,,\quad D_1(s)=\sum_{x^i\ge s}\left(y^i-\tilde{y}_1(s)\right)^2.$$

Суммарное отклонение есть

$$D(s)=D_1(s)+D_2(s).$$

В таблице 9 приведены величины суммарных отклонений для различных положений границы *s*.

На рис. 12 кривая *1* показывает зависимость отношения $d(s)=\frac{D(s)}{D_0}$ от *s*.

Минимальное значение 62,3 величины $d(s)$ достигается при $s=0{,}728$, поэтому первую границу зафиксируем с этой точке (см. рис 11, линия *1*).

Таблица 9. Варианты разбиения по переменной *X*.

s	$\tilde{y}_1(s)$	$\tilde{y}_2(s)$	*D*(*s*)	*s*	$\tilde{y}_1(s)$	$\tilde{y}_2(s)$	*D*(*s*)
0,3	7	5,45	85,2	0,58	6,06	4,86	76,7
0,31	7,5	5,36	78,9	0,583	6,06	4,77	75,2
0,325	7,67	5,26	71,9	0,588	6,11	4,58	70,7
0,335	6,75	5,31	80,3	0,59	6,05	4,55	71,7
0,345	6,8	5,24	77,4	0,6	6,1	4,3	65,9
0,355	7	5,13	70,6	0,615	5,95	4,44	73,2
0,375	7	5,04	67	0,655	6,05	4	63
0,415	6,88	5	66,9	0,693	6	3,86	62,9
0,46	6,44	5,1	76	0,728	5,96	3,67	62,3
0,485	6,2	5,15	80,1	0,79	5,84	3,8	70,2
0,49	6,09	5,16	81,4	0,868	5,81	3,5	69
0,5	6,23	4,94	75,2	0,968	5,74	3,33	71,9
0,535	6,21	4,88	74,1	1,075	5,64	3,5	78,9
0,57	6,2	4,8	72,8	1,158	5,59	3	81

Далее аналогичным образом перебираем варианты постановки второй границы, получая три области разбиения, и вычисляем средние и отклонения на каждой области. Зависимость относительного суммарного (по трём областям) отклонения от положения второй границы показана кривой *2* на рис. 12. Видим, что минимум достигается при $s = 0{,}375$, чему соответствует линия *2* на рис. 11.

Аналогичным образом можно ставить последующие границы, разбивая *X* на заданное число областей. Однако в данном примере постановка последующих границ не даёт существенного уменьшения $d(s)$, поэтому имеет смысл ограничиться тремя областями. Полученная кусочно-постоянная решающая функция $f(x)$ показана сплошной линией на рис. 11.

Заметим, что в рассмотренном примере можно было строить и линейную регрессию. Этот пример слишком простой, чтобы продемонстрировать преимущества деревьев решений, которые проявляются при большой размерности пространства, разнотипности, немонотонной зависимости целевой переменной. Однако более сложные примеры были бы менее наглядны.

§ 1.5. Прогнозирование бинарного временного ряда

Задачу прогнозирования временного ряда проиллюстрируем для случая одной бинарной (двоичной) переменной. Для её решения будем использовать одну из модификаций метода прецедентов, сводящуюся в данном случае к оцениванию частот различных предысторий.

В качестве исходных данных возьмём последовательность, сформированную вручную из 0 и 1, выбиравшихся наугад.

На первый взгляд, может показаться, что вероятности появления 0 и 1 не зависят от ранее выбранных значений и равны 0,5. Если бы это было так, то такую последовательность не имело бы смысла прогнозировать, поскольку для любого решающего правила вероятность ошибки была бы 0,5. Однако человек, вообще говоря, не является идеальным генератором случайных чисел и, даже стараясь выбирать числа равновероятно и «бессистемно»,

он составляет ряд, в котором присутствуют статистические закономерности.

Исходная длина последовательности, представленной в таблице 10, равна 150. Для того, чтобы иметь возможность проверить качество прогноза, разобьём ряд на две части: обучающую, длиной 100, и контрольную, длиной 50.

Таблица 10. Временные ряды

обучение	**011010001010011011101011000010111001111010101101010000101101111101010111101010101001011101010110101010**
контроль	**11101010111100110000010110101001011100101010101011**
прогноз	**10111010101101111110110101101011110111111010101010**

Выберем значения параметра d – длины предыстории, по которой будет производиться прогноз. При используемом методе прогнозирования d должно быть таким, чтобы 2^d – число возможных предысторий было существенно меньше длины обучающей последовательности. Возьмём $d = 4$.

Далее перебираем все возможные последовательности из пяти 0 и 1 и подсчитываем, сколько раз они встречаются в обучающем ряде и результат вносим в таблицу 11. Например, последовательность 00000 не встретилась ни разу, а фрагмент 00001 присутствует два раза, поэтому в первой строке таблицы $N(x,0) = 0$, а $N(x,1) = 2$. Следующая строка отражает частоту появления фрагментов 00010 и 00011 и так далее.

Поскольку после 0000 значение 1 встречается чаще, чем 0, решающая функция $f(x)$ в точке $x = (0,0,0,0)$ равна 1. После 0001 значение 1 встречается, наоборот, реже 0, поэтому $f(0,0,0,1) = 0$.

Таблица 11. Решающая функция

X_{-4}	X_{-3}	X_{-2}	X_{-1}	$N(x,0)$	$N(x,1)$	$N_{\min}(x)$	$f(x)$
0	0	0	0	0	2	0	1
0	0	0	1	3	0	0	0
0	0	1	0	0	4	0	1
0	0	1	1	1	1	1	1
0	1	0	0	2	2	2	1
0	1	0	1	9	7	7	0
0	1	1	0	1	5	1	1
0	1	1	1	3	3	3	1
1	0	0	0	2	1	1	0
1	0	0	1	1	2	1	1
1	0	1	0	4	12	4	1
1	0	1	1	4	5	4	1
1	1	0	0	1	1	1	1
1	1	0	1	8	2	2	0
1	1	1	0	1	5	1	1
1	1	1	1	3	1	1	0

После фрагмента 0011 как 0 так и 1 встретились по разу, поэтому у нас нет оснований для выбора 0 или 1 в качестве прогноза. В подобных случаях решение можно принимать произвольно. Для определённости при равенстве частот будем приписывать 1.

Столбец $N_{\min}(x) = \min(N(x,0), N(x,1))$ приведён для удобства вычисления доли ошибок на обучающей последовательности

$$\tilde{R} = \frac{1}{100-4} \sum_{x} N_{\min}(x) = \frac{29}{96} \approx 0{,}3 .$$

Подсчитаем теперь число ошибок на контрольной последовательности.

Обучающая последовательность (см. таблицу 10) заканчивается на 1010. Для данного фрагмента в таблице 11 находим $f(x) = 1$, поэтому первое прогнозируемое значение будет 1.

Для прогноза следующего значения добавим один элемент контрольной последовательности к обучающей. Теперь последняя предыстория есть 0101. Для неё $f(x)=0$, поэтому следующее прогнозируемое значение 0.

Далее добавляем ещё один элемент контрольной последовательности к обучающей. Теперь последняя предыстория есть 1011. Для неё $f(x)=1$.

Повторяя указанные действия 50 раз получаем последовательность из 50 прогнозированных значений. Сравнив прогноз с контрольной последовательностью, обнаруживаем 17 несовпадений, т.е. доля ошибок прогноза есть $\frac{17}{50}=0{,}34$.

Получившаяся доля ошибок существенно отличается от 0,5. Применив классические критерии проверки гипотез, легко показать, что это отличие статистически значимо и подобный ряд действительно можно прогнозировать с вероятностью ошибки, меньшей 0,5.

Заметим, что в исходном ряде 0 и 1 встречаются с разной частотой. Так в обучающей последовательности 0 встречается 45 раз, а 1 встречается 55 раз. Таким образом, можно было, даже не анализируя предысторию, всегда прогнозировать 1 и ошибаться реже, чем в половине случаев. При этом частота ошибок на контроле составила бы 23, что существенно больше 17 ошибок, сделанных изложенным выше методом.

§ 1.6. Кластерный анализ

Задачи кластерного анализа состоят в том, чтобы разбить заданную совокупность объектов на группы так, чтобы внутри каждой группы объекты были в некотором смысле похожими, а объекты разных групп максимально отличались.

Синонимами кластерного анализа являются таксономия, автоматическая группировка, классификация «без учителя». Последний термин подразумевает, что в отличие от рассматривавшейся ранее задачи классификации («с учителем») в обучающей выборке отсутствует информация о принадлежности объектов классам (т.е. о значениях переменной Y).

Таким образом, обучающая выборка в задаче кластерного анализа имеет вид

$$\nu = \left(x^i \,\middle|\, i = 1,\ldots, N \right).$$

Иногда в значении кластерного анализа используют термин «автоматическая классификация», однако данный термин лучше не использовать ввиду неоднозначности, поскольку классификация «с учителем» имеет такие же основания называться автоматической, поскольку производится компьютерной программой.

Кластеризация обычно использует некоторую функцию сходства объектов, например, расстояние. Некоторые методы ограничиваются понятием соседства объектов в топологическом смысле, дополнительно используя некоторую меру.

1.6.1. Выделение кластеров

Первый подкласс методов — методы «нисходящей» кластеризации. В этих методах выбирается количество кластеров производится разбиение множества объектов на кластеры (группы объектов) в соответствии с некоторым критерием.

Метод k-средних

Метод заключается в разбиении объектов на группы так, чтобы сумма расстояний от объектов до центров групп была минимальной.

Пусть $Y=\{1,...,k\}$ – множество номеров кластеров. Под решающей функцией будем понимать отображение $f: X \to Y$, которое сопоставляет каждой точке $x \in X$ номер кластера $y \in Y$.

Обозначим через $\tilde{x}(y)=\frac{1}{N(y)}\sum\limits_{x \in \nu_y} x$ – центр (среднюю точку) кластера y. Здесь ν_y – множество точек выборки, отнесённых к кластеру y, т.е. таких, что $f(x)=y$, а $N(y)=|\nu_y|$ – число таких точек.

Критерием качества кластеризации будет величина

$$\tilde{K}(f)=\frac{1}{N}\sum_{i=1}^{N}\Delta\left(x^i, \tilde{x}\left(f\left(x^i\right)\right)\right),$$

где $\Delta(x', x'')$ – расстояние между точками x' и x''.

Чем меньше значение $\tilde{K}(f)$, чем лучше кластеризация $f(x)$.

Заметим, что в статистической постановке основным является критерий качества решений на распределениях, который может иметь вид

$$K(f)=\int_X \Delta(x, \tilde{x}(f(x)))\, dP[X],$$

а критерий $\tilde{K}(f)$ играет роль выборочной оценки для $K(f)$. Здесь $dP[X]$ означает интегрирование по вероятностной мере. Читатель, не знакомый с интегралом Лебега, может без ущерба для понимания материала заменять $dP[X]$ на $\varphi(x)dx$, где $\varphi(x)$ – плотность вероятности. При этом следует иметь в виду, что исходное выражение имеет смысл для любых вероятностных мер (в частности, для дискретных и непрерывных распределений), а

плотность вероятности определена только для (абсолютно) непрерывных случайных величин.

Рассмотрим пример использования алгоритма *k*-средних.

Таблица 12. Выборка для кластеризации

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_1^i	2,1	2,9	1,8	5,1	7,2	1,1	6,4	0,9	0,3	0,8	4,8	4,7	5,5	5,2	6,2
x_2^i	2,1	0,5	2,9	5,7	6,4	3,2	5,4	1,4	1,9	0,3	6,3	4,2	7,8	4,1	7,3

В таблице 12 приведена выборка, которую требуется разбить на два кластера.

Заметим, что на практике число кластеров *k*, как правило, заранее не известно. В этом случае обычно перебирают различные *k* и останавливаются, когда при дальнейшем увеличении числа кластеров улучшение критерия качества замедляется.

Выборку объёма *N* можно разбить на два кластера 2^{N-1} способами. При небольших N все эти способы можно перебрать и выбрать тот, на котором достигается минимум критерия. В остальных случаях потребуется более интеллектуальный алгоритм оптимизации, который может быть найден в литературе.

Таблица 13. Варианты кластеризаций

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
y_a^i	1	1	1	2	2	1	2	1	1	1	2	2	2	2	2
y_b^i	2	2	1	1	2	1	2	1	1	1	1	2	2	2	2

В таблице 13 приведены два варианта кластеризации: *a* и *b*, которые наглядно изображены на рис. 13. Значение критерия качества есть сумма длин отрезков, соединяющих точки с центром соответствующего кластера. Точки первого кластера обозначены

треугольными маркерами, второго — круглыми. Легко заметить, что кластеризация *a* (левый рисунок) существенно лучше *b*.

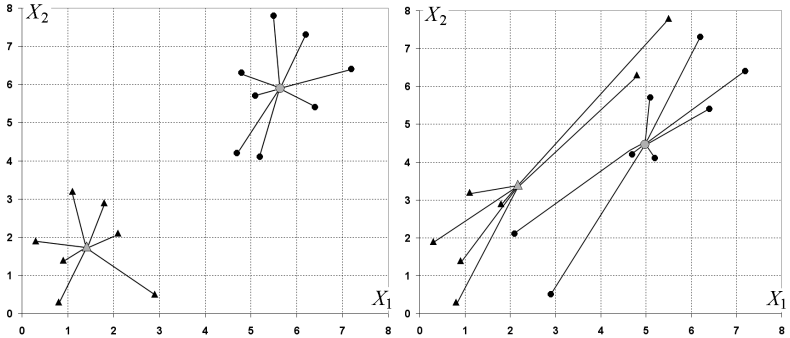

Рис. 13. Кластеризация методом *k* средних

Заметим, что в кластеризации *b* есть точки, расстояние от которых до центра «своего» кластера больше, чем до «чужого». Если эти точки перенести в другой кластер, сразу будет получена оптимальная кластеризация *a*. Такая процедура «уточнения» кластеров часто используется на практике при поиске оптимальной кластеризации.

Оценивание смеси распределений

Смесью распределений называется распределение, функция распределения которого представляется взвешенной суммой некоторых функций распределения заданного вида. Чаще смесь задаётся через плотности вероятности

$$\varphi(x) = \sum_{j=1}^{k} \alpha_j \varphi_j(x), \qquad \sum_{j=1}^{k} \alpha_j = 1,$$

где $\varphi_j(x)$ – плотности заданного вида, α_j – весовые коэффициенты.

Плотности $\varphi_j(x)$ обычно выбираются из некоторого параметрического семейства. В этом случае задача кластеризации становится стандартной статистической задачей оценивания парамет-

ров распределения. Для её решения существуют хорошо развитые методы.

В кластерном анализе чаще всего используются смеси нормальных распределений.

1.6.2. Иерархическая кластеризация.

Данный класс методов использует идею «восходящей» кластеризации, когда точки последовательно присоединяются к кластерам, а кластеры последовательно укрупняются. При этом последовательное объединение кластеров формирует дерево.

На рис. 14 приведён пример иерархической кластеризации для выборки из таблицы 13.

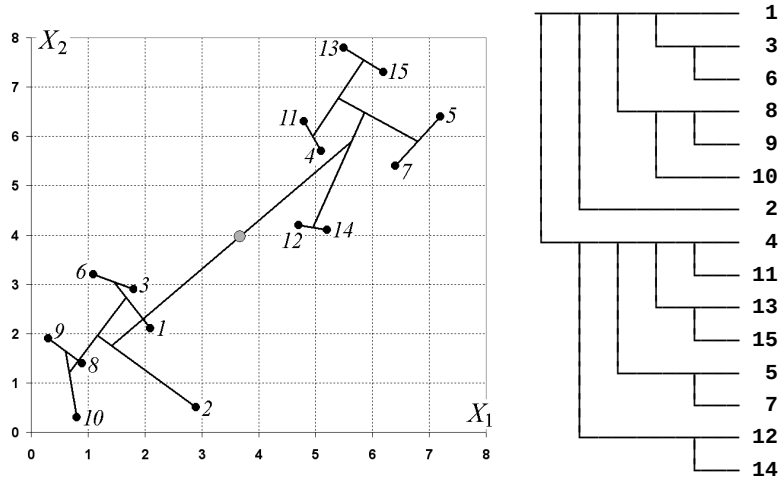

Рис. 14. Иерархическая кластеризация

Справа приведено дерево кластеризации. Два поддерева, выходящие из корневой вершины, в точности совпадают с кластерами, полученными методом *k*-средних. При этом дерево несёт гораздо больше информации о структуре расположения точек, чем обычная кластеризация.

Иерархическая кластеризация может использоваться и в том случае, когда известны только попарные расстояния между точками, а координаты точек неизвестны или вовсе не имеют смысла. Такая ситуация может возникнуть, например, при группировке музыкальных произведений. С помощью экспертов могут быть даны численные оценки похожести (или, наоборот, отличия) произведений. Тогда для каждого произведения будут известны «расстояния» от него до каждого из остальных произведений. При этом музыкальные произведения не задаются точками в пространстве переменных.

На первом шаге кластеризации в матрице попарных расстояний находится минимальное значение. Оно соответствует паре наиболее близких объектов. Их нужно объединить в один кластер, после чего требуется вычислить расстояния от этого кластера до всех оставшихся объектов.

Теперь можно заменить эту пару кластером и получить матрицу попарных расстояний, аналогичную исходной, но меньшего размера.

Далее опять ищем минимальный элемент матрицы и т.д. Продолжаем процесс, пока на верхнем уровне не будет получен один кластер.

Рассмотрим более подробно вычисление расстояний до кластера, получаемого объединением кластеров.

Пусть известны расстояния от заданного объекта *C* (в т.ч. кластера) до кластера *A*, состоящего из k_A точек, и до кластера *B*, состоящего из k_B точек, а также известно расстояние между *A* и *B*. Требуется определить расстояние от объекта *C* до кластера *D*, полученного объединением *A* и *B*.

На самом деле, определить искомое расстояние можно различными способами, получая разный результат.

Наиболее естественным представляется расположить точки *A*, *B*, *C*, *D* на плоскости, причём точку *D* поместить на отрезке *AB* таким образом, чтобы $\frac{|AD|}{|DB|} = \frac{k_B}{k_A}$. Тогда нахождение $|CD|$ по из-

вестным $|CA|$, $|CB|$, $|AB|$ становится элементарной задачей планиметрии.

§ 1.7. Поиск логических закономерностей

Задача поиска закономерностей состоит в том, чтобы найти в пространстве переменных области заданного вида, которые обладали бы заданной отличительной особенностью, например, содержали относительно много объектов выделенного класса и мало объектов остальных классов.

Простейшим, и вполне пригодным для практического использования, критерием выбора области, характеризующей j-й класс, может быть следующий

$$K_j(A) = \frac{1}{N}\left(N\left(x^i \in A,\ y^i = j\right) - N\left(x^i \in A,\ y^i \neq j\right)\right).$$

Критерий представляет собой разность между числом попавших в область A объектов выборки, принадлежащих классу j, и числом попавших в область A объектов выборки, принадлежащих другим классам.

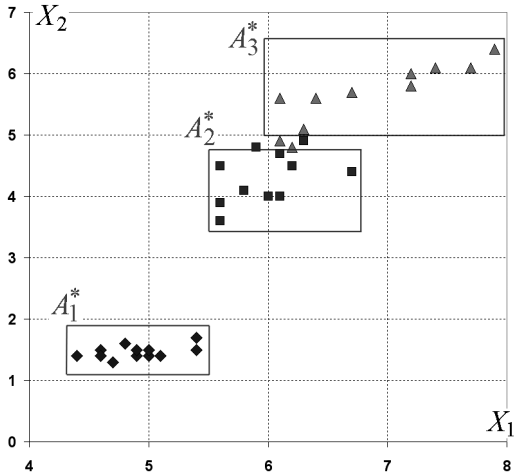

Рис. 15. Логические закономерности для задачи Iris

Логической закономерностью для класса j будем называть область $A_j^* \subseteq X$ в форме многомерного интервала (гиперпараллелепипеда, в частном случае прямоугольника или отрезка), на корой критерий $K_j(A)$ принимает максимальное значение.

Для иллюстрации используем данные по ирисам из таблицы 1. На рис. 15 прямоугольниками изображены закономерности для каждого из трёх классов.

В задаче кластерного анализа имеет смысл поиск закономерностей, содержащих большое число объектов при малом объёме области. В этом случае критерий качества кластера может иметь вид

$$K(A) = \frac{1}{N} N\left(x^i \in A\right) - \frac{\mu(A)}{\mu(X)},$$

где $\mu(A)$ – мера (длина, площадь, объём) области A.

Если требуется найти несколько закономерностей для одного класса, то можно воспользоваться следующим методом: объекты, вошедшие в уже найденные закономерности, удаляются из обучающей выборки, после чего ищутся закономерности для оставшихся объектов в соответствии с тем же критерием. Это повторяется, пока не будет получено заданное число закономерностей.

§ 1.8. Задача поиска глобального экстремума

Под экстремумом понимается минимальное или максимальное значение функции. Без ограничения общности можно считать, всегда ищется максимум, поскольку, если требуется найти минимум, всегда можно вместо исходной функции рассматривать функцию, умноженную на –1, у которой максимум достигается в точке минимума исходной функции.

О поиске глобального экстремума имеет смысл говорить, когда функция является многоэкстремальной, т.е. имеет более одного локального экстремума. Локальным максимумом называется точка, в которой значение функции больше, чем в любой точке некоторой её окрестности.

Одним из наиболее наглядных примеров многоэкстремальной функции является рельеф местности, т.е. зависимость высоты над уровнем моря от географических координат точки.

Задача поиска глобального экстремума (или задача оптимизации) заключается в следующем. Требуется выбрать последовательность точек в пространстве X так, чтобы среди значений функции в выбранных точках оказалось значение, близкое к максимальному. При этом выбор последующих точек зависит от значений функции в точках, выбранных ранее. Другими словами, задана некоторая функция, которая скрыта. Исследователь может выбирать любые точки из области определения и узнавать значение функции в них. Задача заключается в том, чтобы, использовав как можно меньшее число точек, обнаружить как можно большее значение функции.

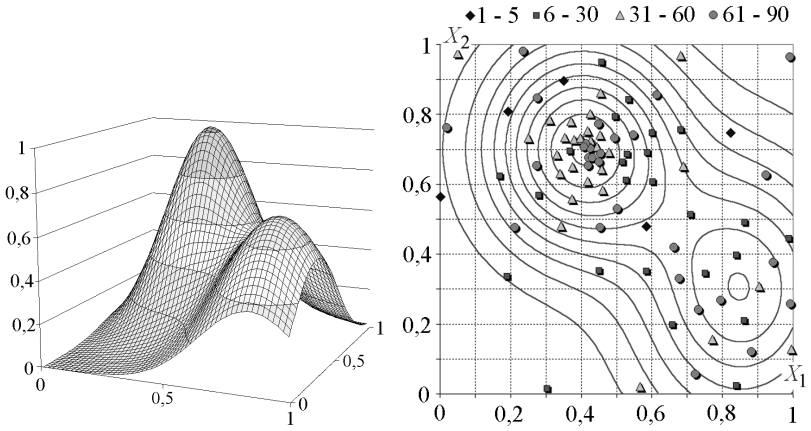

Рис. 16. Случайный адаптивный поиск глобального экстремума

Для решения задач многоэкстремальной оптимизации наиболее часто используется подход, основанный на идее адаптивного поиска. Данная идея заключается в том, что точки выбираются в соответствии с некоторой мерой «перспективности» областей пространства X. При этом «перспективность» области тем выше, чем большие значения функции в ней уже обнаружены и чем

«менее исследована» область, т.е. чем меньше точек в ней поставлено ранее.

Проиллюстрируем идею адаптивного поиска на примере.

Пусть требуется найти глобальный экстремум функции

$$\psi(x_1, x_2) = e^{15(x_1-0,4)^2+12(x_2-0,7)^2} + 0,7e^{12(x_1-0,85)^2+7(x_2-0,3)^2}$$

на интервале $X = X_1 \times X_2$, $X_1 = [0,1]$, $X_2 = [0,1]$. График функции приведён на левой диаграмме рис. 16.

На правой диаграмме рис. 16 линиями уровней изображена целевая функция и показана расстановка точек при поиске экстремума с помощью одной из модификаций алгоритма случайного поиска с адаптацией. Разные маркеры дают информацию о том, в какой последовательности выбирались точки. Легко заметить, что чем позднее ставились точки, тем больше они тяготеют к глобальному экстремуму. За счёт этого алгоритм приближается к экстремуму гораздо быстрее, чем при поиске наугад (равномерная расстановка).

Идеи адаптивного поиска также развиваются в эволюционных (в т.ч. генетических) алгоритмах.

Задача поиска экстремума может рассматриваться как частный случай задачи планирования эксперимента.

Задача планирования эксперимента это особая модификация задачи восстановления зависимостей, когда не задана, а формируется при участии исследователя, который может выбирать точки пространства *X*, для которых будет сообщены значения целевой функции. Требуется выбирать точки таким, образом, чтобы как можно точнее оценить целевую функцию. Критерием качества такой оценки обычно выступает средний квадрат отклонения.

Требование найти экстремум функции может считаться одним из вариантов критерия качества восстановления зависимости.

§ 1.9. Оценивание достоверности решения

В задачах машинного обучения решающая функция строится по обучающей выборке, однако применяется она для новых объектов генеральной совокупности, которых в обучающей выборке

не было. Естественно возникает необходимость оценивания качества решения (например, вероятность ошибочной классификации) на новых объектах.

1.9.1. Использование контрольной выборки

Наиболее простым способом оценивания качества решающей функции является использование контрольной выборки.

Пример такого подхода был приведён при прогнозировании временного ряда.

Контрольной выборкой называется выборка, взятая из того же распределения, что и обучающая, но не использовавшаяся при построении решающей функции.

Обозначим N^* – объём контрольной выборки, N^*_{err} – количество ошибочно классифицированных объектов контрольной выборки, $R^* = \frac{N^*_{err}}{N^*}$ – частота ошибок.

Интервальные оценки риска

Оценивание вероятности ошибочной классификации R на основе полученной частоты R^* есть классическая задача оценивания параметра биномиального распределения.

Её решением будет построение доверительного интервала $[0, \hat{R}(R^*)]$, где оценочная функция $\hat{R}(R^*)$ находится из уравнения

$$P\left(R \le \hat{R}(R^*)\right) = \eta,$$

где η – доверительная вероятность, значение которой следует выбирать близким к 1. В зависимости от требуемой надёжности решения значение η выбирают обычно 0,9, 0,95 или 0,99.

Графики функции $\hat{R}(R^*)$ при различных N^* приведены на рис. 17. Если, например, на контрольной выборке объёма 10 не было ни одной ошибки классификации, то с надёжностью 0,9 можно утверждать, что R не превосходит 0,21.

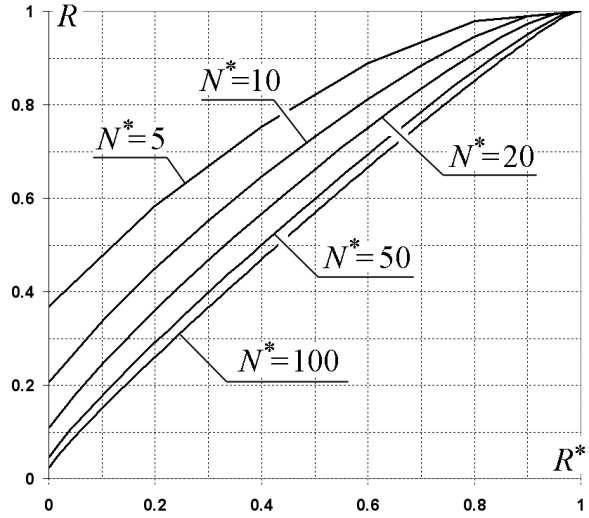

Рис. 17. Доверительные интервалы для риска при $\eta = 0{,}9$

Точечные оценки риска

В некоторых случаях доверительный интервал для риска оказывается не вполне удобен для содержательной интерпретации. Иногда удобнее иметь оценку риска просто в виде числа. Такие оценки называются точечными.

В качестве точечной оценки для *R* может использоваться сама частота R^*. Однако такая оценка будет не вполне адекватной. Причина, главным образом, в том, что R^* вполне может оказаться равной 0, но это ещё не даёт оснований полагать, что и вероятность ошибочной классификации равна 0. Действительно, нулевая вероятность гарантирует, что соответствующее событие не произойдёт (хотя оно при этом не обязательно является невозможным), однако отсутствие ошибок на конечной контрольной выборке ещё не гарантирует полное отсутствие ошибок на новых объектах.

Таким образом, «разумная» точечная оценка риска при $R^* = 0$ должна быть ненулевой. Подходящую оценку можно построить, используя, например, байесовский подход.

К байесовскому подходу относятся статистические методы, в которых на статистических гипотезах задаётся априорное распределение.

В данном случае в роли статистической гипотезы выступает неизвестное значение риска *R*. Поскольку риск изменяется в конечных пределах (от 0 до 1), естественно априорное распределение на нём задать равномерным, т.е. плотность вероятности $\varphi(R) = 1$, при $0 \le R \le 1$.

Вероятность получить на контрольной выборке долю ошибок R^* есть

$$P\left(R^* / R\right) = P\left(N^* / R\right) = C_{N^*}^{N^*_{err}} R^{N^*_{err}} (1-R)^{N^* - N^*_{err}}.$$

По формуле Байеса можно найти апостериорную (при условии полученного в эксперименте R^*) плотность вероятности для *R*

$$\varphi\left(R / R^*\right) = P\left(R^* / R\right) \frac{\varphi(R)}{P(R^*)},$$

где $P(R^*)$ находится по формуле полной вероятности

$$P(R^*) = \int_{-\infty}^{+\infty} P\left(R^* / R\right) \varphi(R) dR.$$

Последний интеграл легко берётся по частям, в результате получается $P(R^*) = \frac{1}{N^* + 1}$.

Найдя апостериорную плотность вероятности для *R*, можно вычислить условное математическое ожидание

$$E\left(R / R^*\right) = \int_{-\infty}^{+\infty} R \varphi\left(R / R^*\right) dR = \frac{N^*_{err} + 1}{N^* + 2}.$$

Полученную величину можно использовать как точечную оценку риска.

Точечную оценку риска можно получить и на основе интервальной. Для этого заметим, что $\hat{R}(R^*)$ зависит от доверительной вероятности η и при фиксированном R^* может рассматриваться как функция от η. Можем также рассмотреть обратную функцию $\eta(\hat{R})$, которая является монотонно неубывающей и изменяется от 0 до 1, т.е. удовлетворяет свойствам функции распределения, поэтому по ней можно произвести усреднение

$$E\left(\hat{R}/R^*\right)=\int_0^1 \hat{R}\, d\eta(\hat{R})=\frac{N^*_{err}+1}{N^*+1}.$$

Полученная величина, конечно, не является математическим ожиданием, но практически может быть использована как ожидаемое (предполагаемое) значение риска, т.е. в качестве точечной оценки.

Заметим, что такая оценка близка оценке, полученной в рамках байесовского подхода.

1.9.2. Оценка скользящего экзамена

На практике для оценивания качества решающей функции чаще всего используется оценка скользящего экзамена.

Эта оценка получается путём многократного разбиения обучающей выборки на две части: по одной строится решение, по второй оценивается качество решения. После чего оценка качества усредняется по всем разбиениям выборки.

Для иллюстрации построения оценки скользящего экзамена рассмотрим решение задачи классификации в разделе 1.1.1 методом ближайшего соседа.

Когда объект исключается из обучающей выборки, решение в точке, где он находился, принимается по ближайшему из оставшихся объектов. При этом если ближайший объект оказывается «чужого» класса, то выбранный объект на скользящем экзамене классифицируется ошибочно.

На рис. 18 ошибочно классифицируемые на скользящем экзамене объекты отмечены изменёнными маркерами (с выделенным контуром). Из 36 объектов обучающей выборки ошибочно классифицированы 7, таким образом оценка риска по скользящему контролю составляет $\breve{R} = \frac{\breve{N}_{err}}{N} = \frac{7}{36}$.

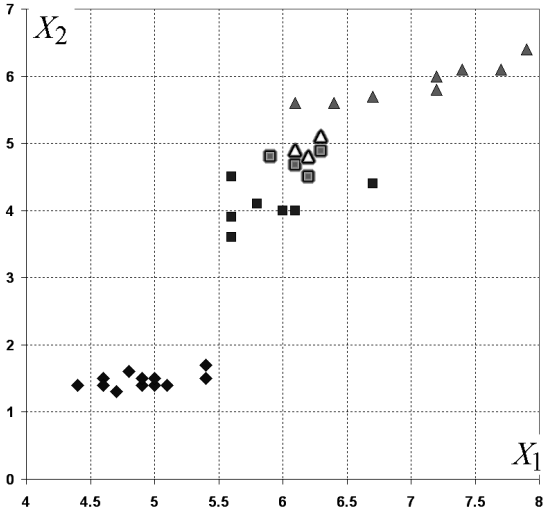

Рис. 18. Применение скользящего экзамена

Оценка скользящего экзамена является несмещённой оценкой вероятности ошибочной классификации, а именно: для любого распределения в пространстве переменных математическое ожидание оценки скользящего экзамена при объёме выборки $N+1$ совпадает с математическим ожиданием вероятности ошибочной классификации при объёме выборки N.

При этом следует понимать, что если мы получили значение оценки скользящего экзамена, например, равное 0, это не значит, что ожидаемое значение риска равно 0. Во-первых, когда решающая функция уже построена, вероятность ошибочной классификации для неё уже не является случайной величиной, поэтому говорить о её математическом ожидании не имеет смысла. Во-

вторых, «ожидаемое», в смысле разумной практической оценки, значение риска (т.е. величина, на которую следует рассчитывать) в этом случае существенно больше 0.

1.9.3. Статистическое моделирование

Оценка скользящего экзамена является точечной оценкой, причём её статистическая погрешность обычно неизвестна.

Получить информацию о свойствах используемого метода построения решающих функций можно с помощью статистического моделирования.

Нулевая гипотеза

Под «нулевой» гипотезой в задаче классификации естественно понимать случай когда распределения для всех классов совпадают, при этом безусловные вероятности классов одинаковы.

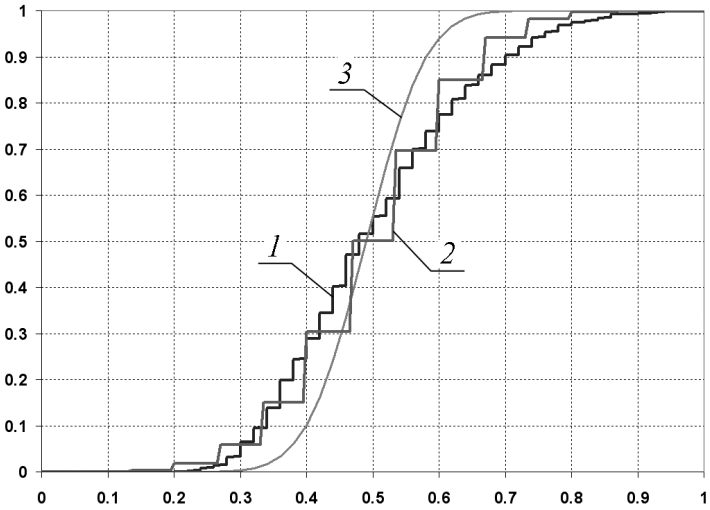

Рис. 19. Эмпирическое распределение скользящего экзамена

Очевидно, что при нулевой гипотезе для любого решающего правила вероятность ошибочной классификации равна $\frac{k-1}{k}$, где k – число классов.

Знание точной вероятности ошибочной классификации позволяет исследовать качество оценок риска, т.е. насколько хорошо они оценивают искомую вероятность.

На рис. 19 кривая 1 отражает эмпирическую функцию распределения для оценки скользящего экзамена при нулевой гипотезе. Использован метод классификации деревьями решений (равномерное распределение на квадрате, «жадный» алгоритм, три конечных вершины), объём обучающей выборки $N = 50$.

Поведение указанной кривой характерно для оценки скользящего экзамена, в частности разброс значений (дисперсия) у неё гораздо больше, чем у оценки (сглаженная функция распределения изображена кривой 3) на контрольной выборке того же объёма. Как можно заметить, распределение оценки скользящего экзамена ближе всего к распределению оценки (кривая 2) на контрольной выборке объёма $N^* = 15$.

Это говорит о том, что скользящий экзамен позволяет оценивать риск гораздо менее точно, чем контрольная выборка того же объёма.

Нормальный закон

Статистическое моделирование заключается в генерации большого числа выборок из заданного распределения и построении решающих функций по этим выборкам.

Могут использоваться любые распределения, чем разнообразнее их выбор, тем лучше.

Рассмотрим способы моделирования выборок из нормального распределения, как одного из наиболее часто используемых.

В стандартную библиотеку большинства языков программирования входит функция, генерирующая псевдослучайное число из равномерного распределения в интервале от 0 до 1. На основе этой функции можно смоделировать реализации случайной величины, имеющей любое заданное распределение.

Пусть величина Z распределена равномерно на [0, 1] и пусть требуется смоделировать случайную величину X с функцией распределения $F(x)$. Для этого достаточно сгенерировать z из равномерного распределения и взять $x = F^{-1}(z)$, т.е. вычислить значение функции, обратной к функции распределения.

Функция нормального распределения представляет собой интеграл, не выражающийся в элементарных функциях (не «берущийся» аналитически). Однако при вычислении на компьютере любые функции (синус, корень, экспонента и т.п.) вычисляются через приближение степенными рядами. А поскольку ряды интегрируются совершенно очевидным образом, вычисление функция нормального распределения программно реализуется не многим сложнее вычисления элементарных функций.

Тем не менее, реализация функции, обратной к функции нормального распределения отсутствует в стандартных библиотеках, а самостоятельная её реализация требует определённых усилий, поэтому для моделирования нормальной случайной величины обычно используют другие методы.

Один из них основан на том, что двумерная нормальная плотность после тригонометрической замены переменных становится интегрируемой аналитически.

Пусть X и Y независимы и имеют нормальное распределение со средним 0 и дисперсией 1. произведём тригонометрическую замену переменных $x = r\cos\alpha$, $y = r\sin\alpha$ и затем замену $2t = r^2$

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \frac{1}{2\pi} e^{-\frac{x^2+y^2}{2}}\, dxdy = \int_{0}^{2\pi}\int_{0}^{+\infty} \frac{1}{2\pi} e^{-\frac{r^2}{2}}\, r\, drd\alpha = \int_{0}^{2\pi}\int_{0}^{+\infty} \frac{1}{2\pi} e^{-t}\, dtd\alpha .$$

Видим, что нормально распределённые величины выразились через α, которая распределена равномерно в $[0,2\pi]$ и t, имеющую экспоненциальное распределение.

Функция экспоненциального распределения есть $1-e^{-t}$. Беря обратную, получаем $t = -\ln(1-z)$.

Таким образом, сгенерировав два псевдослучайных значения z_1 и z_2, получаем два значения из нормального распределения: $x = \sqrt{-\ln(1-z_1)}\cos(2\pi z_2)$ и $y = \sqrt{-\ln(1-z_1)}\sin(2\pi z_2)$.

Приведённый метод удобен тем, что использует только стандартные функции, однако он требует вычисления нескольких функций, поэтому, скорее всего, не вполне оптимален по вычислительной трудоёмкости.

Наиболее простой способ смоделировать значение из нормального распределения — сложить 12 псевдослучайных значений, равномерно распределённых в [0, 1], и вычесть 6. В силу центральной предельной теоремы полученная сумма будет иметь приближённо нормальное распределение.

Пусть теперь требуется сгенерировать вектор значений $x = (x_1, ..., x_n)$ из многомерного нормального распределения с произвольными параметрами $\mu = (\mu_1, ..., \mu_n)$ – вектор средних и $\Lambda = (\lambda_{ij})$ – ковариационная матрица.

Ковариационную матрицу можно представить в виде $\Lambda = TBT'$, где матрица B диагональная, а T – оператор поворота.

Для получения требуемого случайного вектора следует выполнить следующие шаги:

— сгенерировать n значений из одномерного нормального распределения с параметрами (0, 1), обозначим полученный вектор $r = (r_1, ..., r_n)$;

— умножить каждое значение на корень из соответствующего диагонального элемента матрицы B;

— применить к полученному вектору значений оператор поворота;

— прибавить к результату вектор средних.

В результате будет получен случайный вектор из требуемого распределения

$$x = T\sqrt{B}\,r + \mu .$$

На практике часто удобнее вместо ковариационной матрицы изначально задавать B и T.

Дерево решений

Во многих случаях для статистического моделирования полезно задание распределения с помощью дерева решений.

Чтобы таким образом задать совместное распределение, нужно сначала задать безусловное распределение $P(x)$. Для этого проще всего взять равномерное распределение в единичном гиперкубе (если исследуемый метод классификации не инвариантен к линейным преобразованиям переменных, вместо гиперкуба можно взять произвольный многомерный интервал).

Каждой конечной вершине приписывается некоторый класс.

Разыгрывается точка в соответствии с $P(x)$.

Определяем, в какую вершину она попадает.

С вероятностью $1-R_0$ приписываем объекту класс, назначенный вершине, с вероятностью R_0 – другой класс. Параметр R_0 задаёт байесовский уровень ошибки.

Глава 2. Задача машинного обучения в вероятностной постановке

В данной главе будет проведена систематизация изложенного ранее материала и дан обзор задач и методов машинного обучения.

§ 2.1. Статистическая постановка задачи анализа данных

Рассмотрим задачи машинного обучения с их формальными (математическими) постановками.

2.1.1. Задача построения решающей функции

Сформулируем задачу построения решающих функций в общем виде.

Пусть X – пространство значений переменных, используемых для прогноза, а Y – пространство значений прогнозируемых переменных, и пусть C – множество всех вероятностных мер на заданной σ-алгебре подмножеств множества $D = X \times Y$.

При каждом $c \in C$ имеем вероятностное пространство: $\langle D, B, P_c \rangle$, где B – σ-алгебра, $P_c[D]$ – вероятностная мера (в квадратных скобках мы указываем не аргумент функции, а множество, на котором задана σ-алгебра).

Параметр c будем называть стратегией природы. Термин стратегия природы взят из теории игр. Забегая вперёд, отметим, что стратегией игрока (исследователя, распознавателя) будет используемый им алгоритм построения решающих функций. Заметим, что мы не будем использовать какие-либо результаты теории игр, а только некоторую терминологию.

Решающей функцией называется соответствие $f : X \to Y$.

Качество принятого решения оценивается заданной функцией потерь $L : Y^2 \to [0, \infty)$. Функция потерь задает цену ошибки как

меру несоответствия принятого решения $f(x)$ и истинного значения y.

Под риском будем понимать средние потери:

$$R(c, f) = \int_D L(y, f(x)) dP_c[D].$$

Заметим, что значение риска зависит от стратегии природы c — распределения, которое неизвестно.

Пусть $v = \left\{ \left(x^i, y^i \right) \in D \;\middle|\; i = \overline{1, N} \right\}$ — случайная независимая выборка из распределения $P_c[D]$.

Пусть $Q : \{v\} \to \Phi$ – метод построения решающих функций, а $f_{Q,v} \in \Phi$ – функция, построенная по выборке v алгоритмом Q.

Задача построения решающей функции заключается в выборе подходящего алгоритма Q и в оценивании риска принятого решения.

Риск — естественный функционал качества. Мы не можем выбирать его по своему усмотрению, поскольку он должен отражать реальные потери от неверного решения, т.е. «цену ошибки».

Функционалом принято называть функцию, областью значений которой являются числа, а областью определения — функции. В данном случае риск — это функционал от двух функций: вероятностной меры и метода (алгоритма).

Задача заключается в том, чтобы найти метод построения решающих функций, который бы минимизировал риск. Однако риск зависит от неизвестного распределения, и при разных распределениях лучшими будут разные алгоритмы, поэтому задача пока не является строго поставленной.

2.1.2. Общая постановка

Не во всех задачах анализа данных требуется построить отображение $f : X \to Y$. Например, в задаче кластерного анализа требуется выделить области в пространстве X.

Чтобы обобщить постановку задачи следует рассматривать пространство наблюдений, распределение для которого опреде-

ляется стратегией природы, пространство решений и функционал качества (риск). Для кластерного анализа таким функционалом может быть, например, разность между вероятностной мерой и мерой Лебега.

Пространство наблюдений — это множество всех выборок (объём обычно фиксирован). Могут, однако, рассматриваться и другие, более сложные, пространства наблюдений, например, множество экспертных высказываний.

Дальнейшим обобщением будут задачи планирования эксперимента, однако формальная постановка, которая распространялась бы и на эти задачи, будет весьма громоздкой, поэтому здесь приводиться не будет.

2.1.3. Иллюстративный пример

Чтобы проиллюстрировать сложность проблемы построения решающих функций, рассмотрим простейший пример. Это предельно упрощённый вариант задачи классификации.

Пусть пространство X вообще не используется, а решающая функция относит все объекты к одному классу, которых всего два. Формально можно считать, что X состоит из одной точки (в которую попадают все объекты генеральной совокупности). Тогда возможных решающих функций всего две: одна отображает всё X в первый, другая — во второй класс.

Выборки в данной ситуации различаются только количеством объектов первого и второго класса в них. Пусть N_1 – количество объектов первого класса. Очевидно, что N_1 может изменяться от 0 до N, где N – общий объём выборки, поэтому имеем всего $N+1$ возможных выборок.

Алгоритм Q может отобразить любую выборку в любую решающую функцию. Число возможных алгоритмов составляет 2^{N+1}. При небольших N мы можем все их исследовать.

Первая задача — отсеять из этих алгоритмов заведомо непригодные для использования, а именно доминируемые. Напомним, что стратегия игрока называется доминируемой, если существует

другая стратегия, которая при любых условиях не хуже её, а в каких-то случаях — лучше.

В нашем случае доминируемым будет алгоритм, который при любых распределениях (стратегиях природы) хуже некоторого другого алгоритма.

Если мы отсеем доминируемые алгоритмы, то останутся только алгоритмы, оптимальные по Парето. Для любой пары из них найдётся распределение, на котором лучше первый, и найдётся распределение, на котором лучше второй.

Таблица 12. Различные алгоритмы классификации при $N = 6$.

N_1	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*
6	1	1	1	1	1	1	1	1	0	1	1
5	1	1	1	1	1	1	1	0	0	0	1
4	1	1	1	1	1	1	0	0	0	1	0,6
3	1	1	1	1	0,5	0	0	0	0	0	0
2	1	1	1	0	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0

В зависимости от числа объектов первого класса в обучающей выборке алгоритм приписывает всей генеральной совокупности первый или второй класс. Некоторые алгоритмы при объёме выборки 6 приведены в таблице 12. Приведено 11 алгоритмов, число 1 в таблице означает, что выбран первый класс, число 0 — второй. Так алгоритм 1 всегда выбирает решение, которое приписывает первый класс, алгоритм 9 — всегда второй, алгоритм 2 выбирает первый класс всегда, кроме случаев, когда его вообще не встречается в выборке.

Кроме обычных детерминированных будем рассматривать также стохастические решающие функции, которые выбирают класс случайно, с определённой вероятностью. На самом деле, в таблице 12 приведены вероятности того, что решающая функция, выбранная алгоритмом, сопоставит первый класс.

Самым естественным выглядит алгоритм 5, который выбирает класс, которого в выборке больше, а при равенстве частот выбирает решение, приписывающее класс равновероятно.

Алгоритм 10 выглядит, наоборот, нелогичным: он выбирает первый класс при частоте 6, второй класс при $N_1 = 5$, и снова первый класс при $N_1 = 4$.

Алгоритм 1 тоже выглядит сомнительным: он выбирает первый класс, даже если он вообще не встретился в обучающей выборке. Но такой алгоритм всё же может быть оправдан — если мы априори знаем, что в генеральной совокупности преобладаем первый класс, и его отсутствие в выборке — случайность. Однако «оправдать» алгоритм 10 подобным образом не получается.

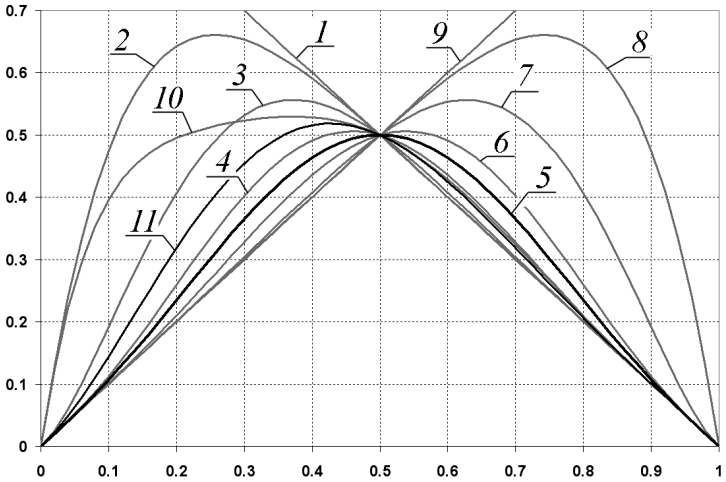

Рис. 20. Зависимости вероятности ошибочной классификации от стратегии природы

На рис. 13 показаны зависимости риска от стратегии природы для алгоритмов из таблицы 12. Стратегия природы здесь задаётся безусловной вероятностью первого класса. Видим, что алгоритм 10 не доминируется никакими детерминированными алгоритмами, но доминируется алгоритмом 11. Таким образом, при-

менять алгоритм 10 не имеет смысла ни при каких предположениях и обстоятельствах.

Применение любых недоминируемых алгоритмов может быть оправдано при соответствующих априорных предположениях о распределении. Однако если априорной информации нет, наиболее разумным выглядит алгоритм 5.

Можно сформулировать, чем именно он предпочтителен. При сравнении с любым другим алгоритмом видим, что его выигрыш (преимущество) в наилучшей ситуации больше, чем проигрыш в наихудшей.

2.1.4. Варианты формальных постановок

Задача нахождения наилучшего алгоритма очень интересна, но в настоящее время нет общепризнанных критериев сравнения алгоритмов, поэтому в том виде, как она сформулирована в разделе 2.1.1, задача редко ставится.

Обычно рассматриваются более частные задачи. Одну из наиболее важных можно сформулировать следующим образом: существует некоторый алгоритм построения решающих функций, который хорошо зарекомендовал себя на практике, требуется дать теоретическое обоснование его применимости. Последнее обычно подразумевает построение оценок качества результатов его работы, а именно, оценивание риска.

Другой важной задачей является эффективная реализация метода, в частности определение класса вычислительной трудоёмкости для оптимизационной задачи, содержащейся в методе.

Задача оценивания риска

Эмпирический риск определим как средние потери на выборке:

$$\tilde{R}(\nu, f) = \frac{1}{N}\sum_{i=1}^{N} L\left(y^i, f\left(x^i\right)\right).$$

Оценка риска на контрольной выборке определяется как

$$R^*\left(v^*, f\right) = \frac{1}{N^*}\sum_{i=1}^{N^*} L\left(y_i^*, f\left(x_i^*\right)\right),$$

где $v^* = \left\{ \left(x_i^*, y_i^*\right) \in D \;\middle|\; i = \overline{1, N^*} \right\}$ – «новая» случайная независимая выборка из $P_c[D]$.

Пусть $Q : \{v\} \to \Phi$ – метод построения решающих функций, а $f_{Q,v} \in \Phi$ – функция, построенная по выборке *v* алгоритмом *Q*.

Функционал скользящего экзамена определяется как

$$\breve{R}(v, Q) = \frac{1}{N}\sum_{i=1}^{N} L\left(y^i, f_{Q,v_i'}\left(x^i\right)\right),$$

где $v_i' = v \setminus \left\{\left(x^i, y^i\right)\right\}$ – выборка, получаемая из v удалением *i*-го наблюдения.

Интервальные оценки риска более информативны, по сравнению с точечными.

Доверительный интервал для R будем задавать в виде $[0, \hat{R}(v)]$. Имеет смысл ограничиться односторонними оценками, поскольку на практике для риска важны именно оценки сверху.

Таким образом, в данном случае построение доверительного интервала эквивалентно выбору функции $\hat{R}(v)$, которую будем называть оценочной функцией или просто оценкой (риска).

При этом должно выполняться условие:

$$\forall c, \quad P\left(R \le \hat{R}(v)\right) \ge \eta,$$

где η – заданная доверительная вероятность.

Известные на данный момент оценки риска строятся не как функции непосредственно выборки, а через композицию $\hat{R}(v) = R_e\left(\overline{R}(v)\right)$, то есть как функции значений некоторого эмпирического функционала $\overline{R}(v)$, в качестве которого обычно выступает эмпирический риск или скользящий экзамен.

Анализ заданной таблицы данных

Зачастую в постановках задач машинного обучения имеющаяся таблица данных не рассматривается как выборка из некоторой генеральной совокупности, а полагается существующей самой по себе. При этом требуется построить решающую функцию для данной выборки, не заботясь, как она будет работать для новых объектов.

Такая постановка часто используется в задачах кластерного анализа.

В этих случаях задаётся эмпирический (выборочный) функционал качества. Зачастую он выбирается эвристически – например как отражение представлений о «хорошей» кластеризации.

Математическая задача состоит в минимизации функционала качества.

Такая постановка задачи не является статистической (вероятностной), но может быть связана с ней. Действительно, минимизируя выборочный аналог критерия, мы в какой-то степени минимизируем и сам критерий. Например, метод, минимизирующий эмпирический риск, в существенной мере минимизирует и риск.

Однако эта связь не однозначна. Как минимум, нужно учитывать сложность решения.

Перечень задач

Ниже приведён неполный перечень основных задач анализа данных.

Распознавание образов или классификация «с учителем».

Регрессионный анализ или восстановление зависимостей.

Кластерный анализ. Синонимы: таксономия, автоматическая группировка, классификация «без учителя».

Задача упорядочивания объектов. Пример: ранжирование участков территории месторождения по запасам полезных ископаемых.

Задача обнаружения закономерностей в эмпирических данных.

Прогнозирование временных рядов.

Планирование эксперимента. Как уже говорилось, планирование эксперимента — это не самостоятельная задача анализа данных, а особая постановка задач, возникающая при возможности активно влиять на формирование таблицы данных.

Поиск глобального экстремума.

§ 2.2. Обзор методов машинного обучения

Поскольку метод (алгоритм) построения решающей функции есть функция (отображение множества выборок во множество решающих функций), то методы, как функции, можно классифицировать на явные и неявные.

Такая классификация имеет смысл, однако более распространённым и важным является разделение методов на методы, оценивающие распределения, и методы непосредственного построения решающих функций.

2.2.1. Методы, основанные на восстановлении распределений

Методы основаны на том, чтобы сначала оценить распределения, а потом построить решающую функцию, подставив оценку вместо распределения.

Традиционно в прикладной статистике принято разделять параметрические и непараметрические методы.

Параметрические методы

В параметрических методах предполагается, что оцениваемое распределение принадлежит заданному параметрическому семейству, например, классу нормальных распределений. В этом случае для восстановления распределения достаточно оценить параметры.

Частным случаем параметрического подхода является байесовский подход. В этом случае на параметрах постулируется некоторое априорное распределение и по формуле Байеса вычисляется апостериорное (при условии выборки) распределение, по которому можно строить решение.

Непараметрические методы

Данный класс методов основан на непараметрическом оценивании распределения. Пример — парзеновские оценки плотности, которые можно рассматривать как обобщение одного из базовых методов оценивания плотности — с помощью гистограммы.

2.2.2. Методы, конструирующие решающие правила

Методы с явным заданием решения

Примером метода из данного класса может служить дискриминант Фишера, который аналитически выражается как явная функция выборки.

К этой категории также относятся метрические методы построения решающей функции, в частности, рассмотренный ранее метод прецедентов.

Методы на основе оптимизации эмпирического критерия

Большой класс методов предполагает оптимизацию некоторого функционала качества (как правило, эмпирического риска) в некотором классе решающих функций, например метод опорных векторов, решающие деревья, нейронные сети.

Идея метода опорных векторов заключается в построении разделяющей поверхности (в простейшем случае — линейной), находящейся на максимальном расстоянии от ближайших представителей разделяемых классов.

Для построения решающих функций используются нейронные сети прямого распространения, которые представляют собой суперпозицию базовых функций (нейронов).

Базовые функции обычно выбираются в виде суперпозиции линейной комбинации аргументов и так называемой передаточной функции (нелинейной).

Результирующее значение нейронной сети прямого распространения является однозначной функцией входных значений.

Другим классом нейронных сетей являются сети «с памятью». Простейший пример такой сети — триггер, см. рис. 21.

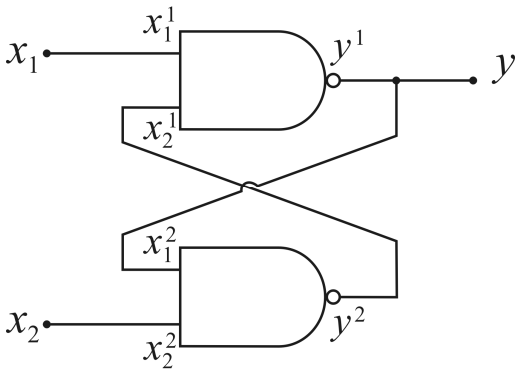

Рис. 21. Простейший триггер

Триггер может быть составлен из двух логических элементов «и-не», которые могут рассматриваться как частный случай базовых функций.

Данная сеть может находиться в пяти состояниях, см. таб. 13, три из которых однозначно определяются входными аргументами x_1 и x_2, а последние (4*a* и 4*b*) зависят от предыдущего состояния.

Таблица 13. Состояния триггера

i	x_1	x_2	x_1^1	x_2^1	y^1	x_1^2	x_2^2	y^2	y
1	0	0	0	1	1	1	0	1	1
2	0	1	0	1	1	1	1	0	1
3	1	0	1	1	0	1	0	1	0
4*a*	1	1	1	0	1	1	1	0	1
4*b*	1	1	1	1	0	0	1	1	0

Состояние 4*a* возникает после 2, 4*b* — после 3. При одновременном изменении x_1 и x_2 с 0 на 1 переход в 4*a* или 4*b* недетерминирован.

Литература

1. Г.С. Лбов. Анализ данных и знаний : учебное пособие. – Новосибирск : Изд-во НГТУ, 2001. – 86 с.

2. К.В. Воронцов. Машинное обучение (курс лекций) – 2009. http://www.machinelearning.ru/, http://www.ccas.ru/voron/teaching.html.

3. В. М. Неделько. Основы математической статистики: методы анализа данных. Учебно-методическое пособие. НГТУ. 2008. 44 с.

4. Загоруйко Н. Г. Прикладные методы анализа данных и знаний / Н. Г. Загоруйко. – Новосибирск: Изд-во Института математики СО РАН, 1999. – 270 с.

5. Лбов Г. С. Методы обработки разнотипных экспериментальных данных / Г. С. Лбов. – Новосибирск: Наука, 1981. – 160 с.

6. Лбов Г. С., Старцева Н. Г. Логические решающие функции и вопросы статистической устойчивости решений / Г. С. Лбов, Н. Г. Старцева. – Новосибирск: Изд-во Института математики СО РАН, 1999. – 211 с.

7. Основы математической статистики : учебно-метод. пособие / Г.Н. Миренкова, С.В. Неделько, В.М. Неделько, Т.В. Тренёва. – Новосибирск : Изд-во НГТУ, 2008. – 36 с.

8. В.И. Лотов. Теория вероятностей и математическая статистика : курс лекций. – Новосибирск : Изд-во НГУ, 2006. – 128 с.

9. В.М. Неделько, Т.А. Ступина. Основы теории вероятностей и математической статистики в примерах и задачах. Учебное пособие. – Новосибирск: Издательство НГУ, 2006. – 82с.

10. Сборник задач по теории вероятностей, математической статистике и теории случайных функций / Под ред. А.А. Свешникова. – М.: Наука, 1965. – 632 с.

11. А.А. Боровков. Математическая статистика. Учебник. – М.: Наука, 1984. – 472 с.