



МИР математики

С.В. Умняшкин

Введение
в статистическую теорию
распознавания образов
и машинного обучения

2-е исправленное издание

Учебное пособие

ТЕХНОСФЕРА
Москва
2026

УДК 004.93 + 004.852

ББК 16.63

У54

У54 Умняшкин С.В.

**Введение в статистическую теорию распознавания образов
и машинного обучения: учебное пособие.**

2-е исправленное издание

М.: ТЕХНОСФЕРА, 2026. – 324 с.: ил. ISBN 978-5-94836-748-4

В рамках статистического подхода излагаются основы машинного обучения и распознавания образов. Изучаются методы и модели, применяемые для решения задач регрессии и классификации, кластеризации и анализа данных. Рассматриваются аспекты применения для решения указанных задач искусственных нейронных сетей, машин опорных векторов, ансамблей распознавателей, а также сопутствующие методы предварительной обработки данных, скрытые марковские модели. Для студентов, обучающихся по направлениям подготовки «Прикладная математика», «Математика и компьютерные науки».

УДК 004.93 + 004.852

ББК 16.63

©Умняшкин С.В., 2026

©АО «РИЦ «ТЕХНОСФЕРА», оригинал-макет, оформление, 2026

ISBN 978-5-94836-748-4

Содержание

Предисловие	6
Используемые обозначения	8
Глава 1. Основные задачи, модели и подходы к машинному обучению	9
1.1. Задачи распознавания образов и машинного обучения.....	9
1.2. Способность распознавателя к обобщению. Переобучение и регуляризация.....	16
1.3. Применение формулы Байеса для оценок параметров моделей.....	29
1.4. Обучение, проверка, выбор модели распознавателя. Характеристики качества классификации.....	37
1.5. Классификатор Байеса.....	46
1.6. Энтропия. Расхождение Кульбака — Лейблера.....	54
Глава 2. Линейные модели регрессии и классификации в машинном обучении	60
2.1. Линейная модель регрессионного распознавателя.....	60
2.2. Последовательное обучение регрессионного распознавателя.....	65
2.3. Регуляризация в методе наименьших квадратов.....	66
2.4. Представление ошибки регрессионного распознавателя в виде составляющих смещения и разброса.....	72
2.5. Линейные модели в задаче классификации.....	79
2.6. Нахождение параметров линейных моделей.....	83
2.7. Порождающие вероятностные модели классификаторов.....	89
2.8. Дискриминантные вероятностные модели классификаторов. Логистическая регрессия для двух классов.....	96
2.9. Логистическая регрессия с несколькими классами.....	103
2.10. Пробит-регрессия.....	105

Глава 3. Применение искусственных нейронных сетей для распознавания образов	108
3.1. Персептрон и его использование в бинарной классификации.....	108
3.2. Нейронные сети прямого распространения.....	118
3.3. Обучение нейронной сети методом обратного распространения ошибки.....	124
3.4. Перекрестная энтропия как функция штрафа в методе обратного распространения ошибки.....	132
3.5. Модифицированные методы градиентного поиска, ускоряющие обучение нейронных сетей.....	134
3.6. Инициализация сети и предобработка данных. Пакетная нормализация.....	141
3.7. Регуляризация при обучении нейронных сетей.....	147
3.8. Упрощение структуры сети на основе матрицы Гессе.....	154
3.9. Теорема об универсальной аппроксимации. Проблемы глубокого обучения.....	160
3.10. Сверточные нейронные сети — инструмент для распознавания изображений.....	163
Глава 4. Метод опорных векторов	176
4.1. Задача выпуклого программирования. Теорема Куна — Таккера.....	176
4.2. Метод опорных векторов для двух линейно разделимых классов.....	179
4.3. Метод опорных векторов для линейно неразделимых классов.....	186
4.4. Машины опорных векторов.....	192
4.5. Применение метода опорных векторов для задачи регрессии.....	201
4.6. Выявление аномалий данных с использованием метода опорных векторов.....	207



Глава 5. Ансамбли распознавателей.	
Отбор и преобразование признаков	213
5.1. Модель распознавателя как дерево решений. Деревья регрессии	213
5.2. Деревья классификации	219
5.3. Ансамбли распознавателей. Бэггинг, случайный лес	225
5.4. Усиление ансамбля распознавателей (бустинг)	230
5.5. Градиентный бустинг	238
5.6. Формирование и преобразование признаков	241
5.7. Селекция признаков. Метод главных компонент	247
5.8. Оценка значимости признаков для модели распознавателя	256
Глава 6. Обучение распознавателя без учителя	259
6.1. Задача кластеризации данных	259
6.2. Сети Кохонена	268
6.3. Самоорганизующиеся карты Кохонена	273
6.4. Модели распределений в виде гауссовых смесей	281
6.5. Общая схема алгоритма EM	289
Глава 7. Марковские модели последовательных данных	294
7.1. Цепи Маркова	294
7.2. Скрытые марковские модели	300
7.3. Определение параметров скрытых марковских моделей по выборке последовательных данных	304
7.4. Прямой и обратный проход в EM-алгоритме для скрытой марковской модели	309
7.5. Определение параметров скрытых марковских моделей по выборке: итоги	318
Библиографический список	322

Предисловие

Предлагаемое вниманию читателя пособие было написано как часть методического обеспечения (не включающая лабораторный практикум) учебного курса «Распознавание образов и машинное обучение», который на протяжении ряда лет читался автором в национальном исследовательском университете МИЭТ. Этот курс является введением в научную дисциплину, которая представляет собой один из краеугольных камней такой крайне важной и бурно развивающейся области знаний, как искусственный интеллект (ИИ).

В начале XXI века мощным импульсом развития теории и практики ИИ вообще и распознавания образов в частности послужил революционный прогресс в создании высокопроизводительных систем и платформ на основе технологий распределенных и параллельных вычислений. Вследствие этого, в частности, открылись новые горизонты для применения алгоритмов и методов на основе искусственных нейронных сетей (ИНС), которые на сегодняшний день являются технологической основой многих, если не большинства, систем ИИ.

Вопросам, связанным с применением ИНС для распознавания образов, в пособии уделено немалое внимание, но все же в ключе первого знакомства, причем в большей степени с теоретической точки зрения. ИНС — это очень важный для ИИ, но не единственный инструмент. Его совершенствование и усложнение невозможно без опоры на общую теорию машинного обучения и распознавания образов, в основе которой лежит весьма прочный математический фундамент. Учебное пособие представляет собой введение в данную теорию.

В концептуальном плане по используемому математическому аппарату в теории распознавания образов традиционно выделяются два подхода: детерминистский и статистический (вероятностный). Первый основывается прежде всего на математическом программировании, теории графов, математической логике и лингвистике. Во втором подходе доминирующую роль играют теории вероятностей и математическая статистика. Однако подобное

разделение становится все более условным вследствие наблюдаемой конвергенции этих подходов.

На взгляд автора, в терминах статистической теории исходные формулировки практических задач распознавания выглядят более естественно, хотя часто их постановки (и решения) могут быть даны и в таких формах, которые вообще не требуют привлечения понятия «вероятность». Как следует из названия пособия, при его написании автор придерживался главным образом статистического подхода. На такое методологическое видение, а также на используемые методические приемы изложения, в значительной степени повлияла классическая монография К. Бишопа [11].

Автор стремился придерживаться такого изложения, которое, с одной стороны, было бы в достаточной степени обосновано теоретически, а с другой — предполагало все-таки ориентацию на аудиторию студентов профильных инженерных направлений (математика и компьютерные науки, прикладная математика, программная инженерия), т. е. практиков. Для изучения материала пособия никаких начальных знаний из области машинного обучения и распознавания образов не требуется, однако предполагается, что в объеме вузовских математических курсов читатель достаточно хорошо владеет аппаратом математического анализа, линейной алгебры, теории вероятностей, а также знаком с основами методов оптимизации. Учебное пособие предназначено прежде всего для студентов бакалавриата и магистратуры, обучающихся по направлениям «Прикладная математика» и «Математика и компьютерные науки». Однако автор надеется, что пособие окажется полезным и для более широкого круга читателей.

Конечно, для инженера, и тем более исследователя, который будет далее специализироваться в области ИИ, освоения изложенного в пособии материала общего теоретического характера, скорее всего, окажется недостаточно для дальнейшей профессиональной деятельности. Со многими важными современными тенденциями, такими как обучение с подкреплением, состязательное обучение и многое другое, нужно будет познакомиться уже на следующем шаге изучения данной чрезвычайно быстро развивающейся предметной области. Для этого шага можно рекомендовать прежде всего монографии [1, 5, 7, 12].

Используемые обозначения

Везде в данном пособии автор стремился использовать однотипные обозначения в соответствии со следующими соглашениями.

Скалярные величины обозначаются латинским курсивом (как строчными, так и прописными буквами) или прямыми строчными греческими буквами, например: x , C , ω . Векторные величины считаются векторами-столбцами и обозначаются прямыми полужирными символами (по возможности строчными), например: \mathbf{x} , $\boldsymbol{\omega}$. Матрицы обозначаются прописными буквами полужирным шрифтом: \mathbf{X} , $\boldsymbol{\Omega}$. Верхний индекс T означает транспонирование матриц (векторов) и в записи вида $\mathbf{x} = (x_1, x_2, \dots, x_M)^T$ указывает на то, что \mathbf{x} — это вектор-столбец. Если интерпретация какого-либо вектора как строки или как столбца не играет роли, то указывающий на транспонирование индекс T может быть опущен: $\mathbf{x} = (x_1, x_2, \dots, x_M)$.

Случайные величины скалярного типа обозначаются прописным латинским курсивом (например X) или прямыми греческими буквами (например δ). Для векторных случайных величин используется прямой полужирный шрифт (как строчные, так и прописные буквы): \mathbf{X} , $\boldsymbol{\Omega}$, \mathbf{x} , $\boldsymbol{\omega}$. Вновь, если в контексте изложения трактовка некоторого случайного вектора как вектора-строки или вектора-столбца непринципиальна, может использоваться как обозначение $\mathbf{X} = (X_1, \dots, X_N)^T$, так и запись $\mathbf{X} = (X_1, \dots, X_N)$. Или $\mathbf{x} = (X_1, \dots, X_N)^T$, $\mathbf{x} = (X_1, \dots, X_N)$.

Для математического ожидания случайной величины X используется обозначение $M[X]$, для дисперсии — $D[X]$. Вероятности обозначаются прописными курсивными буквами P (например $P(y_k) = P\{Y = y_k\}$), а плотности вероятностей — строчными курсивными буквами: $p(x)$.

В тексте пособия используется двойная нумерация для рисунков, формул, примеров и упражнений: первая цифра обозначает главу, вторая — порядковый номер формулы (рисунка, примера, упражнения) в главе. Начало и окончание решений примеров обозначается соответственно символами ◀ и ▶.

ГЛАВА I

ОСНОВНЫЕ ЗАДАЧИ, МОДЕЛИ И ПОДХОДЫ К МАШИННОМУ ОБУЧЕНИЮ

I.1. Задачи распознавания образов и машинного обучения

Приведем сначала некоторые примеры практических задач, относящихся к предметной области *распознавания образов*:

- распознавание людей по фотографиям лиц;
- распознавание рукописного текста;
- распознавание речи (голосовых команд);
- медицинская диагностика заболеваний;
- геологическая диагностика;
- экономическое прогнозирование и т. д.

При теоретическом рассмотрении подобных задач под термином «распознавание образов» мы будем понимать отнесение *объекта* к тому или иному *классу* из конечного множества (*задача классификации*) или нахождение некоторого скалярного (векторного) значения, характеризующего определенные скрытые свойства объекта (*задача регрессии*) [8, 16, 22, 23, 25]. Для распознавания объектов используется их описание с помощью *образов*.

Так, в задачах медицинской диагностики объектами будут пациенты, а их образами — результаты анализов и другие числовые характеристики, такие как возраст, вес и т. п. Задача классификации в данном случае может состоять в отнесении пациента к здоровым или к больным (возможно, больным конкретным

заболеванием). Задачей регрессии будет, например, получение оценки вероятности полного выздоровления пациента после лечения.

Числовые характеристики, используемые для классификации образов, называются *признаками*. Признак — это некоторое количественное измерение объекта произвольной природы, а сформированный из них вектор признаков как раз и представляет собой образ объекта.

Пример 1.1. Рассмотрим задачу диагностики опухолей по результатам биопсии. Положим, что доброкачественные (класс А) и злокачественные (класс Б) изменения дают разную оптическую картину при изучении под микроскопом: гистологические образцы отличаются интенсивностью и контрастностью изображения. Для формирования вектора признаков будем использовать две числовых характеристики: среднее значение μ и среднеквадратичное отклонение σ яркости пикселей в цифровом изображении гистологического образца. Предположим, что имеется база данных образцов (рис. 1.1), для которых известна их правильная классификация: принадлежность к классам А (кружки) и Б (треугольники).

Множества точек, соответствующие образцам разных классов, на плоскости признаков (μ , σ) в данном случае разделимы прямой линией. Классификация неизвестного образа (соответствующая точка обозначена на плоскости признаков звездочкой) состоит

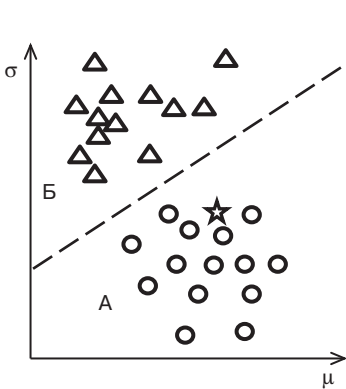


Рис. 1.1. Иллюстрация к примеру 1.1

в проверке положения этой точки относительно разделяющей прямой. Для приведенного на рис. 1.1 примера объект будет отнесен по его образу к классу А (доброкачественная опухоль).

Таким образом, далее в распознавании образов мы будем выделять задачи двух типов: *регрессии* и *классификации*. В математике регрессия — это условное математическое ожидание одной случайной величины относительно других; во многих

задачах «распознавательная» регрессия (нахождение скалярной или векторной численной характеристики объекта) оказывается именно математической регрессией относительно вектора признаков.

Классификацией называем распознавание качественной (дискретной) характеристики объекта $k \in \{1, \dots, q\}$, где q — число классов, а множество C_k объектов, для которых эта характеристика принимает значение $k = K$, называем K -м классом.

Как мы позднее увидим, ответ распознавателя-классификатора лучше получать не как номер класса k , а как q -мерный вектор $\mathbf{y} = (y_1, \dots, y_q)^T$ «уверенностей» в принадлежности объекта к каждому из классов (часто компонента y_k — это вероятность принадлежности объекта к классу C_k). Тогда, определяя номер класса K по максимальной компоненте выходного вектора классификатора как $K = \arg \max_{k \in \{1, \dots, q\}} y_k$, мы сводим задачу классификации к частному случаю векторной регрессии. Далее мы будем полагать, что при классификации объектов на q классов отклик распознавателя представляет собой вектор $\mathbf{y} \in \mathbb{R}^q$.

Упражнение 1.1. Приведите практические примеры задач регрессии и задач классификации.

Для построения распознавателя необходимо задать *решающее правило*, т.е. функцию f , определяющую отображение пространства векторов признаков (образов) \mathcal{X} в пространство откликов распознавателя \mathcal{Y} . Термины *решающая функция* и *решающее правило* мы будем понимать как синонимы для термина *распознаватель*.

Чаще всего исходные данные для нахождения решающего правила представлены только *обучающей выборкой*, т.е. набором пар $T = \{(\mathbf{x}_n, \mathbf{t}_n)\}_{n=1}^N$, где скаляр или вектор $\mathbf{t}_n \in \mathcal{Y}$ — это желаемый результат распознавания образа $\mathbf{x}_n \in \mathcal{X}$. По обучающей выборке необходимо подобрать решающую функцию f так, чтобы распознавание образов было оптимальным, т.е. «наилучшим» в каком-то смысле (для этого необходимо определить способ оценки качества распознавания). Для функции распознавателя f могут использоваться различные модели, подбор параметров этих моделей (чаще всего итерационный) по выборке $T = \{(\mathbf{x}_n, \mathbf{t}_n)\}_{n=1}^N$ называется *обучением* распознавателя. Такое статистическое обучение распознавателя называют также *машинным обучением*.

Перейдем теперь к формальной постановке задачи построения оптимального статистического распознавателя f по обучающей выборке T . Сначала необходимо задать следующие исходные условия задачи [4].

1. *Пространство векторов признаков* \mathcal{X} , точками которого кодируются распознаваемые объекты (например d -мерное евклидово пространство \mathbb{R}^d).
2. *Пространство ответов* \mathcal{Y} , точками которого кодируются результаты распознавания (например q -мерное пространство \mathbb{R}^q).
3. *Пространство \mathcal{F} распознавателей* $f: \mathcal{X} \rightarrow \mathcal{Y}$. В случае евклидовых пространств \mathcal{X} и \mathcal{Y} , например, непрерывных, дифференцируемых, линейных, полиномиальных и т. п.
4. *Пространство \mathcal{P} распределений* (вероятностных мер) на $\mathcal{X} \times \mathcal{Y}$, которые удовлетворяют каким-либо специфическим для задачи условиям. В случае евклидовых пространств \mathcal{X} и \mathcal{Y} этими условиями могут быть непрерывность функции плотности, ее представимость в виде гауссовых смесей и т. п.
5. *Функция штрафа* $E: \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$ (называемая также функцией ошибки, *потерь*, *риска* и т. п.), как правило, неотрицательная и равная нулю при совпадении прогнозного ответа \mathbf{y} с верным ответом \mathbf{t} . Например, в случае евклидова пространства \mathcal{Y} применяется *квадратичный штраф*

$$E(\mathbf{t}, \mathbf{y}) = \|\mathbf{t} - \mathbf{y}\|^2, \quad (1.1)$$

а в случае дискретного пространства \mathcal{Y} — *бинарный штраф*

$$E(\mathbf{t}, \mathbf{y}) = \begin{cases} 0 & \text{при } \mathbf{t} = \mathbf{y} \\ 1 & \text{при } \mathbf{t} \neq \mathbf{y} \end{cases}. \quad (1.2)$$

6. *Обучающий набор* $T = \{(\mathbf{x}_1, \mathbf{t}_1), \dots, (\mathbf{x}_N, \mathbf{t}_N)\}$, состоящий из пар (вектор признаков, желаемый ответ распознавателя) $(\mathbf{x}_i, \mathbf{t}_i) \in \mathcal{X} \times \mathcal{Y}$, которые считаются значениями независимых случайных векторных величин с одним и тем же неизвестным распределением $\chi \in \mathcal{P}$.

Обозначив для распределения $\chi \in \mathcal{P}$ функцию плотности вероятности как $p_\chi(x, t)$, оптимальным будем считать такой распознаватель $f \in \mathcal{F}$, который при выполнении условий 1–6 доставляет минимум математического ожидания функции штрафа [4]:

$$E_\chi(f) = \mathbb{M}[E(f(\mathbf{x}), \mathbf{t})] = \int_{(\mathbf{x}, \mathbf{t}) \in \mathcal{X} \times \mathcal{Y}} E(f(\mathbf{x}), \mathbf{t}) p_\chi(\mathbf{x}, \mathbf{t}) d\mathbf{x} d\mathbf{t} \rightarrow \min_{f \in \mathcal{F}}. \quad (1.3)$$

Однако в нашем распоряжении имеется только выборка T , взятая из генеральной совокупности с неизвестным распределением $\chi \in \mathcal{P}$. Совершенно очевидно, что при неизвестной плотности распределения вероятностей $p_\chi(\mathbf{x}, \mathbf{t})$ определить минимум в (1.3) невозможно и задача построения оптимального распознавателя f в приведенной выше формулировке неразрешима. По этой причине придется заменить в (1.3) математическое ожидание штрафа его оценкой по выборке T :

$$E_\chi(f) \approx E(f, T) = \frac{1}{N} \sum_{i=1}^N E(f(\mathbf{x}_i), \mathbf{t}_i), \quad (1.4)$$

и вместо (1.3) рассматривать минимизацию *среднего штрафа обучения*, или *эмпирического риска*¹

$$E(f, T) = \frac{1}{N} \sum_{i=1}^N E(f(\mathbf{x}_i), \mathbf{t}_i) \rightarrow \min_{f \in \mathcal{F}}. \quad (1.5)$$

При замене (1.3) на (1.5) возникает вопрос о том, насколько это влияет на итоговые характеристики распознавателя, прежде всего на его *способность к обобщению*, т. е. насколько успешно будет работать распознаватель, когда после обучения ему будут предъявлены новые образы, отсутствовавшие в обучающей выборке T , по которой настраивалось решающее правило f . Сделаем по этому поводу следующие важные замечания.

- Распознаватель, который получен в результате обучения, минимизирующего штраф (1.5), *зависит* от обучающего набора данных T , а приближение (1.4) справедливо для функций f , *не зависящих* от выборки T . В частности, как мы увидим позднее, на обучающем наборе иногда можно добиться даже нулевого среднего штрафа (1.5).
- Математическое ожидание штрафа (1.3) можно оценить более корректно, взяв другой независимый от T набор *тестовых данных* $T' = \{(\mathbf{x}'_1, \mathbf{t}'_1), \dots, (\mathbf{x}'_N, \mathbf{t}'_N)\}$ той же природы (из той же генеральной совокупности), что и выборка T , и посчитав по (1.4) средний штраф тестирования $E(f, T')$. Результат почти наверняка будет хуже: $E(f, T') > E(f, T)$, особенно при большой

¹ Поиск решающей функции $y = f(\mathbf{x})$ минимизацией эмпирического риска (1.5) с квадратичным штрафом (1.1) представляет собой смысловое содержание *метода наименьших квадратов*.

размерности пространства \mathcal{X} и малом количестве обучающих векторов¹.

Отметим, что использование другого тестового набора данных T' , отличного и независимого от обучающего T , является общепринятым подходом при проверке характеристик качества распознавания.

Пример 1.2. Распознавание по ближайшим соседям. Пусть пространство векторов признаков \mathcal{X} — метрическое² (например $\mathcal{X} = \mathbb{R}^d$), пространство откликов \mathcal{Y} — любое; задана обучающая выборка $T = \{(\mathbf{x}_1, \mathbf{t}_1), \dots, (\mathbf{x}_N, \mathbf{t}_N)\}$. Рассмотрим решающее правило *ближайшего соседа* (nearest neighbor — NN): $\mathbf{y} = f(\mathbf{x}) = \mathbf{t}_m$, где индекс m определяется по обучающей выборке T из условия ближайшего для \mathbf{x} соседа в пространстве \mathcal{X} : $\rho(\mathbf{x}, \mathbf{x}_m) = \min_{k=1, \dots, N} \rho(\mathbf{x}, \mathbf{x}_k)$. Ответим на следующие вопросы.

1. Как поступать в случае, если у входного вектора \mathbf{x} в обучающей выборке T имеется два или более ближайших соседа на одинаковом расстоянии?
2. Как обобщить распознавание по ближайшему соседу на распознавание по $K \geq 2$ ближайшим соседям (не обязательно равноудаленным)?

◀ 1. Если пространство ответов \mathcal{Y} — линейное, например $\mathcal{Y} = \mathbb{R}^n$ (векторная регрессия), то ответы ближайших соседей можно усреднить; если пространство ответов \mathcal{Y} — конечное с небольшим числом элементов, например $\{0, 1\}$ (классификация на два класса), то из ответов ближайших соседей можно выбрать самый частый. Возможен также вариант, при котором отклик классификатора выбирается случайным образом из откликов равноудаленных ближайших соседей.

¹ Высокая размерность пространства признаков \mathcal{X} может привести к тому, что необходимый для удовлетворительного обучения распознавателя объем N обучающей выборки оказывается невозможно обеспечить или же обучение становится неприемлемо долгим. Это явление в машинном обучении называют *проклятием размерности* (curse of dimensionality). Очевидно также, что чем больше у модели распознавателя f настраиваемых параметров, тем больше ему требуется данных для обучения.

² Это означает, что в пространстве \mathcal{X} определена *метрика* (расстояние) $\rho(\mathbf{u}, \mathbf{v}) \geq 0$ между любыми элементами $\mathbf{u} \in \mathcal{X}$, $\mathbf{v} \in \mathcal{X}$.

2. Отличие от п. 1 состоит только в том, что K ближайших от распознаваемого вектора x соседей из выборки T не являются в общем случае равноудаленными от x в пространстве \mathcal{X} . В остальных правила формирования отклика можно сохранить (усреднение, выбор наиболее частого ответа, случайный выбор). ►

Укажем на некоторые характерные особенности и свойства распознавателя по K ближайшим соседям (K nearest neighbors — K -NN):

- NN-распознаватель ($K = 1$) не делает ни одной ошибки на предъявленном ему наборе T и может ошибаться только на новых, неизвестных ему векторах признаков;
- K -NN-распознаватель ($K > 1$) не всегда распознает точки обучающего набора T безошибочно, но при небольших K , как правило, меньше ошибается на неизвестных ему векторах по сравнению с вариантом $K = 1$;
- для $K = N$ ближайших соседей, где N — объем обучающей выборки T , распознаватель дает постоянный ответ, не зависящий от входного вектора признаков x ;
- обучение происходит тривиально и состоит в простом запоминании обучающего набора. Распознавание также тривиально, но его сложность растет пропорционально объему обучающей выборки N ;
- при малом количестве обучающих векторов N (когда пространство признаков \mathcal{X} заполнено ими «слабо») метод работает плохо. Поэтому при большой размерности пространства \mathcal{X} может оказаться необходим такой большой объем обучающего набора N , который сделает применение распознавателя K -NN невозможным. Этот эффект является проявлением вышеупомянутой проблемы проклятия размерности.

Упражнение 1.2. Исследуйте на практике характеристики распознавателя K -NN. Для этого, используя какой-либо математический программный пакет, проведите эксперименты по следующему плану.

1. Создайте обучающий набор данных T из $N = 1000$ векторов (x_i, t_i) , где x — наудачу выбранная из отрезка $[0;1]$ величина, $t = \sin(5\pi x/2) + \varepsilon$, а ε — гауссова случайная величина с нулевым математическим ожиданием и СКО = 0,25.

2. Создайте аналогичным образом тестовый набор данных T' из других $N/4 = 250$ векторов (x_i, t_i) .
3. Выбирая различные значения $K \geq 1$ (обязательно рассмотрите случай $K = 1$), определите среднюю ошибку (1.5) (штрафная функция — квадрат разности t_i и отклика распознавателя) для обучающего T и для тестового T' наборов данных. Постройте зависимости средних ошибок от K в виде двух графиков.
4. По графикам, полученным в п. 3, сделайте выводы о характере зависимости эмпирического риска (1.5) от K . Определите значение K , для которого значение ошибки (1.5) получается наименьшим. Сравните значения ошибок, получаемых на обучающем и тестовом наборах данных.

Рассмотренная в этом разделе задача построения оптимального статистического распознавателя f по обучающей выборке $T = \{(\mathbf{x}_1, \mathbf{t}_1), \dots, (\mathbf{x}_N, \mathbf{t}_N)\}$ относится к варианту машинного обучения *с учителем*, когда для каждого вектора признаков \mathbf{x}_i из обучающей выборки известен желаемый отклик распознавателя \mathbf{t}_i . В этом случае говорят также, что обучающая выборка T *размечена*, т.е. каждый вектор-образ \mathbf{x}_i из T заранее помечен «учителем» меткой \mathbf{t}_i . Каждую пару $(\mathbf{x}_i, \mathbf{t}_i) \in T$ называют также *прецедентом*, а обучение с учителем — обучением по прецедентам.

На практике возникают также задачи, в которых имеется только выборка из векторов признаков $\{\mathbf{x}_i\}_{i=1}^N$, а их метки \mathbf{t}_i отсутствуют (неизвестны). В этом случае распознаватель должен осуществлять группировку данных по классам на основе обучающей выборки $\{\mathbf{x}_i\}_{i=1}^N$ без внешнего участия учителя [19, 22]. Вопросы, связанные с машинным обучением *без учителя*, будут рассмотрены нами в главе 6.

1.2. Способность распознавателя к обобщению. Переобучение и регуляризация

Рассмотрим простейший случай регрессии, когда образ объекта является скаляром x (вектор признаков состоит из одной компоненты), а желаемый отклик распознавателя, или «метка»

объекта, — это также скалярная величина t , которая присвоена ему по следующему правилу:

$$t = g(x) + \varepsilon,$$

где $g(x)$ — некоторая неизвестная распознавателю функция; ε — случайная величина с математическим ожиданием $M[\varepsilon] = 0$ и дисперсией σ^2 . Требуется построить оптимальный распознаватель f , дающий минимальное среднее значение функции штрафа $E(t, y) = (t - y)^2$, где $y = f(x)$.

Очевидно, что средний по всей генеральной совокупности штраф (1.1) будет минимальным, если для решающего правила $y = f(x)$ выбрать $f(x) = g(x)$, так как $\forall x$:

$$\begin{aligned} M[(t - y)^2] &= M[(g(x) + \varepsilon - f(x))^2] = \\ &= \underbrace{(g(x) - f(x))^2}_{\geq 0} + 2 \underbrace{M[\varepsilon]}_0 (g(x) - f(x)) + \underbrace{M[\varepsilon^2]}_{\sigma^2} = \underbrace{(g(x) - f(x))^2}_{0, \text{ при } g(x)=f(x)} + \sigma^2. \end{aligned}$$

Однако функция $g(x)$ распознавателю неизвестна, поэтому необходимо построить ее аппроксимацию $f(x) \approx g(x)$ путем обучения на выборке $T = \{(x_1, t_1), \dots, (x_N, t_N)\}$, минимизируя эмпирический риск (1.4) с функцией ошибки $E(t, y) = (t - y)^2$.

Пример 1.3. Как и в упражнении 1.2, наблюдаемые значения меток будем формировать по правилу

$$t = \sin(5\pi x/2) + \varepsilon, \tag{1.6}$$

где ε — гауссова случайная величина с нулевым средним и СКО = 0,25. Решающую функцию $y = f(x)$ будем искать в пространстве \mathcal{F} алгебраических многочленов степени M , т.е. будем использовать для распознавателя модель вида

$$y = f(x) = \sum_{j=0}^M w_j x^j. \tag{1.7}$$

Таким образом, по наблюдаемой выборке $T = \{(x_1, t_1), \dots, (x_N, t_N)\}$ необходимо построить аппроксимацию $f(x) \approx \sin(5\pi x/2)$ в виде многочлена (1.7). Сделаем это для частного случая $N = 11$ при некоторых значениях M и изучим способность полученных распознавателей к обобщению, которая характеризуется малостью ошибок при распознавании образов, отсутствовавших в обучающей выборке.

◀ Очевидно, что при $M \geq N - 1$ можно подобрать такие коэффициенты полинома (1.7), что для всех элементов обучающей выборки T

получим $y_i = f(x_i) = t_i$ и, соответственно, нулевое значение для эмпирического риска (1.4).

Для порядков $M < N - 1$ вектор коэффициентов $\mathbf{w} = (w_0, \dots, w_M)^T$ полинома (1.7) будем искать, минимизируя по параметрам w_j квадратичную функцию (1.4) или (что даст тот же самый результат при фиксированном значении N) функцию

$$E(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^N (t_i - f(x_i))^2, \quad (1.8)$$

где множитель $1/2$ включают в выражение для удобства его последующего дифференцирования.

Выбрав $x_i = (i-1)/10$ ($i = 1, \dots, 11$) и сгенерировав признаки t_i по правилу (1.6), получим на плоскости (x, t) некоторые точки, случайным образом отклоняющиеся по оси t от графика функции $\sin(5\pi x/2)$ на значения реализации нормальной величины ϵ (рис. 1.2). Эти точки будем использовать в качестве обучающей выборки $T = \{(x_1, t_1), \dots, (x_{11}, t_{11})\}$.

Методом наименьших квадратов найдем на этой выборке коэффициенты $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$ полиномов (1.7) порядка $M = 0, 1, \dots, 10$, которые обращают в минимум штраф (1.8). Графики полученных полиномов (1.7) для $M = 1, 3, 6, 10$ приведены на рис. 1.3.

Как и следовало ожидать, полином-прямая ($M = 1$) дает плохое приближение функции $\sin(5\pi x/2)$. Хотя полином десятого порядка и обеспечивает нулевую ошибку приближения в точках заданной обучающей последовательности T (и только в них),

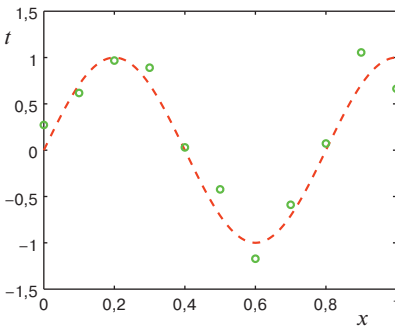


Рис. 1.2. График функции $\sin(5\pi x/2)$ (штриховая кривая) и точки обучающей выборки (11 шт.)

полином шестого порядка ($M = 6$) дает визуально лучшее приближение. Таким образом, анализ графиков рис. 1.3 позволяет предположить, что увеличение порядка аппроксимирующего алгебраического полинома сначала повышает точность аппроксимации «скрытой» функции $g(x)$ [в нашем примере $g(x) = \sin(5\pi x/2)$], а затем точность приближения ухудшается.

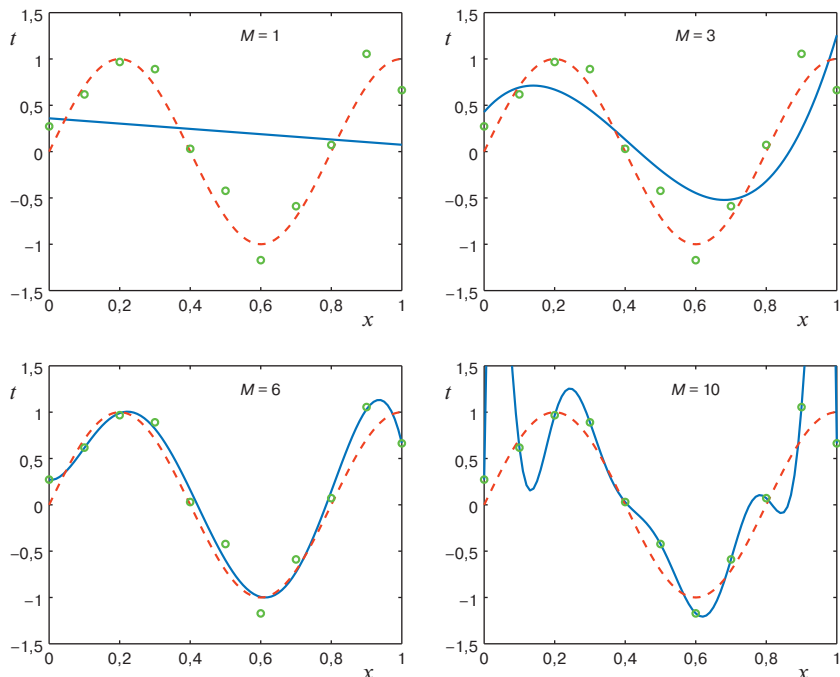


Рис. 1.3. Полиномы наилучшего (для заданной выборки) квадратичного приближения порядка M . При $M=10$ ошибка приближения равна нулю

Для изучения способности распознавателя к обобщению исследуем среднюю ошибку распознавания на тестовой выборке T' , полученной по тому же правилу (1.6), но не содержащей данные, которые были включены в обучающую последовательность. Для того чтобы можно было сравнивать выборки разного объема, от (1.8) вернемся к измерению среднего эмпирического риска (1.4). На рис. 1.4 приведены значения среднеквадратичной ошибки распознавания

$$E_{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^N (t_n - f(x_n))^2}$$

[квадратного корня из (1.4)], полученные на рассмотренной обучающей выборке T из $N = 11$ элементов, а также на независимо сгенерированной по правилу (1.6) тестовой выборке $T' = \{(x_1, t_1), \dots, (x_{101}, t_{101})\}$

($N = 101$), для которой была взята сетка значений $x_i = (i - 1)/100$ ($i = 1, \dots, 101$).

Из рис. 1.4 видно, что на тестовой выборке среднеквадратичные ошибки распознавания принимают наименьшие и примерно одинаковые значения для полиномов, имеющих порядок $4 \leq M \leq 9$. Причем при одном и том же порядке M ошибка на тестовой выборке всегда больше ошибки на обучающей выборке.

При $M = 10$, когда ошибку на обучающей выборке становится возможным обратить в ноль, ошибка на тестовой выборке резко возрастает. Если посмотреть на график соответствующего полинома, у которого появляются большие осцилляции относительно графика функции $\sin(5\pi x/2)$ (см. рис. 1.3), то это не покажется удивительным. Как можно видеть из табл. 1.1, рост осцилляций полинома десятого порядка сопровождается значительным увеличением абсолютных величин коэффициентов $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$.

Эффект резкого увеличения средней ошибки на тестовой выборке при увеличении количества варьируемых параметров модели распознавателя (т.е. при увеличении сложности модели, которая в нашем примере характеризуется значением M) называется *переобучением* распознавателя. Термин «переобучение» понимается как чрезмерное, а не повторное обучение (overlearning, или

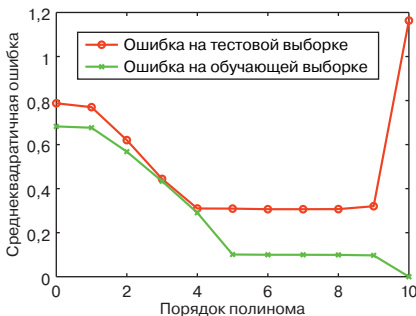


Рис. 1.4. Среднеквадратичная ошибка распознавания E_{RMS} , полученная на обучающей выборке из 11 элементов и тестовой выборке из 101 элемента для моделей полиномов (1.7) разных порядков $M = 0, 1, \dots, 10$

overfitting, — чрезмерная подгонка под обучающие данные).

При большом количестве параметров модели, используемой для решающей функции f , и малом объеме обучающей выборки часто оказывается возможным настроить модель так, что распознаватель как бы «запоминает» обучающие данные и дает для них вообще нулевую ошибку (мы это наблюдали в примере 1.2, где запоминание данных происходило в буквальном смысле). Вместе с тем каждый

Таблица 1.1. Коэффициенты полинома (1.7) порядка M , полученные по обучающей выборке объема $N=11$ методом наименьших квадратов

	$M=1$	$M=3$	$M=6$	$M=10$
w_0^*	0,3596	0,4272	0,2725	0,2725
w_1^*	-0,2856	4,401	-0,9053	279,38
w_2^*	0	-19,05	73,14	-7524
w_3^*	0	15,47	350,78	81700
w_4^*	0	0	547,67	-472947
w_5^*	0	0	-312,68	1632833
w_6^*	0	0	43,95	-3530061
w_7^*	0	0	0	4822552
w_8^*	0	0	0	-4042083
w_9^*	0	0	0	1896384
w_{10}^*	0	0	0	-381133

прецедент в обучающей выборке предполагает наличие некоторого случайного фактора ошибки, содержащегося в присвоенной учителем метке [как и в нашем примере, см. (1.6)], поэтому точное запоминание прецедентов при обучении распознавателя лишено смысла. Переобучение распознавателя влечет за собой ухудшение его способности к обобщению, т. е. к распознаванию «незнакомых» новых образов с малой средней ошибкой.

Для предотвращения переобучения необходимо, чтобы количество настраиваемых независимых параметров модели распознавателя было намного меньше объема обучающей выборки. Посмотрим, как для нашего примера с моделью решающей функции (1.7) и правилом разметки (1.6) меняется способность распознавателя к обобщению при увеличении объема обучающей выборки N . Графики решающей функции $f(x)$ (полиномов наилучшего среднеквадратичного приближения десятого порядка, полученных на обучающих выборках объемов $N=21$ и $N=101$) приведены на рис. 1.5. Как и следовало ожидать, точность приближения функции $\sin(5\pi x/2)$ полиномом (1.7) повышается с увеличением объема обучающей выборки.

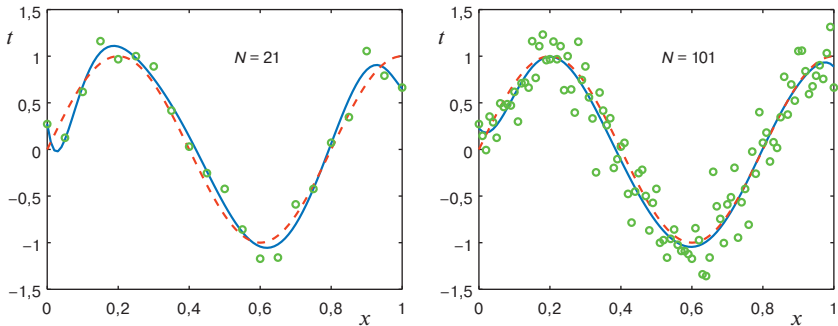


Рис. 1.5. Полиномы (1.7) наилучшего квадратичного приближения порядка $M=10$ (сплошная кривая), полученные на выборках объема $N=21$ (слева) и $N=101$ (справа). Сравните с рис. 1.3

К сожалению, объем N реально имеющейся для обучения распознавателя выборки часто оказывается недостаточным. В этом случае для исключения эффекта переобучения применяется *регуляризация*, которая состоит в добавлении в функцию эмпирического риска некоторого неотрицательного слагаемого, дополнительно штрафующего за каждый выбор того или иного вектора параметров \mathbf{w} в модели решающей функции.

Большие абсолютные значения весов $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$ в модели (1.7) нежелательны и должны штрафоваться сильнее, так как увеличивают осцилляцию решающей функции. Для рассматриваемого примера модели (1.7) введем дополнительное слагаемое регуляризации, например квадрат нормы вектора параметров¹ $\|\mathbf{w}\|^2 = \mathbf{w}^T \mathbf{w} = \sum_{k=0}^M w_k^2$. Тогда новая функция штрафа, полученная в результате регуляризации, будет выглядеть для решающего правила $f(x)$ (1.7) следующим образом:

$$E(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^N (t_i - f(x_i))^2 + \frac{\lambda}{2} \|\mathbf{w}\|^2, \quad (1.9)$$

где параметр $\lambda \geq 0$ задает баланс между «значимостями» штрафа за ошибку распознавания и штрафа за величину абсолютных значений весов модели (1.7). Как и ранее, дополнительный множитель

¹ Поскольку коэффициент w_0 влияет лишь на смещение по оси абсцисс функции (1.7) и не влияет на «размах» ее осцилляций, параметр w_0 часто не учитывается при подсчете квадрата нормы вектора \mathbf{w} .

$1/2$ в слагаемом регуляризации введен для удобства последующего дифференцирования; конкретное значение веса λ определяется экспериментально.

На рис. 1.6 приведены графики полиномов (1.7) десятого порядка, полученных на исходной (см. рис. 1.2) выборке из $N = 11$ элементов в результате минимизации функции (1.9) для значений $\lambda = 10^{-10}$ и $\lambda = 1$. Заметим, что случаю $\lambda = 0$ (отсутствие регуляризации) будет соответствовать правый нижний график на рис. 1.3. Видим, что при $\lambda = 10^{-10}$ (рис. 1.6, график слева) полученная решающая функция $f(x)$ приближает функцию $\sin(5\pi x/2)$ намного лучше, чем в случае отсутствия регуляризации, но все же хуже той решающей функции, которая была найдена по обучающей выборке объема $N = 101$ (правый график на рис. 1.5).

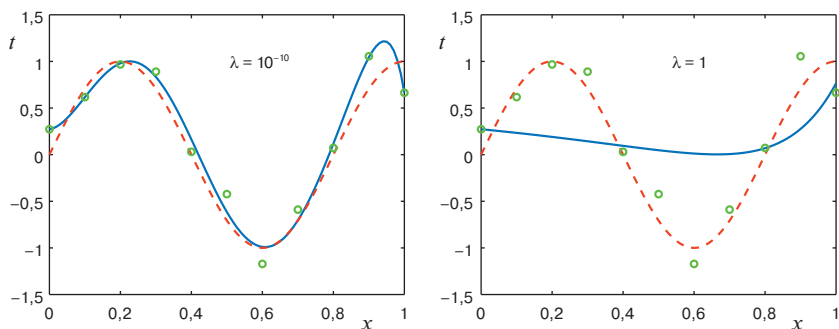


Рис. 1.6. Графики решающих функций (1.7) порядка $M = 10$, полученных в результате минимизации штрафа (1.9) со слагаемым регуляризации

Значения весовых коэффициентов $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$, полученных при использовании регуляризации, представлены в табл. 1.2. Как и следовало ожидать, введение слагаемого регуляризации уменьшило абсолютные величины коэффициентов полинома (1.7).

На рис. 1.7 приведены зависимости среднеквадратичной ошибки распознавания E_{RMS} от значения параметра λ из (1.9), полученные на тех же обучающей T (11 элементов, см. рис. 1.2) и тестовой T' (101 элемент) выборках, которые использовались для построения графиков рис. 1.4. Видим, что выбор $\lambda \approx 10^{-8}$ позволяет получить для решающей функции полинома порядка $M = 10$ примерно

Таблица 1.2. Коэффициенты полинома (1.7) порядка $M=10$, полученные по обучающей выборке объема $N=11$ при использовании регуляризации

	$\lambda = 0$	$\lambda = 10^{-10}$	$\lambda = 1$
w_0^*	0,2725	0,2723	0,2728
w_1^*	279,38	0,5698	-0,3686
w_2^*	-7524	44,9265	-0,2124
w_3^*	81 700	-169,4394	-0,0281
w_4^*	-472 947	65,4488	0,0908
w_5^*	1 632 833	82,4711	0,1542
w_6^*	-3 530 061	466,8941	0,1812
w_7^*	4 822 552	-577,2536	0,1862
w_8^*	-4 042 083	-688,525	0,1785
w_9^*	1 896 384	1 284,8539	0,1641
w_{10}^*	-381 133	-509,5763	0,1463

такую же ошибку распознавания на тестовой выборке, которая имела место ранее без применения регуляризации для лучшего варианта полинома порядка $M=6$.

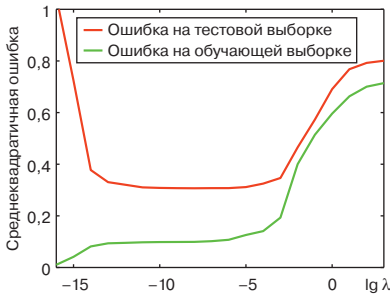


Рис. 1.7. Среднеквадратичная ошибка распознавания E_{RMS} , полученная для разных значений параметра λ функции штрафа (1.9) на обучающей выборке из 11 элементов и тестовой выборке из 101 элемента для полинома (1.7) порядка $M=10$

В рассмотренном нами примере модели распознавателя (1.7) использование сложных решающих функций (с порядком полинома $M > 4$) при обучающей выборке объема $N=11$ оказалось практически лишеным смысла, так как крайне незначительно влияло на изменение ошибки распознавания. Однако в общем случае применение более сложных моделей с большим количеством параметров способно потенциально понизить ошибку распознавания. При этом возможно возникновение проблем

обучения сложных моделей из-за необходимости использования нереально больших объемов обучающих выборок. Эффективным средством повышения качества обучения на выборках недостаточного объема является регуляризация, другие примеры и варианты которой встретятся нам в следующих главах. ►

Удобным свойством часто используемой функции штрафа (1.8) с полиномиальной моделью (1.7) является ее *выпуклость*, вследствие чего равенство нулю градиента $\nabla E(\mathbf{w}) = \mathbf{0}$ является не только необходимым, но и достаточным условием минимума этой функции.

Напомним, что функция $F(\mathbf{x})$ является выпуклой на выпуклой области D , если $\forall \mathbf{x}_1 \in D, \forall \mathbf{x}_2 \in D$:

$$F(\alpha \mathbf{x}_1 + (1 - \alpha) \mathbf{x}_2) \leq \alpha F(\mathbf{x}_1) + (1 - \alpha) F(\mathbf{x}_2)$$

при любых значениях $\alpha \in [0; 1]$.

Пример 1.4. Показать, что функция $E(\mathbf{w})$ в (1.8) выпуклая.

◀ Подставим выражение для $f(x)$ из (1.7) в (1.8):

$$E(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^N \left(t_i - \sum_{j=0}^M w_j x_i^j \right)^2 = \sum_{i=1}^N E_i(\mathbf{w}), \quad \text{где} \quad E_i(\mathbf{w}) = \frac{1}{2} \left(t_i - \sum_{j=0}^M w_j x_i^j \right)^2.$$

Для частных производных квадратичной функции $E_i(\mathbf{w})$ имеем:

$$\frac{\partial}{\partial w_k} E_i(\mathbf{w}) = \frac{1}{2} \frac{\partial}{\partial w_k} \left(t_i - \sum_{j=0}^M w_j x_i^j \right)^2 = (-x_i^k) \left(t_i - \sum_{j=0}^M w_j x_i^j \right), \quad k = 0, 1, \dots, M;$$

$$\frac{\partial^2}{\partial w_k \partial w_m} E_i(\mathbf{w}) = (-x_i^k) (-x_i^m) = x_i^{k+m}, \quad k = 0, 1, \dots, M; \quad m = 0, 1, \dots, M.$$

Для выпуклости квадратичной функции $E_i(\mathbf{w})$ необходимо и достаточно, чтобы матрица вторых производных (матрица Гессе)

$$\mathbf{H}_i = \left(\frac{\partial^2}{\partial w_k \partial w_m} E_i(\mathbf{w}) \right)_{k,m=0}^M = \left(x_i^{k+m} \right)_{k,m=0}^M$$

была положительно полуопределенной. Поскольку для любого вектора $\mathbf{y} = (y_0, y_1, \dots, y_M)^T$ квадратичная форма

$$\mathbf{y}^T \mathbf{H}_i \mathbf{y} = \sum_{k=0}^M \sum_{m=0}^M x_i^{k+m} y_k y_m = \sum_{k=0}^M x_i^k y_k \sum_{m=0}^M x_i^m y_m = \left(\sum_{n=0}^M x_i^n y_n \right)^2 \geq 0,$$

то матрица \mathbf{H}_i положительно полуопределена, а функция $E_i(\mathbf{w})$ выпуклая. Функция $E(\mathbf{w})$ (1.8) является выпуклой как сумма выпуклых функций $E_i(\mathbf{w})$. ▶

Упражнение 1.3. Покажите, что при использовании модели (1.7) в функции штрафа (1.8) выражения для оптимальных коэффициентов $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$, обращающих в минимум (1.8), находятся по обучающей выборке $T = \{(x_1, t_1), \dots, (x_N, t_N)\}$ в результате решения следующей системы линейных уравнений:

$$\sum_{j=0}^M a_{i,j} w_j = b_i, \quad i = 0, \dots, M, \tag{1.10}$$

где
$$a_{i,j} = \sum_{n=1}^N x_n^{i+j}, \quad b_i = \sum_{n=1}^N t_n x_n^i.$$

Убедитесь, что с использованием обозначений

$$\mathbf{X} = \begin{pmatrix} x_1^0 & x_1^1 & \dots & x_1^M \\ x_2^0 & x_2^1 & \dots & x_2^M \\ \vdots & \vdots & \ddots & \vdots \\ x_N^0 & x_N^1 & \dots & x_N^M \end{pmatrix} = \begin{pmatrix} 1 & x_1 & \dots & x_1^M \\ 1 & x_2 & \dots & x_2^M \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & \dots & x_N^M \end{pmatrix}, \quad \mathbf{t} = \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_N \end{pmatrix}$$

систему уравнений (1.10) можно представить в матричном виде следующим образом: $\mathbf{A}\mathbf{w} = \mathbf{b}$, где $\mathbf{A} = \mathbf{X}^T \mathbf{X}$ и $\mathbf{b} = \mathbf{X}^T \mathbf{t}$. Причем в обозначениях примера 1.4 $\mathbf{A} = \sum_{i=1}^N \mathbf{H}_i$, поэтому матрица \mathbf{A} симметрическая и положительно полуопределена.

Оказывается, что рассмотренную в примере 1.3 процедуру регуляризации, состоящую во включении в выражение для штрафной функции ошибки распознавания дополнительного слагаемого штрафа за выбор конкретных параметров модели, можно интерпретировать как определенную модификацию системы уравнений (1.10), направленную на повышение устойчивости ее решения¹. Поясним это утверждение.

Если матричное уравнение вида $\mathbf{A}\mathbf{w} = \mathbf{b}$ имеет *плохо обусловленную* матрицу \mathbf{A} , т.е. число обусловленности $\mu(\mathbf{A}) \gg 1$, то решение

¹ Общая теория метода регуляризации, используемого для решения некорректных задач, приводящих к плохо обусловленным системам линейных уравнений, разработана академиком А. Н. Тихоновым и его школой в 60-х гг. XX в.

уравнения $\mathbf{w} = \mathbf{A}^{-1}\mathbf{b}$ является неустойчивым¹. Для симметрических положительно полуопределенных матриц число обусловленности выражается через собственные числа (которые являются неотрицательными) как $\mu(\mathbf{A}) = \lambda_{\max}/\lambda_{\min}$, поэтому у плохо обусловленных матриц $\lambda_{\max} \gg \lambda_{\min}$. В частности, для вырожденной матрицы $\lambda_{\min} = 0$ и $\mu(\mathbf{A}) = \infty$.

Если «поправить» симметрическую положительно полуопределенную матрицу следующим образом: $\mathbf{A}' = \mathbf{A} + \lambda\mathbf{E}$, где скаляр $\lambda > 0$, \mathbf{E} — единичная матрица, то собственные векторы у новой матрицы \mathbf{A}' останутся теми же, а собственные числа станут больше на величину λ . Действительно, обозначив для матрицы \mathbf{A} некоторый собственный вектор и соответствующее ему собственное число как \mathbf{r}_k и λ_k , для матрицы \mathbf{A}' получаем: $\mathbf{A}'\mathbf{r}_k = (\lambda_k + \lambda)\mathbf{r}_k$. По этой причине для новой (также симметрической и положительно полуопределенной) матрицы число обусловленности уменьшится:

$$\mu(\mathbf{A}') = \frac{\lambda_{\max} + \lambda}{\lambda_{\min} + \lambda} < \frac{\lambda_{\max}}{\lambda_{\min}} = \mu(\mathbf{A}),$$

и устойчивость решения $\mathbf{w} = (\mathbf{A}')^{-1}\mathbf{b}$ повысится. Добавление слагаемого регуляризации в (1.9) приводит к подобной модификации системы уравнений (1.10).

Упражнение 1.4. Убедитесь, что функция (1.9) является выпуклой и покажите, что замена штрафа (1.8) на (1.9) приводит к оптимальным параметрам $\mathbf{w}^* = (w_0^*, \dots, w_M^*)^T$ модели (1.7), определяемым системой линейных уравнений $(\mathbf{A} + \lambda\mathbf{E})\mathbf{w} = \mathbf{b}$, которая получена из $\mathbf{A}\mathbf{w} = \mathbf{b}$ (1.10).

В примере 1.3 полиномиальная модель использовалась для решающей функции распознавателя $y = f(x)$ в простейшем случае, когда вектор признаков \mathbf{x} был однокомпонентным, т.е. скаляром ($\mathbf{x} = x$). Если размерность пространства признаков $\dim \mathcal{X} = d > 1$, то количество параметров, задающих алгебраический полином степени M , будет заметно больше, чем $M + 1$. Чему оно равно?

¹ Такая ситуация имеет место, в частности, для системы линейных уравнений (1.10), которая была получена в примере 1.3 на выборке объема $N = 11$ с полиномом (1.7) порядка $M = 10$.

Пример 1.5. Подсчитать количество параметров, задающих алгебраический полином степени M от d переменных x_1, \dots, x_d .

◀ Рассматриваемый алгебраический полином имеет следующий общий вид:

$$P(x_1, \dots, x_d) = w_0 + \sum_{i=1}^d w_i x_i + \sum_{i_1=1}^d \sum_{i_2=1}^d w_{i_1, i_2} x_{i_1} x_{i_2} + \dots \\ \dots + \sum_{i_1=1}^d \sum_{i_2=1}^d \dots \sum_{i_M=1}^d w_{i_1, i_2, \dots, i_M} x_{i_1} x_{i_2} \dots x_{i_M}.$$

Постоянная составляющая полинома определяется одним параметром w_0 . В первой сумме, определяющей линейную часть полинома, имеем $s_1(d) = d$ параметров w_1, \dots, w_d .

В двойной сумме, определяющей квадратичную форму, пара коэффициентов $w_{i,j}$ и $w_{j,i}$ при $i \neq j$ соответствует подобным слагаемым и поэтому определяет только один параметр $a_{i,j}$ в квадратичной части полинома: $w_{i,j}x_i x_j + w_{j,i}x_j x_i = a_{i,j} x_i x_j$; обычно для симметрии принимают $w_{i,j} = w_{j,i} = a_{i,j}/2$. Тогда количество независимых параметров $a_{i,j}$ в квадратичной части полинома:

$$s_2(d) = \frac{|\{(i, j) | i \neq j\}|}{2} + |\{(i, j) | i = j\}| = \frac{d^2 - d}{2} + d = \frac{d(d+1)}{2},$$

где обозначение $|A|$ принято для количества элементов в множестве A .

Обобщим способ подсчета параметров, использованный выше для квадратичной части полинома. Для того чтобы определить количество s_k коэффициентов-параметров в k -кратной сумме

$$\sum_{i_1=1}^d \sum_{i_2=1}^d \dots \sum_{i_k=1}^d w_{i_1, i_2, \dots, i_k} x_{i_1} x_{i_2} \dots x_{i_k},$$

которые останутся после приведения подобных слагаемых, необходимо подсчитать количество различных комбинаций (i_1, i_2, \dots, i_k) , не различающихся порядком следования индексов, но с возможностью их повторений. Воспользуемся формулой для количества сочетаний с повторениями из d по k :

$$s_k(d) = \tilde{C}_d^k = C_{d-1+k}^k = \frac{(d-1+k)!}{(d-1)!k!}.$$

Заметим, что эта формула является верной и для уже найденных нами значений $s_0(d) = 1$, $s_1(d) = d$, $s_2(d) = d(d+1)/2$.

В итоге количество параметров полинома порядка M от d переменных получаем равным

$$S_M(d) = \sum_{k=0}^M s_k(d) = \sum_{k=0}^M \frac{(d-1+k)!}{(d-1)!k!} = S_{M-1}(d) + \frac{(d-1+M)!}{(d-1)!M!} = \frac{(d+M)!}{d!M!}.$$

Последнее равенство обоснуйте самостоятельно методом математической индукции, обратив сначала внимание на то, что итоговое выражение верно для $S_0(d)$ и $S_1(d)$. Тогда в предположении его применимости для $S_{M-1}(d)$ нужно показать справедливость полученного выражения для $S_M(d)$. ►

Таким образом, количество числовых параметров, определяющих алгебраический полином порядка M от d переменных, равно

$$S_M(d) = \frac{(d+M)!}{d!M!}. \quad (1.11)$$

Упражнение 1.5. Покажите, что порядок величины (1.11) $S_M(d) = O(d^M)$ при $d \gg M$ и $S_M(d) = O(M^d)$ при $M \gg d$, воспользовавшись формулой Стирлинга $n! \approx \sqrt{2\pi n} e^{-n} n^n$, применяемой для аппроксимации факториала при больших значениях n .

Рост числа параметров полиномиальной модели, наблюдаемый при увеличении размерности пространства векторов признаков, представляет собой пример проклятия размерности.

1.3. Применение формулы Байеса для оценок параметров моделей

Напомним понятие условной вероятности. Пусть A и B — некоторые случайные события. Условной вероятностью события B при условии, что произошло событие A , называется величина

$$P(B|A) = \frac{P(AB)}{P(A)},$$

где $P(AB)$ — вероятность совместной реализации событий A и B ; $P(A)$ — вероятность события A .

Формула Байеса получается непосредственно из приведенного определения условной вероятности:

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)},$$

и позволяет «уточнить» априорную вероятность $P(B)$ события B , если имеется дополнительная апостериорная информация о том, что произошло событие A .

Упражнение 1.6. Используя формулу Байеса, решите следующую задачу. В студенческой группе 3 человека имеют высокий уровень подготовки, 19 человек — средний и 3 — низкий. Вероятности успешной сдачи экзамена для данных студентов соответственно равны 0,95; 0,7 и 0,4. Известно, что некоторый студент сдал экзамен. Какова вероятность того, что он:

- был подготовлен очень хорошо;
- был подготовлен средне;
- был подготовлен плохо?

В машинном обучении формулу Байеса часто называют *теоремой Байеса*; мы будем использовать следующую форму ее записи и терминологию [5]:

$$p(\theta | D) = \frac{p(\theta)p(D | \theta)}{p(D)}, \quad (1.12)$$

где $p(\theta | D)$ — апостериорная вероятность (posterior probability),
 $p(\theta)$ — априорная вероятность (prior probability),
 $p(D | \theta)$ — правдоподобие (likelihood),
 $p(D) = \int p(D | \theta)p(\theta)d\theta$ — вероятность данных (evidence).

В зависимости от того, непрерывным или дискретным является аргумент функций $p(\dots)$ в формуле (1.12), они могут обозначать как вероятности, так и плотности вероятностей.

Часто по наблюдаемым данным D необходимо найти параметры θ некоторой выбранной модели распознавателя. В классической статистике оптимальный вектор параметров находится как гипотеза *максимального правдоподобия* (МП, maximum likelihood — ML):

$$\theta_{ML} = \underset{\theta}{\arg \max} p(D | \theta). \quad (1.13)$$

В машинном обучении чаще применяется более общий байесовский подход, когда рассматривается апостериорное (для известных наблюдаемых данных D) распределение вектора параметров с плотностью

$$p(\theta|D) \propto p(\theta)p(D|\theta),$$

где знак \propto означает «пропорционально» и заменяет знак равенства в (1.12) после устранения фиксированного для наблюдаемых данных D нормировочного множителя $1/p(D)$. Тогда оптимальный вектор параметров θ в смысле *максимальной апостериорной* (МА) вероятности (maximum a posteriori probability — MAP) находится как

$$\theta_{MAP} = \arg \max_{\theta} p(\theta|D) = \arg \max_{\theta} p(\theta)p(D|\theta). \quad (1.14)$$

Правило (1.14) является более общим, чем (1.13), и дает тот же результат, что и (1.13), если положить $p(\theta) = \text{const}$ (равномерное распределение).

Пример 1.6. Подбрасывается монета, для которой вероятность выпадения решки неизвестна и равна q (монета может быть «неправильной», поэтому не обязательно $q = 1/2$). Построим МП- и МА-оценки для параметра $\theta = q$.

◀ Пусть после N подбрасываний решка выпала n раз. Вероятность такого события D равна (это классический пример *схемы испытаний Бернулли*)

$$P(D|q) = \frac{N!}{n!(N-n)!} q^n (1-q)^{N-n}.$$

Вместо поиска максимума правдоподобия по (1.13) в данном случае удобнее искать максимум логарифма правдоподобия

$$L(D|q) = \ln P(D|q) = \ln \left(\frac{N!}{n!(N-n)!} \right) + n \ln q + (N-n) \ln(1-q).$$

Отсюда, удаляя из полученного выражения первое слагаемое, не зависящее от q , находим МП-оценку (убедитесь):

$$q_{ML} = \arg \max_q (n \ln q + (N-n) \ln(1-q)) = \frac{n}{N}.$$

Полученная МП-оценка представляет собой относительную частоту выпадений решки и при $N \rightarrow \infty$ сходится по вероятности к параметру q . Но вряд ли можно назвать «оценкой» результат, который даст однократное подбрасывание монеты: 0 при выпадении орла или 1 при выпадении решки.

Вместе с тем до начала подбрасывания монеты мы уже можем предполагать что-то о возможном значении параметра q и с той

или иной степенью точности смоделировать эти предположения с помощью априорной плотности вероятности $p(q)$. Тогда мы можем получить следующую МА-оценку:

$$q_{MAP} = \arg \max_q \ln(p(q)p(D|q)) = \arg \max_q (\ln p(q) + n \ln q + (N - n) \ln(1 - q)).$$

Если положить $p(q) = 1$ при $0 \leq q \leq 1$ (равномерное распределение параметра q), то это будет означать, что фактически никаких «предпочтений» для параметра q не делается; тогда получим, что $q_{MAP} = q_{ML}$.

Если же предположить, что значение параметра q более вероятно при $q \approx 1/2$ и маловероятно у крайних значений q (0 и 1), то для модели априорной плотности $p(q)$ можно выбрать, например, гауссов закон с математическим ожиданием $m_q = 1/2$ и некоторой дисперсией σ^2 . Тогда МА-оценка будет иметь вид

$$\begin{aligned} q_{MAP} &= \arg \max_q \left(-\frac{(q - 0,5)^2}{2\sigma^2} + n \ln q + (N - n) \ln(1 - q) \right) = \\ &= \arg \min_q \left(-n \ln q - (N - n) \ln(1 - q) + \beta (q - 0,5)^2 \right), \end{aligned}$$

где множитель $\beta = 1/(2\sigma^2) > 0$. Масштабируя функцию, минимум которой необходимо найти для получения МА-оценки, получим

$$q_{MAP} = \arg \min_q \left(\underbrace{-\frac{n}{N} \ln q - \left(1 - \frac{n}{N}\right) \ln(1 - q)}_{f_1(q)} + \frac{\beta}{N} \underbrace{(q - 0,5)^2}_{f_2(q)} \right). \quad (1.15)$$

Минимум введенной в (1.15) функции $f_1(q)$, как несложно установить, соответствует МП-оценке

$$q_{ML} = \arg \min_q \left(f_1(q) = -\frac{n}{N} \ln q - \left(1 - \frac{n}{N}\right) \ln(1 - q) \right) = \frac{n}{N},$$

а минимум функции $f_2(q)$ лежит в точке $q = 1/2$. При $\beta = 0$ «предпочтения» в оценке параметра q в соответствии с (1.15) отсутствуют и $q_{MAP} = q_{ML}$. Увеличивая параметр $\beta > 0$, мы уменьшаем дисперсию в модели плотности априорного распределения $p(q)$ и повышаем значимость штрафа $f_2(q)$ за отклонение МА-оценки от значения $q = 1/2$. Однако роль априорного распределения снижается по мере увеличения числа подбрасываний монеты: весовой коэффициент β/N при функции $f_2(q)$ в (1.15) стремится к нулю при $N \rightarrow \infty$, поэтому в пределе получаем $q_{MAP} = q_{ML} = p$.