

Введение

Актуальность темы диссертации. Проблема распознавания образов сохраняет свою актуальность уже без малого полвека. Несмотря на относительную новизну проблемы, ей посвящено огромное число научных статей и монографий.

Процесс распознавания образов состоит в том, что система распознавания, как правило, компьютер, анализируя предъявленный сигнал или изображение, характеризующие некоторый недоступный для непосредственного наблюдения объект или явление, принимает решение о принадлежности этого скрытого объекта к одному из конечного множества классов [19,30]. Пожалуй, все, по крайней мере, слышали о системах распознавания речевых команд и даже слитной речи, отпечатков пальцев, анализа электрокардиограмм, а что касается программ автоматического чтения печатного текста, то многие уже активно ими пользуются.

Правило, которое каждому объекту, точнее, представляющему его сигналу или изображению, ставит в соответствие определенное наименование класса, называют решающим правилом. Ставшая уже классической теория распознавания образов разрабатывает методы построения формальных решающих правил в предположении, что на анализируемом сигнале уже измерены значения так называемых полезных признаков, или, как принято говорить, он представлен в виде точки в признаковом пространстве. Вектор признаков каждого объекта характеризует некоторый фиксированный набор присущих ему свойств, существенных для распознавания скрытого класса и представляемых обычно в виде действительных чисел, например, температура, размер, вес и т.п.

Основным принципом формирования решающего правила распознавания является, так называемое, обучение по прецедентам или, как еще говорят, «с учителем». Его сущность заключается в анализе относительно небольшой обучающей выборки объектов распознавания, на которой «учитель», то есть некто, полностью владеющий информацией о природе изучаемого явления, указал истинные классы объектов. Практически все алгоритмы обучения ищут такую поверхность или систему поверхностей в пространстве признаков, которая по возможности разделяла бы точки, помеченные «учителем» индексами разных классов, и позволяла бы в дальнейшем, как говорят, на этапе «экзамена», определять класс каждого нового объекта, не участвовавшего в обучении, делая при этом как можно меньше ошибок. Формирование решающих правил распознавания в процессе обучения основано на так называемой гипотезе

компактности, согласно которой объекты, отобразившиеся в близкие точки в пространстве признаков, скорее всего, принадлежат к одному и тому же классу.

Тем не менее, в такой уже ставшей классической области кибернетической науки, как теория обучения распознаванию образов, еще остается множество нерешенных проблем, две из которых являются предметом рассмотрения в данной работе.

Первая проблема, рассматриваемая в данной работе, связана с большой размерностью пространства признаков, что типично для многих прикладных задач. Ярким примером задач такого типа является задача распознавания речевых команд. В большинстве современных систем распознавания речи анализируется последовательность мгновенных спектров сигнала. $x_t(f)$. На практике в каждый момент времени достаточно вычислить конечное число спектральных составляющих для некоторых фиксированных частот $f^{(1)}, \dots, f^{(n)}$. Естественно, что как число спектров в последовательности, так и число частот, образующих каждый из них, должны быть достаточно велики, чтобы отразить особенности каждого звука в составе произнесенной команды. Как результат, произнесенная команда оказывается представленной вектором весьма большой размерности, содержащим, по крайней мере, сотни компонент.

Суть проблемы большой размерности заключается в том, что для обучающей выборки фиксированного размера увеличение количества признаков переменных, т. е. увеличение объема информации о разделяемых классах до некоторого значения, уже более не приводит к улучшению качества распознавания и даже, наоборот, существенно его ухудшает. В этом случае решающее правило оказывается привязанным к особенностям данной конкретной выборки и плохо отражает свойства генеральной совокупности. Действительно, в таком случае удастся сократить количество ошибок на этапе обучения, но под сомнение ставятся экстраполяционные свойства алгоритма. Как ни парадоксально, основная задача обучения состоит вовсе не в сохранении всей имеющейся информации, а, наоборот, в максимальном сокращении внимания к несущественной информации об объектах.

Первый подход [1], основанный на поиске подмножества наиболее информативных признаков, помимо увеличения качества классификации может быть полезен в случаях, когда нежелательно использование полной совокупности признаков переменных, например, из-за непомерно высокой стоимости сбора и обработки данных.

Второй подход [48] обычно используется в случае коротких выборок или в случае, когда селекция признаков по каким-либо соображениям неприемлема.

Суть подхода заключается в наложении на класс решающих правил некоторых ограничений, зависящих от априорных свойств данных. Задача подобной регуляризации состоит в улучшении экстраполяционных свойств полученного решающего правила, возможно, даже за счет принесения в жертву его качества на этапе обучения.

Во-вторых, следует заметить, что выбор признаков, образующих удобное для распознавания пространство, представляет собой отдельную весьма сложную проблему. Однако, в то же время существует широкий класс прикладных задач распознавания образов, в которых легко удастся непосредственно вычислить степень «непохожести» любых двух объектов, но трудно указать набор осмысленных характеристик объектов, которые могли бы служить координатными осями пространства признаков. Наглядным примером задач является задача распознавания рукописных символов [43], например, букв, при их вводе в ЭВМ непосредственно в процессе написания с помощью специального пера, и задача классификации пространственной структуры белков опираясь на знание лишь первичной структуры (последовательности аминокислот) [44].

Эти две на первый взгляд разные проблемы имеют как не странно одно решение. В данной работе для улучшения качества распознавания в обоих случаях предлагается при построении решающего правила учесть априорную информацию об обучающей выборке. В случае проблемы большой размерности пространства признаков, если селекция признаков по каким-либо причинам невозможна, например, в случае распознавания различного рода сигналов, на решающее правило необходимо наложить ограничение, зависящее от априорных свойств данных. В том случае, когда невозможно подобрать удобное для распознавания линейное пространство признаков учет априорной информации об объектах необходимо провести путем выбора подходящего пространства распознавания.

Цель работы. Разработка нового подхода к повышению качества решающего правила на стадии эксперимента для широкого класса задач, а именно распознавания сигналов, и задач, когда невозможно подобрать удобное линейное признаковое пространство.

Научная новизна. В настоящей работе впервые

- предлагается новый подход к проблеме регуляризации решающего правила, полученного при обучении распознаванию на многомерных данных, упорядоченных вдоль оси некоторого аргумента. Типичным примером объектов распознавания такого вида являются сигналы;

- . для прикладных задач, в которых затруднительно, а порой и невозможно явно указать фиксированный набор легко измеряемых признаков объектов вместо линейного векторного признакового пространства предлагается рассматривать множество всех потенциальных объектов непосредственно как метрическое пространство.

Степень обоснованности результатов. Научные положения, результаты и выводы, сформулированные в диссертационной работе, обоснованы теоретически и обсуждены на семинарах и конференциях. Обоснованность предложенных алгоритмов подтверждается результатами модельных экспериментов и обработки реальных данных.

Прикладные результаты работы. Разработанные алгоритмы были использованы для распознавания, во-первых, модельных данных, специальным образом сгенерированных, во-вторых для распознавания рукописных символов, при их вводе в ЭВМ непосредственно в процессе написания с помощью специального пера.

Публикации. Основные результаты исследований по теме диссертации опубликованы в 5 печатных работах.

Апробация работы. Результаты диссертации докладывались

- на молодежной научной конференции «XXVI Гагаринские чтения» (Москва, 1996);
- 8th International Student Olympiad on Automatic Control (BOAC'2000), St. Petersburg, May 24-26, 2000;
- на IX Всероссийской конференции ММРО-9, Вычислительный центр РАН, Москва, 1999г.

Структура работы. Работа состоит из пяти глав. В первой главе рассматриваются современные проблемы теории распознавания образов, в частности, особенности обучения в условиях малого относительного размера обучающей выборки по сравнению с размерностью признакового пространства. Формулируются основные задачи исследования. Во второй главе излагается метод опорных векторов, предлагается реализация метода опорных векторов для многих классов. В третьей главе будет рассмотрен новый подход к проблеме регуляризации решающего правила, полученного при обучении распознаванию на многомерных данных, упорядоченных вдоль оси некоторого аргумента., а также метод обучения в метрическом пространстве на основе понятия опорных элементов, представляющий собой модификацию метода опорных векторов В.Н. Вапника. В четвертой главе приводится описание программного комплекса обучения распознаванию образов, реализующего алгоритмы, предложенные в

основной части работы. Результаты экспериментального исследования алгоритмов приводятся в пятой главе.

1. Проблема обучения распознаванию образов.

1.1 Прикладные задачи, приводящие к проблемам малонаполненных выборок и необходимости использования метрического пространства признаков.

В теории распознавания образов часто встречаются задачи, оперирующие с большим числом признаковых переменных. Это, во-первых, приложения, в которых объединены данные от большого числа датчиков, во-вторых, задачи, объединяющие множество многомерных моделей, параметры которых моделей могут быть использованы для классификации, и, в-третьих, приложения, основанные на получении (извлечении) скрытых зависимостей между признаками.

Яркими примерами задач первого типа могут служить задачи распознавания речевых команд.

Рассмотрим речевую команду. Подобно тому, как сигнал с некоторым неизменным характером колебаний характеризуется его спектром, представляющим собой функцию частоты $x(f)$, адекватной характеристикой нестационарного сигнала будет последовательность его спектров $x_t(f)$, каждый из которых отражает локальный характер колебаний в некоторой окрестности текущей точки. На практике в каждый момент времени достаточно вычислить конечное число спектральных составляющих для некоторых фиксированных частот $f^{(1)}, \dots, f^{(n)}$. Таким образом, результат спектрально-временного анализа сигнала $X = (x_t, t \in T)$ представляет собой последовательность его мгновенных спектров $x_t = (x_t^{(1)}, \dots, x_t^{(n)})$, принимающих значения из множества X действительных векторов некоторой фиксированной размерности. В большинстве современных систем распознавания речи анализируется именно последовательность мгновенных спектров сигнала. Естественно, что как число спектров в последовательности, так и число частот, образующих каждый из них, должны быть достаточно велики, чтобы отразить особенности каждого звука в составе произнесенной команды. Как результат, произнесенная команда оказывается представленной вектором с весьма большой размерности, содержащим, по крайней мере, сотни компонент.

На рис. 1.1 показан сигнал речи, зарегистрированный при произнесении слова “один”, и его представление в виде последовательности мгновенных интенсивностей спектральных составляющих в семи полосах частот, охватывающих диапазон от 10 до 3000 герц. Спектральные кривые упорядочены в соответствии с увеличением частоты. Интенсивности спектральных составляющих показаны в условных индивидуальных масштабах, что позволяет визуально сравнить динамику их изменения во времени.

К задачам подобного типа относится также задача распознавания рукописных символов при их вводе в ЭВМ непосредственно в процессе написания с помощью специального пера. При таком способе ввода каждый символ первоначально оказывается представленным сигналом, состоящим из двух компонент, а именно, текущих координат пера по вертикали и горизонтали, однако, может оказаться целесообразным использовать и дополнительные локальные характеристики процесса написания, например, угловой азимут мгновенного направления движения пера, его скорость, силу прижатия к бумаге, временные отрывы от нее, наклон и т.п. В качестве аргумента сигнала может выступать либо время, либо длина пути, пройденного пером от точки первого касания бумаги. На рис.1.2 представлен трехкомпонентный сигнал в виде функций координат и азимута движения пера от пройденного пути вдоль его траектории, полученный при написании рукописной буквы “d”.

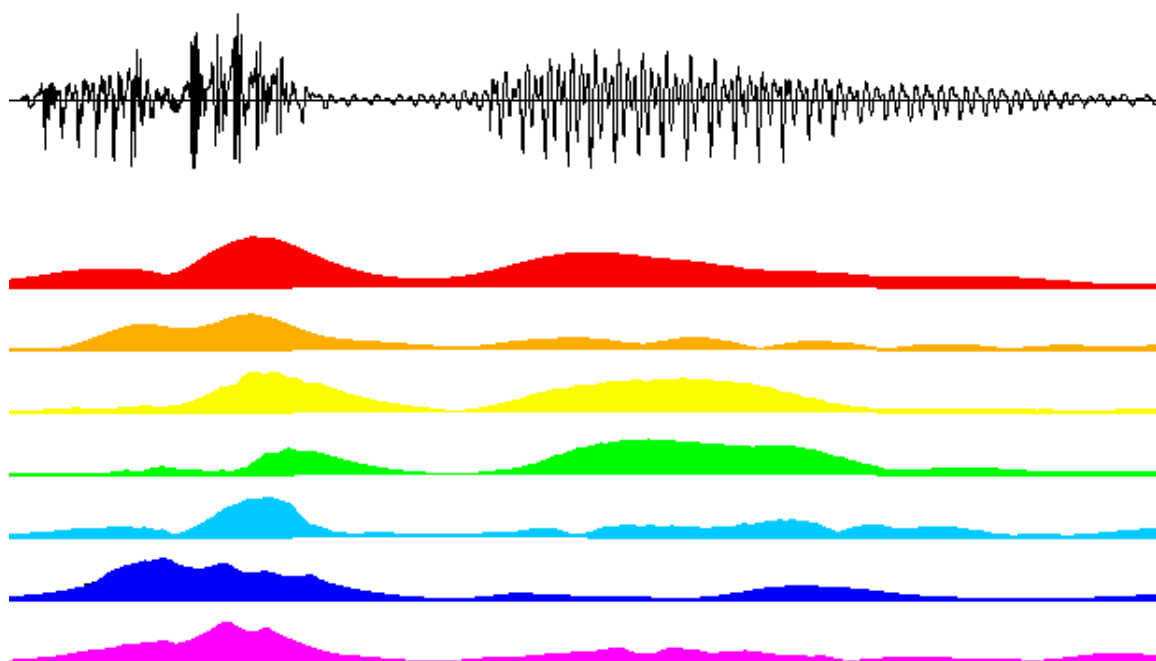


Рисунок. 1.1 Сигнал речи, зарегистрированный при произнесении слова “один”, и его представление в виде последовательности мгновенных интенсивностей спектральных составляющих в семи полосах частот, охватывающих диапазон от 10 до 3000 герц.

Примером задачи второго типа может служить задача классификации использования земельных ресурсов с использованием SAR (синтетического апертурного радара) изображений (рис. 1.2). Например, Солберг и Джэйн [49] использовали текстурные характеристики, рассчитанные на SAR изображениях для классификации каждого пикселя. Все 18 признаков для каждого образа (пикселя) были рассчитаны на основе четырех текстурных моделей: локальные статистики (5 признаков), матрицы уровня серого (6 признаков), фрактальные признаки (2 признака) и логнормальная модель случайного поля (5 признаков).

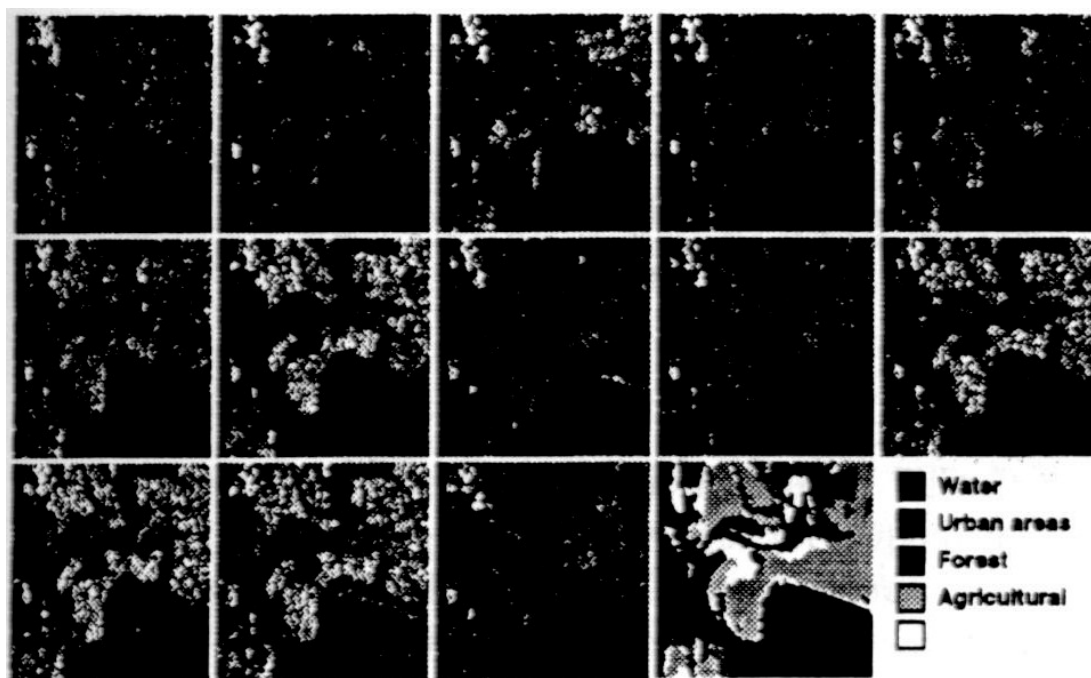


Рисунок. 1.2. Пример SAR изображения.

В качестве примера задач третьего типа [24] может выступить описание физико-химических объектов, которые могут быть представлены в виде совокупности (сочетаний) из n упорядоченных элементов (a, b, c, d, e...) по m [20]. Чтобы отчетливо представлять о каких элементах идет речь, перечислим их.

Во-первых, под (a, b, c, d, e) можно понимать химические элементы H, F, Cl, Br, и J. Тогда, например, метан запишется как CHNNH, а дифторхлорбромметан - CHFFCl. Образуя различные сочетания из пяти элементов по четыре с четырьмя повторениями, получают весь набор галогенметанов (70 молекул).

Во-вторых, молекулярной комбинаторикой могут быть охвачены смеси газов и жидкостей. Принимая за a, b, c, d 25-процентный шаг в концентрациях соответствующего чистого компонента (aaaa), (bbbb), (cccc) или (dddd), путем комбинации элементов получим соответствующую смесь. Например,

| | | | | |
|-----------------------|----------------------|----------------------|----------------------|--------------------|
| 100% H ₂ O | 75% H ₂ O | 50% H ₂ O | 25% H ₂ O | |
| | 25% | 50% | 75% | 100% |
| | CH ₃ OH | CH ₃ OH | CH ₃ OH | CH ₃ OH |

Таким образом, поиск закономерностей формирования класса молекул в структуры приводят к большой размерности вектора признаков.

Следует отметить, что приведенные задачи являются наиболее яркими примерами класса приложений, оперирующих с многомерными данными, а не исключениями из правил. Круг подобных задач чрезвычайно широк. Даже если разработчику и удастся

провести в той или иной мере эффективное снижение размерности или определить наиболее важные факторы, на первых этапах исследования он все же старается запасть как можно большей информацией, чтобы наиболее полно описать изучаемое явление.

Следует заметить, что выбор признаков, образующих удобное для распознавания пространство, представляет собой отдельную весьма сложную проблему. В то же время существует широкий класс прикладных задач распознавания образов, в которых легко удастся непосредственно вычислить степень «непохожести» любых двух объектов, но трудно указать набор осмысленных характеристик объектов, которые могли бы служить координатными осями пространства признаков.

Наглядным примером задачи распознавания образов в метрическом пространстве является задача распознавания рукописных символов, например, букв, при их вводе в ЭВМ непосредственно в процессе написания с помощью специального пера. При таком способе ввода каждый символ первоначально оказывается представленным сигналом, состоящим из двух компонент, а именно, текущих координат пера по вертикали и горизонтали, однако, может оказаться целесообразным использовать и дополнительные локальные характеристики процесса написания, например, угловой азимут мгновенного направления движения пера, его скорость, силу прижатия к бумаге, временные отрывы от нее, наклон и т.п. В качестве аргумента сигнала может выступать либо время, либо длина пути, пройденного пером от точки первого касания бумаги. На рис.1.3 представлен трехкомпонентный сигнал в виде функций координат и азимута движения пера от пройденного пути вдоль его траектории, полученный при написании рукописной буквы “d”.

В этой задаче трудно указать заранее фиксированное число признаков сигнала, которые могли бы сформировать пространство, удовлетворяющее гипотезе компактности. Нельзя использовать в качестве признаков и отсчеты сигнала, взятые с некоторым шагом вдоль оси аргумента, поскольку сигналы, полученные от разных написаний даже одного и того же символа неизбежно будут иметь разную длину, и, следовательно, не существует единого линейного пространства, в котором могли бы быть представлены написания распознаваемых символов [43].

Заметим, что разные варианты написания одного и того же символа естественно представить как результат некоторого нелинейного преобразования оси аргумента, приводящего к ее «короблению». Эти различия между разными написаниями, несущественные с точки зрения распознавания символов, легко компенсировать с помощью процедуры так называемого парного выравнивания (рис. 1.3), тогда остающееся несовпадение сигналов будет нести информацию об «истинной»

непохожести сигналов, которую естественно принять в качестве рабочей метрики при построении процедуры обучения распознаванию символов.

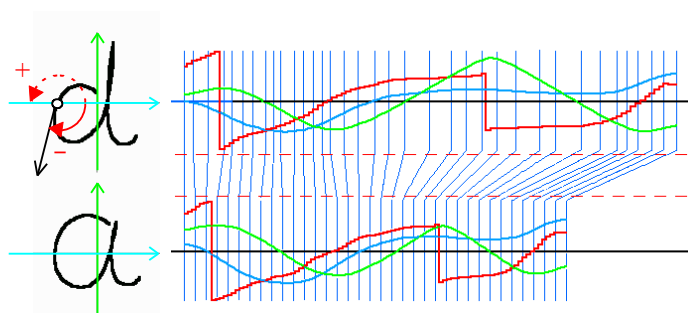


Рисунок. 1.3 Два сигнала в виде функций координат и азимута движения пера от пройденного пути вдоль его траектории, полученные при вводе рукописных символов в компьютер непосредственно в процессе написания. Совмещение произведено по значениям азимута.

Другим примером задач, требующих введения метрического пространства распознавания является задача классификации пространственной структуры белков опираясь на знание лишь первичной структуры (последовательности аминокислот). Информация о пространственной организации белка (третичная структура) является очень важной для понимания механизмов работы макромолекул и их функций. Третичную структуру макромолекул определяют экспериментальными методами (рентгеноструктурный анализ, ядерный магнитный резонанс). Эти методы являются чрезвычайно трудоемкими и требуют больших затрат времени, но позволяют получить достоверные сведения о пространственной организации молекул. Характерно, что для больших групп эволюционно родственных белков, подчас очень значительно отличающихся по первичной структуре и, значит, по распределению всех атомов в пространстве, способ укладки полипептидной цепи остается в главных чертах неизменным. С другой стороны, при всем разнообразии пространственных структур белков удастся выделить относительно небольшое число типов укладки полипептидной цепи. Налицо задача классификации – выделение групп белков, достаточно близких друг к другу по пространственной структуре. Один из фундаментальных принципов молекулярной биологии говорит о том, что последовательность аминокислотных остатков полипептидной цепи белка несет в себе всю информацию, необходимую и достаточную для формирования однозначной пространственной структуры. Учитывая это положение, в настоящее время большие усилия прилагаются для разработки методов предсказания третичной структуры молекул на основе известной первичной структуры. Разумеется, абсолютно точно произвести такой прогноз невозможно, остается надежда на то, чтобы правильно «угадать» группу к которой относится исследуемый белок.

В качестве примера, на рис. 1.4 представлено схематичное трехмерное изображение белка Cytochrome C4 и его первичная структура. На сегодняшний момент биологи выделяют определенные типы (семейства, фолды) известных пространственных структур [44]. Положительным результатом процедуры распознавания считается достоверное отнесение белка к одному из таких классов. Налицо задача распознавания образов.

Пространственная структура



Первичная структура:

```

1   AGDAEAGQGK
11  VAVCGACHGV
21  DGNSPAPNFP
31  KLAGQGGERYL
41  LKQLQDIKAG
51  STPGAPEGVG
61  RKVLEMTGML
71  DPLSDQDLED
81  IAAYFSSQKG
91  SVGYADFPALA
101 KQGEKLFRRGG
111 KLDQGMFACT
121 GCHAPNGVGN
131 DLAGFPKLGG
141 QHAAYTAKQL
151 TDFREGNRTN
161 DGD TMIMRGV
171 AAKLSNKDIE
181 ALSSYIQGLH

```

Рисунок. 1.4. 3D представление и первичная структура белка Cytochrome C4

Имея в наличии информацию только о первичной структуре белков входящих в изучаемые данные, первым шагом представляется попытка получить некоторые количественные характеристики, которые бы отражали существо пространственной классификации. В настоящее время открыто более четырехсот признаков аминокислот (наиболее важные - гидрофобность, степень поляризации, размер и др.), однако, прямое использование этих признаков затруднено тем, что различные протеины имеют разную длину, и, следовательно, непосредственное представление первичных структур как векторных "сигналов" их свойств потребует учета специфики работы с задачами такого типа. Другой простейший подход получения количественных признаков из аминокислотной последовательности заключается в подсчете относительного числа остатков каждой из аминокислот к общей длине последовательности. В таком случае каждый протеин будет представлен точкой в

двадцатимерном пространстве. Однако наиболее разумной представляется схема, использующая знания о взаимной близости аминокислотных последовательностей. Существуют априорные объективные данные о близости в химико-биологическом смысле всех пар аминокислот (210 – пар, включая близость «самой с собой»), которые обычно выражаются в виде матрицы соответствия 20×20 . Для двух аминокислотных последовательностей пытаются найти такое их взаимное соответствие, чтобы величина «невязки» близостей аминокислот была по возможности минимальной. При этом, для белков различной длины более короткий приходится искусственно «вытягивать» за счет введения в определенные позиции делеций, т.е. разрывать исходную структуру. Результат такой процедуры обычно выражается величиной несходства выравниваемых последовательностей. Опираясь на какую либо процедуру выравнивания последовательностей, например Fasta3 (<ftp://ftp.virginia.edu/pub/fasta>), строится матрица всех взаимных расстояний между первичными структурами. Такая матрица и рассматривается как метрическое пространство.

1.2 Современные методы обучения распознаванию образов в пространствах действительных признаков.

1.2.1 Постановка задачи обучения распознаванию образов.

Центральными понятиями формальной постановки задачи обучения распознаванию образов [2, 5, 10, 15, 24, 27, 29, 45] являются:

1. Гипотетическое множество (генеральная совокупность) Ω объектов распознавания. Существенно, что это множество не подлежит восприятию.

2. Индикаторная функция $g(\omega): \Omega \rightarrow M$, $M = \{1, \dots, m\}$, ставящая в соответствие каждому элементу из Ω индекс из конечного множества M , который называется классом. Эта функция разбивает множество Ω на m непересекающихся классов $\Omega^1, \dots, \Omega^m$. Она также не известна наблюдателю.

3. Некоторое пространство наблюдений X , $x \in X$ в пределах которого некоторая функция $x(\omega): \Omega \rightarrow X$, также неизвестная наблюдателю, ставит в соответствие каждому объекту $\omega \in \Omega$ его образ $x(\omega) \in X$, непосредственно воспринимаемый наблюдателем.

Конечной целью распознавания образов является способность наблюдателя угадать класс объекта $\omega \in \Omega$ по его видимому образу $x \in X$, т.е. определить функцию $\hat{g}(x): X \rightarrow M$ - конечное решающее правило, которое позволило бы угадать класс скрытого объекта и при этом и “не слишком часто” ошибаться.

Пункты 1-3 представляют собой первичную модель источника данных. Если бы мы эту модель знали в виде функций $g(\omega)$ и $\mathbf{x}(\omega)$, то задача распознавания, т.е. построение функции $\hat{g}(x)$, свелась бы к инвертированию первичной модели.

Но у наблюдателя нет модели. При этом доступная наблюдателю информация о функциях $g(\omega)$ и $\mathbf{x}(\omega)$, составляющих вместе с множествами Ω , M и X первичную модель источника данных, ограничивается результатами измерений над конечным числом объектов $\omega_j, j=1, \dots, N$, составляющих обучающую совокупность. В зависимости от того, какие измерения могут быть произведены на объектах обучающей совокупности, различают задачи обучения распознаванию образов с учителем, без учителя, а также промежуточный вариант.

Задача обучения с учителем предполагает, что каждый объект ω_j в обучающей совокупности представлен номером своего класса $g_j = g(\omega_j)$ и образом в пространстве наблюдений $\mathbf{x}_j = \mathbf{x}(\omega_j)$, то есть обучающая совокупность в целом есть конечное множество пар $(g_j, \mathbf{x}_j), j=1, \dots, N$. Таковую задачу называют также задачей обучения по классифицированной обучающей совокупности.

В случае задачи обучения без учителя в обучающей совокупности отсутствуют данные о принадлежности объектов к классам, а обучающая совокупность в целом представляет собой просто конечное множество образов объектов в пространстве наблюдений $\mathbf{x}_j, j=1, \dots, N$. При таком понимании задачи обучения говорят об обучении по неклассифицированной обучающей совокупности.

Если в обучающей совокупности принадлежность объектов к классам известна для части объектов и неизвестна для остальных, то обучающую совокупность называют частично классифицированной.

В принципе, пространство наблюдений X может иметь любую природу, но, как правило, под наблюдением $\mathbf{x}(\omega)$ понимают вектор с некоторым фиксированным числом компонент $\mathbf{x} = (x_1, \dots, x_n)^T$, которые называются признаками объекта. Обычно полагают, что признаки принимают действительные $x_i \in \mathbb{R}$ либо дискретные значения, в последнем случае обычно $x_i \in \{0, 1\}$. Мы остановимся на первом варианте, полагая в дальнейшем, что пространство наблюдений X является n -мерным евклидовым пространством \mathbb{R}^n , или пространством признаков, или признаковым пространством.

Качество решающего правила распознавания содержательно интерпретируется в простейшем случае как “частота” правильных решений о классе объекта, но легко придумать ситуацию, когда не все правильные ответы равносильны друг другу и поэтому обычно вводят понятие функции потерь

$$\begin{aligned} \lambda(g, \hat{g}) &\geq 0 \text{ если } g \neq \hat{g} \\ \lambda(g, \hat{g}) &= 0 \text{ если } g = \hat{g} \end{aligned}$$

Понятие частоты ошибки обычно формализуют, рассматривая множество Ω как вероятностное пространство $\langle \Omega, \mathbf{F}, P \rangle$, наделяя его некоторой σ -алгеброй подмножеств \mathbf{F} и вероятностной мерой P . В этом случае для функции $\hat{g}(x)$ также необходимо потребовать измеримость.

Пусть $\hat{g}(x)$ - конкретное решающее правило. Рассмотрим в множестве Ω подмножество Ω^- , для которых наше решение о классе объекта не совпадает с истинным

$$\Omega^- = \{\omega \in \Omega \mid g(\omega) \neq \hat{g}[x(\omega)]\}.$$

Потребуем, чтобы функции $\hat{g}(x)$, $g(x)$ и $x(\omega)$ были таким, чтобы подмножество Ω^- было измеримо в пространстве \mathbf{F} . Тогда для него существует вероятностная мера $P(\Omega^-) \geq 0$, которая характеризует частоту неправильного решения, т.е. оказывается определенным функционал на множестве всех решающих правил

$$J[g(\cdot)] = P\{\Omega^-[g(\cdot)]\}.$$

Естественно выбирать решающее правило таким образом, чтобы $J[\cdot]$ был минимальным.

Тогда решающее правило следует выбирать из условия минимума вероятности ошибки $P(\hat{g}[x(\omega)] \neq g(\omega))$, где ω понимается как случайная переменная, принимающая значения из множества Ω .

Остается дать количественное выражение понятию качества решающего правила распознавания. Пусть зафиксирована функция потерь

$$\begin{aligned} \lambda(g, \hat{g}) &\geq 0 \text{ если } g \neq \hat{g} \\ \lambda(g, \hat{g}) &= 0 \text{ если } g = \hat{g} \end{aligned}$$

зафиксируем также решающее правило, тогда для каждого значения объекта $\omega \in \Omega$ определено значение потерь при распознавании его класса с помощью решающего правила.

$$\omega \in \Omega : [\lambda\{g(\omega), \hat{g}[x(\omega)]\}]$$

Поскольку множество Ω наделено структурой вероятностного пространства то потери от неверного распознавания представляют собой случайную величину. Ее математическое ожидание будем понимать как степень некачественности данного решающего правила. Математическое ожидание потерь при распознавании принято называть средним риском ошибки распознавания:

$$J[\hat{g}(\cdot)] = M[\lambda\{g(\omega), \hat{g}[x(\omega)]\}]$$

1.2.2 Структура оптимального решающего правила

Предположим, что модель источника данных известна. Тогда для каждого класса объектов $\omega \in \Omega^k$, $g(\omega) = k$ в пространстве наблюдений X определено некоторое распределение вероятности. Допустим, что это распределение выразимо в виде

плотности вероятности $\varphi^k(x)$, $x \in X$. Пусть для каждого класса определена вероятность появления объекта этого класса

$$q^k = P[g(\omega) = k], \quad \sum_{k=1}^m q^k = 1, \quad k = 1, \dots, m$$

Т.е., в сущности, наблюдатель имеет дело с двухкомпонентной случайной величиной (g, x) $g \in M = \{1..m\}$, которая принимает значения из декартового произведения $M \times X$, где множество M - дискретно. Вероятностная модель источника данных определяет некоторое совместное распределение вероятности на этом множестве для двухкомпонентной случайной величины (g, x) . Известно, что полное распределение вероятности для двухкомпонентного случайного объекта можно выразить двумя способами:

$$\psi(k, x) = P[g(\omega) = k] \varphi[x(\omega) | g(\omega) = k] = q^k \varphi^k(x) \quad (1.2.1)$$

и

$$\psi(k, x) = \varphi[x(\omega)] P[g(\omega) = k | x(\omega) = x] = f(x) \pi^k(x) \quad (1.2.2)$$

, где $f(x)$ - плотность полного распределения вероятности в пространстве наблюдений X , $\pi^k(x)$ - условная вероятность того, что объект принадлежит к классу k , если известно, что он отобразился в точку x пространства наблюдений X .

Таким образом, с точки зрения наблюдателя полная вероятностная модель источника данных может быть выражена двумя способами.

I. q^k , $k = 1..m$, - априорная вероятность класса,

$$\sum_{k=1}^m q^k = 1;$$

$\varphi^k(x)$ - условные плотности вероятности распределений в пространстве наблюдений X для некоторого класса k .

$$\varphi^k(x) \geq 0, \quad \int_X \varphi^k(x) dx = 1, \quad k = 1..m$$

II. $f(x)$, $x \in X$ - полная плотность распределения в пространстве наблюдений

$$f(x) \geq 0, \quad \int_X f(x) dx = 1,$$

$\pi^k(x)$, $k = 1..m$ - апостериорная вероятность класса объекта.

$$\sum_{k=1}^m \pi^k(x) = 1, \quad 0 \leq \pi^k(x) \leq 1.$$

Функцию $\pi^k(x)$ принято называть функцией степени достоверности. Почему это так будет понятно из дальнейшего изложения.

Предположим, что вероятностная модель источника данных известна полностью. выясним какое решающее правило является оптимальным, т.е. доставляет минимальное значение функционалу среднего риска

$$J[\hat{g}(\cdot)] = M\{\lambda[g, \hat{g}(x)]\} = \int_{M \times X} \lambda[g, \hat{g}(x)] \psi(g, x) \mu(d(g, x)) = \sum_{k=1}^m \int_X \lambda[k, \hat{g}(x)] \psi(k, x) dx \quad (1.2.3)$$

Подставив в выражение для среднего риска определение для полной вероятности (1.2.2), получим

$$J[\hat{g}(\cdot)] = \int_X \left\{ \sum_{k=1}^m \lambda[k, \hat{g}(x)] \pi^k(x) \right\} f(x) dx \quad (1.2.4).$$

Выражение в фигурных скобках естественно называть условным риском при условии, что объект отобразился в точку x пространства наблюдений X . Внутри скобок $\hat{g}(x)$ есть константа при фиксированном x , т.е. это тот класс, в пользу которого решающее правило $\hat{g}(x)$ принимает решение.

Общий средний риск $J[\hat{g}(\cdot)]$ есть ни что иное, интеграл или сумма условных рисков в каждой точке пространства наблюдений.

Кроме того, мы договорились не налагать никаких ограничений на вид решающего правила $\hat{g}(x)$. Это означает, что мы каждой точке пространства наблюдений в праве поставить в соответствие тот индекс класса, который будем приписывать, когда появится объект этого класса. Тогда совершенно очевидно, что для минимизации среднего риска надо выбрать такое решающее правило $\hat{g}(x)$, которое каждой точке пространства наблюдений $x \in X$ ставит в соответствие индекс класса $j = \hat{g}(x)$, доставляющее минимальное значение условному риску ошибки в данной точке, т.е. выражению в фигурных скобках.

Отсюда

$$\hat{g}(x) = \arg \min_{j=1..m} \sum_{k=1}^m \lambda[k, j] \pi^k(x) \quad (1.2.5).$$

Из этой общей структуры решающего правила распознавания видно, что оно опирается не на всю модель источника данных в целом, а лишь на часть этой модели, а именно на векторную функцию $\pi^k(x)$, которая называется функцией степени достоверности.

Полное же распределение вероятности в пространстве наблюдений $f(x)$ оказалось никак не влияющим на структуру оптимального решающего правила.

1.2.3. Восстановление плотностей распределений классов в пространстве признаков.

В выражении (1.2.5) для оптимального решающего правила присутствует функция $\pi^k(x)$, выражающая зависимость вероятности класса от точки пространства наблюдений. Эти вероятности легко получить, если известны априорные вероятности классов и условные плотности распределения для каждого класса [2,10,45]

$$\pi^k(x) = \frac{\varphi^k(x) g^k}{f(x)}$$

Обратим внимание, что после подстановки в (1.2.5) общая плотность распределения играет роль лишь общего коэффициента и не влияет на вид решающего правила. Т.о. надо по обучающей выборке оценить априорные вероятности классов q^k и условные плотности распределений φ^k , $k = 1..m$, $x \in X$.

Сделаем основной акцент на случай того, что $X = R^n$ и $x = \mathbf{x} = (x_1, \dots, x_n)^T$

1. q^k -априорные вероятности классов. Оценкой максимального правдоподобия для них являются частоты

$$\hat{q}_N^k = \frac{N^k}{N},$$

где N – объем выборки, N^k - число объектов k -го класса в выборке.

Возникает вопрос - насколько хороша такая оценка. Предположим, что из генеральной совокупности выбираются объекты независимо, случайно, с возвратом. в результате такого бесконечного эксперимента мы получим последовательность \hat{q}_N^k . Эта последовательность может либо иметь предел, либо не иметь. Факт наличия у последовательности предела будем рассматривать как случайное событие. Можно доказать, что вероятность этого события равна 1. в математической статистике в этом случае говорят, что событие происходит почти наверное.

В свою очередь, если числовая последовательность \hat{q}_N^k сходится, то она может сходиться либо к истинной вероятности класса q^k , либо к какому-то другому числу. Можно доказать, что пределом последовательности \hat{q}_N^k с вероятностью 1 будет q^k . Сходимость почти наверное является наиболее сильным видом сходимости.

2. $\varphi^k(x)$ -условные плотности вероятности распределений в пространстве наблюдений. Ситуация с оцениванием $\varphi^k(x)$ бесконечно сложнее.

Среди методов оценивания плотности распределения различают параметрические [41, 42] и непараметрические.

Обратим внимание, что восстановить плотность распределения – это восстановить функцию. Ограничимся рассмотрением пространства наблюдений в виде $X = R^n$ $x = \mathbf{x} = (x_1, \dots, x_n)^T$, x_1, \dots, x_n - признаки объекта распознавания, R^n - признаковое пространство.

Тогда плотность распределения k -го класса в признаковом пространстве есть плотность распределения некоторой n -мерной случайной величины.

Плотность распределения некоторой есть действительная функция векторного аргумента

$$\varphi^k(\mathbf{x}), \mathbf{x} \in R^n \quad \varphi^k(x) \geq 0, \quad \int_X \varphi^k(x) dx = 1.$$

Непараметрическое восстановление плотности.

Пусть учитель предъявил N объектов k -го класса $\mathbf{x}_1 \dots \mathbf{x}_N$. На рис.1.2..1 показана иллюстрация двумерного случая. Рассмотрим точку $\mathbf{x} \in R^n$. Представляется естественным присвоить этой точке тем большее значение плотности распределения,

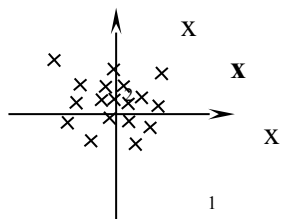


Рисунок 1.2.1

чем больше объектов обучающей выборки попали в достаточно близкую окрестность этой точки. Один из наиболее распространенных способов реализации такой идеи заключается в выборе так называемой потенциальной функции $\eta(\mathbf{x}', \mathbf{x}'')$. Эта функция принципиально двух аргументов $\mathbf{x}', \mathbf{x}'' \in R^n$. Она максимальна, когда $\mathbf{x}' = \mathbf{x}''$, и

равна нулю, когда $\|\mathbf{x}' - \mathbf{x}''\| \rightarrow \infty$. Пример такой функции для одномерного вектора признаков показана на рис 1.2.2.

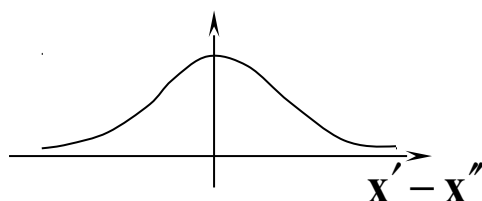


Рисунок 1.2.2. Потенциальная функция $\eta(\mathbf{x}', \mathbf{x}'')$ для одномерного случая

Потребуем от потенциальной функции, чтобы она была неотрицательна

$$\eta(\mathbf{x}', \mathbf{x}'') \geq 0,$$

и, чтобы

$$\int_{R^n} \eta(\mathbf{x}', \mathbf{x}'') d\mathbf{x}' = 1, \mathbf{x}'' \in R^n.$$

Отсюда следует, что потенциальная функция неизбежно стремится к нулю при $\|\mathbf{x}' - \mathbf{x}''\| \rightarrow \infty$. Кроме того потребуем, чтобы потенциальная функция монотонно убывала при увеличении нормы разности $\|\mathbf{x}' - \mathbf{x}''\| \rightarrow \infty$.

В соответствии с вышеизложенным будем оценивать плотность распределения следующим образом:

$$\hat{\phi}^k(\mathbf{x}) = \frac{1}{N} \sum_{j=1}^N \eta(\mathbf{x}, \mathbf{x}_j).$$

Особенности данной оценки:

1. Чтобы вычислить значение функции в точке \mathbf{x} необходимо хранить в памяти всю обучающую выборку
2. Оценка плотности существенным образом зависит от выбора потенциальной функции. Если потенциальную функцию выбрать «острой», то на малых выборках оценка плотности будет неровной, причем эта неровность будет существенно зависеть от данной выборки. Т.е. если $\phi^*(\mathbf{x})$ - истинная плотность, то ее оценки

$\hat{\varphi}(\mathbf{x})$ при острой форме функции $\eta(\mathbf{x}', \mathbf{x}'')$ будут отличаться очень большой вариабельностью. В тоже время, если выбрать функцию $\eta(\mathbf{x}', \mathbf{x}'')$ очень «размытой», то оценка $\hat{\varphi}(\mathbf{x})$ также окажется очень отличной от истинной. Ясно, что здесь есть золотая середина, и она будет зависеть от размера выборки. Чем меньше выборка, тем «размытее» должна быть потенциальная функция. Чем больше выборка, тем лучше она отражает детали истинной плотности, тем более островершинной должна быть потенциальная функция для отражения этих деталей.

Вопрос о том, как выбирать вид потенциальной функций и степень ее островершинности в зависимости от размера выборки и расположения точек в ней, называется теорией непараметрического оценивания [5,11,45].

Параметрические оценки.

Идея параметрического оценивания заключается в том, что оценка $\hat{\varphi}(\mathbf{x})$ некоторой плотности $\varphi^*(\mathbf{x})$, представленной в виде выборки, ищется в пределах некоторого семейства плотностей распределения, задаваемых некоторой общей формулой $\varphi(\mathbf{x}; \mathbf{a})$, содержащей свободно изменяемый в некоторой области параметр \mathbf{a} . Естественно, что такое параметрическое семейство должно удовлетворять условиям:

$$\varphi(\mathbf{x}; \mathbf{a}) \geq 0, \mathbf{x} \in \mathbf{R}^n, \mathbf{a} \in \mathbf{A},$$

$$\int_{\mathbf{x}} \varphi(\mathbf{x}; \mathbf{a}) d\mathbf{x} = 1, \mathbf{a} \in \mathbf{A}$$

Тогда указать конкретную плотность распределения значит указать конкретное значение параметра \mathbf{a} . Наиболее распространенным параметрическим семейством является семейство нормальных распределений

$$\varphi(\mathbf{x}; \mathbf{x}_0, \mathbf{K}) = \frac{1}{|\mathbf{K}|^{\frac{1}{2}} (2\pi)^{\frac{n}{2}}} \exp\left[-\frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^T \mathbf{K}^{-1} (\mathbf{x} - \mathbf{x}_0)\right],$$

где $\mathbf{a}(\mathbf{x}_0, \mathbf{K})$ - вектор параметров, \mathbf{x}_0 - вектор математического ожидания, \mathbf{K} – ковариационная матрица. Но это семейство принципиально унимодально. Существует способ формирования более сложных параметрических семейств. Этот способ заключается в формировании так называемых смесей распределений. Особенно часто используют смеси нормальных распределений.

Договоримся обозначать плотность нормального распределения как $N(\mathbf{x}; \mathbf{x}_0, \mathbf{K})$. Пусть в построении смеси участвуют k нормальных распределений

$$N^i(\mathbf{x}; \mathbf{x}_0^i, \mathbf{K}^i), i = 1..k.$$

Поставим в соответствие каждому нормальному распределению его вес $p_i, i = 1..k, \sum p_i = 1, p_i \geq 0$. Общую плотность распределения смеси выберем в виде линейной комбинации

$$\varphi(\mathbf{x}; \mathbf{a}) = \sum_{i=1}^k p_i N^i(\mathbf{x}; \mathbf{x}_0^i, \mathbf{K}^i)$$

Вектор параметров такого распределения

$$\mathbf{a} = (\mathbf{x}_0^i, \mathbf{K}^i, p^i, i = 1..k)$$

Смесь распределений имеет очень простую вероятностную интерпретацию. При выборе значения случайной величины \mathbf{x} сначала разыгрывают номер распределения с вероятностями p_i , а уже потом разыгрывается то распределение, на которое пал случайный выбор.

Проинтерпретируем некоторые особенности оценивания смесей на примере смесей нормальных распределений.

Чем более сложную форму имеет истинная плотность распределения, подлежащая восстановлению, тем большее число элементов смеси понадобится для ее аппроксимации. Однако нетрудно понять, что число элементов в смеси должно быть меньше размера обучающей выборки, чтобы на каждое распределение приходилось по несколько элементов выборки. При увеличении выборки число элементов можно увеличивать.

Рассмотрим один из методов оценивания параметров распределения, называемый методом максимального правдоподобия.

Пусть $\varphi(\mathbf{x}; \mathbf{a})$ параметрическое семейство распределений. Предположим, что выборка $\mathbf{x}_j, j = 1..N$ получена в результате n независимых испытаний с одним распределением $\varphi(\mathbf{x}; \mathbf{a}^*)$ из этого семейства.

Вся выборка – это вектор из векторов, полученных в каждом эксперименте

$$\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T.$$

Поскольку компоненты этого вектора независимы по условиям эксперимента, то плотность распределения этого вектора есть произведений плотностей

$$f(\mathbf{X}; \mathbf{a}) = \prod_{j=1}^N \varphi(\mathbf{x}_j; \mathbf{a}).$$

При каждом значении параметра \mathbf{a} в комбинированном пространстве $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^T$ общая плотность распределения $f(\mathbf{X}; \mathbf{a})$ имеет, вообще говоря, свой вид. Выборка же в целом образует одну точку в комбинированном пространстве. Идея оценивания по методу максимального правдоподобия заключается в том, чтобы выбрать параметр \mathbf{a} т.о., чтобы эта точка приходилась на максимум плотности распределения.

$$\hat{\mathbf{a}} = \operatorname{argmax}_{\mathbf{a} \in A} f(\mathbf{X}; \mathbf{a}) .$$

Очевидно что ничего не изменится, если от $f(\mathbf{X}; \mathbf{a})$ взять любую монотонную функцию, например логарифм, тогда получим следующее выражения для оценки максимального правдоподобия

$$\hat{\mathbf{a}} = \operatorname{argmax}_{\mathbf{a} \in A} \sum_{i=1}^N \log \varphi(\mathbf{x}_i; \mathbf{a})$$

Для оценивания параметра распределения рассмотрим еще один способ, основанный на оценке корня уравнения регрессии.

Пусть в пространстве R^n задано параметрическое семейство $\varphi(\mathbf{x}; \mathbf{a})$, $\mathbf{a} \in A$, в рамках которого оценивается истинная плотность распределения $\varphi(\mathbf{x}; \mathbf{a}^*)$. Пусть максимальное значение плотности определяется неизвестным наблюдателю значением \mathbf{x} . Рассмотрим плотность распределения $\varphi(\mathbf{x}; \mathbf{a})$ как функцию двух переменных \mathbf{x} и \mathbf{a} . Нам будет удобнее рассматривать не саму эту функцию, а ее логарифм. Здесь \mathbf{x} – случайная величина, имеющая плотность распределения $\varphi(\mathbf{x}; \mathbf{a}^*)$, тогда $\log[\varphi(\mathbf{x}; \mathbf{a})]$ - случайная функция параметра \mathbf{a} .

Рассмотрим математическое ожидание этой случайной функции, которая также будет функцией этого варьируемого параметра \mathbf{a}

$$L(\mathbf{a}) = M_{\mathbf{x}}[\log \varphi(\mathbf{x}; \mathbf{a})] = \int_{R^n} \varphi(\mathbf{x}; \mathbf{a}^*) \log \varphi(\mathbf{x}; \mathbf{a}) d\mathbf{x} .$$

Можно показать, что этот интеграл максимален при $\mathbf{a} = \mathbf{a}^*$.

Обратим внимание, что $\log[\varphi(\mathbf{x}; \mathbf{a})]$ есть случайная функция варьируемого параметра \mathbf{a} . Т.о. мы имеем случайную величину, зависящую от параметра. Допустим, что при любом \mathbf{a} эта случайная величина имеет математическое ожидание, в нашем случае это $L(\mathbf{a})$. Это условное математическое ожидание принято называть уравнением регрессии. Т.о. мы свели задачу оценивания параметра \mathbf{a} к задаче оценивания максимума функции регрессии

$$\mathbf{a}^* = \operatorname{argmax}_{\mathbf{a} \in A} M_{\mathbf{x}}[\log \varphi(\mathbf{x}; \mathbf{a})]$$

Если функция регрессии гладкая, то в точке \mathbf{a} должно выполняться условие:

$$\nabla_{\mathbf{a}} M_{\mathbf{x}}[\log \varphi(\mathbf{x}; \mathbf{a})] = 0 \quad (1.2.6)$$

Если параметрическое семейство $\varphi(\mathbf{x}; \mathbf{a})$ удовлетворяет условию регулярности, то операции дифференцирования и взятия математического ожидания можно поменять местами, т.е. справедливо следующее

$$M_{\mathbf{x}}[\nabla_{\mathbf{a}} \log \varphi(\mathbf{x}; \mathbf{a})] = 0 \text{ при } \mathbf{a} = \mathbf{a}^* . \quad (1.2.7)$$

Уравнение такого вида принято называть уравнением регрессии.

Итак для оценивания параметра неизвестного распределения можно воспользоваться либо утверждением (1.2.6), либо утверждением (1.2.7).

Однако в реальной ситуации мы не знаем функцию регрессии, т.е. не знаем условного математического ожидания. Единственно, чем мы обладаем это выборкой $\mathbf{x}_j, j = 1..N$. Идея оценивания случайной величины заключается в использовании вместо математического ожидания случайной величины его оценки в виде среднего арифметического выборочных значений. Если пользоваться выражением (1.2.5) то получим оценку следующего вида

$$\hat{\mathbf{a}} = \arg \max_{\mathbf{a} \in A} \frac{1}{N} \sum_{j=1}^N \log \varphi(\mathbf{x}_j; \mathbf{a})$$

Если не обращать внимание на N , то это ни что иное как оценка максимального правдоподобия. Если же опираться на утверждение (1.2.5), то получим

$$\frac{1}{N} \sum_{j=1}^N \nabla_{\mathbf{a}} \log \varphi(\mathbf{x}_j; \mathbf{a}) = 0 \Rightarrow \hat{\mathbf{a}}_N$$

Уравнение этого типа принято называть уравнениями правдоподобия.

Если выборка бесконечна и у наблюдателя нет запоминающего устройства, способного поместить даже часть этой выборки, используют рекуррентные процедуры оценивания.

Пусть на шаге j получена оценка $\hat{\mathbf{a}}_j$, пусть пришло очередное наблюдение \mathbf{x}_{j+1} . Новая оценка ищется как некоторая функция

$$\hat{\mathbf{a}}_{j+1} = \eta(\hat{\mathbf{a}}_j, \mathbf{x}_{j+1}).$$

Очень простые оценки для поиска точки максимума функции регрессии дает процедура Киффера-Вольфовица [39].

Аналогичная процедура для оценки корня уравнения регрессии носит название процедуры Робинса-Монро [16,39,51].

Эти процедуры практически эквивалентны друг другу и обеспечивают сходимость почти наверное при очень необременительных предположениях о семействе $\varphi(\mathbf{x}; \mathbf{a})$.

1.2.3 Непосредственное восстановление функции степени достоверности.

Напомним, что для поиска оптимального решающего правила нужны были не плотности распределения, а апостериорные вероятности классов в точке наблюдения

$$\pi^k(\mathbf{x}) = P(g(\omega) = k | \mathbf{x}(\omega) = \mathbf{x}), \mathbf{x} \in X$$

Идея восстанавливать плотности появилась из формулы Байеса в которой наличие $f(\mathbf{x})$ мы сочли излишним

$$\pi^k(\mathbf{x}) = \frac{\varphi^k(\mathbf{x}) g^k}{f(\mathbf{x})}, k = 1..m$$

Именно этот шаг - удаления из знаменателя общей функции распределения $f(\mathbf{x})$ - является очень неблагоприятным. Нетрудно убедиться, что в большинстве случаев функции апостериорных вероятностей $\pi^k(\mathbf{x})$ много проще, чем исходные функции распределения $\varphi^k(\mathbf{x})$. Даже если функции $\varphi^k(\mathbf{x})$ достаточно вычурны, в формуле Байеса они делятся практически на свою копию $f(\mathbf{x}) = g^1\varphi^1(\mathbf{x}) + g^2\varphi^2(\mathbf{x})$ (случай двух классов $m=2$) и очень интенсивно «выглаживаются» (рис. 1.2.3).

Поэтому всегда целесообразно непосредственно восстанавливать апостериорные вероятности классов, а не исходные плотности распределения.

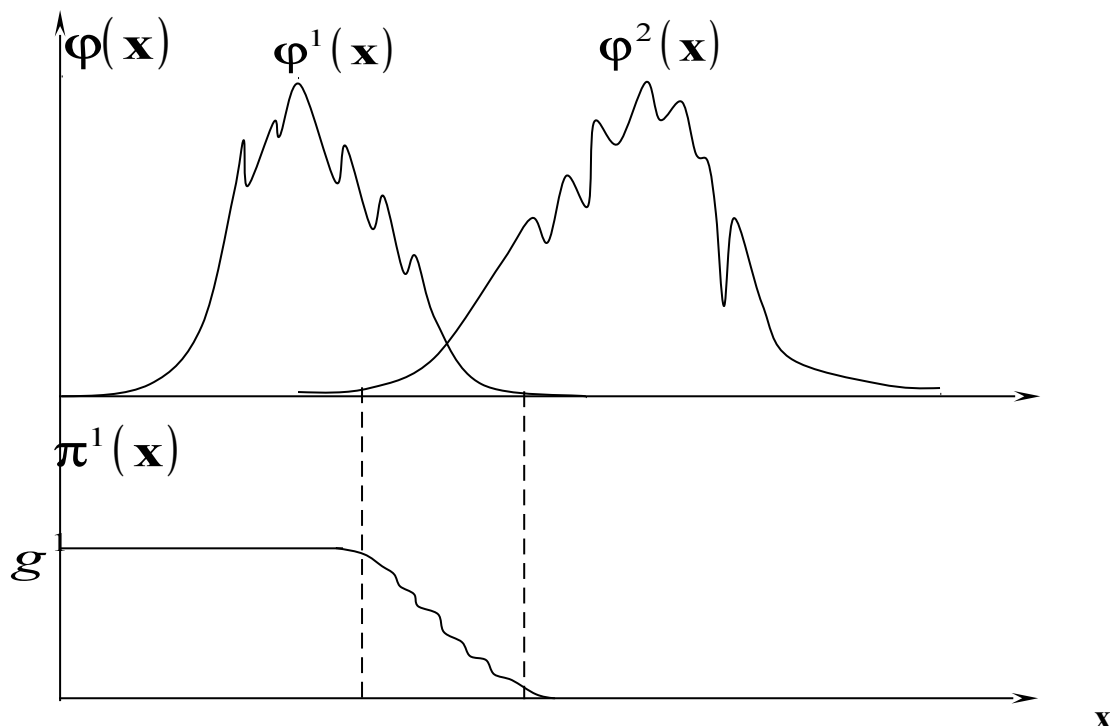


Рисунок 1.2.3 Плотности вероятности классов и апостериорная вероятность одного из классов.

Восстановление функций апостериорных вероятностей классов [17] удобно осуществлять в рамках некоторого параметрического семейства $\pi^k(\mathbf{x}; \mathbf{a})$. Такое семейство должно удовлетворять следующим условиям

$$\mathbf{x} \in R^n, \mathbf{a} \in A$$

$$\pi^k(\mathbf{x}; \mathbf{a}) \geq 0 \sum_{k=1}^m \pi^k(\mathbf{x}; \mathbf{a}) = 1.$$

В простейшем случае, когда число классов два, достаточно определить параметрическое семейство для вероятности лишь одного класса $0 \leq p(\mathbf{x}; \mathbf{a}) \leq 1$ $\mathbf{x} \in R^n, \mathbf{a} \in A$

$$\pi^1 = p(\mathbf{x}; \mathbf{a})$$

$$\pi^2 = 1 - p(\mathbf{x}; \mathbf{a})$$

Пусть $p(\mathbf{x}; \mathbf{a})$ как раз такое параметрическое семейство. Это семейство необходимо выбирать таким образом, чтобы степень его изменчивости как функции \mathbf{x}

не слишком подчинялась параметру \mathbf{a} . График апостериорной вероятности первого класса удобно представить в пространстве (на плоскости) как совокупность поверхностей равных значений. Надо так ограничить параметрическое семейство функций, чтобы эти поверхности не были слишком сложными. Иногда, после того, как представлена обучающая выборка (g_j, \mathbf{x}_j) , $j=1, \dots, N$, и мы потребуем от алгоритма подобрать такое значение параметра \mathbf{a} , чтобы $p(\mathbf{x}; \mathbf{a}) \rightarrow 1$ в тех точках, которые помечены индексом первого класса, и, чтобы $p(\mathbf{x}; \mathbf{a}) \rightarrow 0$ в точках, помеченных индексом второго класса, то у алгоритма появляется соблазнительная возможность сделать это буквально.

Ниже мы это богатство назовем емкостью класса решающих правил.

Если эта емкость слишком велика, то функция $p(\mathbf{x}; \mathbf{a})$, идеально аппроксимировав обучающую выборку, будет плохо согласовываться с другими контрольными выборками. При слишком большой емкости класса решающих правил $p(\mathbf{x}; \mathbf{a})$ будет аппроксимировать индивидуальные особенности обучающей выборки, ошибочно принимая их за истинные различия между классами.

Именно этот аспект есть центральный момент обучения распознаванию образов.

Итак, пусть параметрическое семейство $p(\mathbf{x}; \mathbf{a})$ выбрано. Пусть (g_j, \mathbf{x}_j) , $j=1, \dots, N$ обучающая выборка $g_j = 1..m$. Рассмотрим один из путей поиска параметра \mathbf{a} , при котором $p(\mathbf{x}; \mathbf{a})$ наилучшим образом может быть согласована с выборкой.

Рассмотрим случайную величину
$$z(\omega) = \begin{cases} 1 & g(\omega) = 1 \\ 0 & g(\omega) = 2 \end{cases}$$

Если выбранное параметрическое семейство таково что существует значение параметра \mathbf{a} , при котором в точности воспроизводится апостериорная вероятность первого класса в точке \mathbf{x} , то в каждой точке \mathbf{x} условное математическое ожидание случайной величины z совпадает с апостериорной вероятностью класса в этой точке $P(g(\omega) = 1 | \mathbf{x}(\omega) = \mathbf{x})$, $\mathbf{x} \in X$, т.е. $M[z(\omega) - p(\mathbf{x}; \mathbf{a}) | \mathbf{x}(\omega) = \mathbf{x}] = 0$ для всех $\mathbf{x} \in X$.

В свою очередь, рассматривая $\mathbf{x} = \mathbf{x}(\omega)$ как случайный вектор, получим

$$M\{z(\omega) - p[\mathbf{x}(\omega); \mathbf{a}]\} = 0 \quad (1.2.8).$$

Именно это условие и будем использовать для поиска параметра \mathbf{a} .

Рассмотрим неотрицательную случайную величину $(z(\omega) - p[\mathbf{x}; \mathbf{a}])^2$. Известно, что математическое ожидание случайной величины $z(\omega)$ минимизирует квадрат разности этой случайной величины и наперед заданной неслучайной величины, поэтому поиск корня будем проводить из условия

$$M\{(z(\omega) - p[\mathbf{x}; \mathbf{a}])^2\} \rightarrow \min_{\mathbf{a} \in A} \quad (1.2.9)$$

Если параметрическое семейство $p(\mathbf{x}; \mathbf{a})$ дифференцируемо по \mathbf{a} , то условием минимума будет

$$\nabla_{\mathbf{a}} M\{(z(\omega) - p[\mathbf{x}; \mathbf{a}])^2\} = 0$$

Если семейство $p(\mathbf{x}; \mathbf{a})$ удовлетворяет условию регулярности, то знаки взятия производной и математического ожидания можно поменять местами

$$M\{\nabla_{\mathbf{a}}(z(\omega) - p[\mathbf{x}; \mathbf{a}])^2\} = 0$$

$$\nabla_{\mathbf{a}}(z(\omega) - p[\mathbf{x}; \mathbf{a}])^2 = -2(z(\omega) - p[\mathbf{x}; \mathbf{a}]) \cdot \nabla_{\mathbf{a}} p[\mathbf{x}; \mathbf{a}].$$

Таким образом, приходим к следующему условию оценивания

$$M\{(z(\omega) - p[\mathbf{x}; \mathbf{a}]) \cdot \nabla_{\mathbf{a}} p[\mathbf{x}; \mathbf{a}]\} = 0 \quad (1.2.10)$$

До сих пор мы исходили из того, что условие (3.3) при некотором \mathbf{a} выполняется точно. Но в действительности $p(\mathbf{x}; \mathbf{a})$ не воспроизводит в точности функцию апостериорной вероятности классов. Однако ясно, что условия (3.4) и (3.5) остаются вполне разумными критериями для выбора подходящего параметра \mathbf{a} в данном случае.

Однако использовать эти условия напрямую нельзя, так как нам не известны математические ожидания. Но вместо них у нас есть обучающая выборка (g_j, \mathbf{x}_j) , $j = 1, \dots, N$. Тогда параметр \mathbf{a} следует выбирать исходя из следующих условий

$$\begin{aligned} \frac{1}{N} \sum_{j=1}^N \{I[g_j = 1] - p(\mathbf{x}_j; \mathbf{a})\}^2 &\rightarrow \min \\ \frac{1}{N} \sum_{j=1}^N \{I[g_j = 1] - p(\mathbf{x}_j; \mathbf{a})\} \nabla_{\mathbf{a}} p(\mathbf{x}_j; \mathbf{a}) &= 0. \end{aligned}$$

Эти условия и являются рабочими для построения алгоритма обучения.

Здесь

$$I[g_j = 1] = \begin{cases} 1, & g_j = 1 \\ 0, & g_j = 0 \end{cases} -$$

индикаторная функция.

1.2.4. Прямое восстановление решающего правила распознавания.

В тех случаях, когда невозможно принять какие-либо простые предположения о вероятностной модели данных, а следовательно и о априорной вероятности классов $\pi^k(x)$, $k = 1..m$, то неуместно ставить задачу обучения как задачу восстановления этой модели. Но ведь модель нужна только для того, чтобы затем, опираясь на нее, построить решающее правило распознавания $\hat{g}(x)$, поэтому наиболее

распространенный подход к распознаванию образов заключается в прямом восстановлении решающего правила распознавания.

Однако, говоря о восстановлении модели источника данных, мы всегда ограничивали сложность этой модели. Естественно, из простой модели получались простые решающие правила. Теперь между обучающей выборкой и решающим правилом промежуточного звена нет, и необходимо ограничивать непосредственно сложность решающего правила. Обычно это означает необходимость использования некоторого достаточно простого параметрического семейства решающих правил.

В случае двух классов $m=2$ такие параметрические семейства строятся на основании некоторой дискриминантной функции $d(\mathbf{x}; \mathbf{c})$, принимающей в каждой точке признакового пространства X действительные значения, \mathbf{c} – некоторый векторный параметр $\mathbf{c} \in C$.

$$\hat{g}(\mathbf{x}) = \begin{cases} 1, & d(\mathbf{x}; \mathbf{c}) > 0 \\ 2, & d(\mathbf{x}; \mathbf{c}) \leq 0 \end{cases}$$

В простейшем случае дискриминантная функция может быть взята линейной

$$d(\mathbf{x}; \mathbf{a}, b) = \mathbf{a}^T \mathbf{x} + b, \mathbf{a} \in R^n, b \in R.$$

Заметим, что уравнение $\mathbf{a}^T \mathbf{x} + b = 0$ при $b = 0$ задает в пространстве R^n множество точек $\mathbf{x} \in R^n$, удовлетворяющих этому уравнению и это уравнение принято называть гиперплоскостью.

Вектор \mathbf{a} называют в этом случае направляющим вектором гиперплоскости. Очевидно, что гиперплоскость не изменится если направляющий вектор умножить на любой коэффициент отличный от нуля. Обратим внимание, что гиперплоскость является подпространством пространства R^n и его размерность на единицу меньше размерности исходного пространства.

Существенным свойством подпространства вообще и гиперплоскости в частности является то, что ему всегда принадлежит нулевая точка.

Рассмотрим ситуацию, когда $b \neq 0$. В этом случае точка $\mathbf{x} = \mathbf{0}$ уже не будет принадлежать множеству точек пространства, удовлетворяющего уравнению

$\mathbf{a}^T \mathbf{x} + b = 0$. Следовательно, множество таких точек не является подпространством и не является, таким образом, гиперплоскостью. Множества такого типа принято называть аффинными многообразиями (рис. 3.4). Нетрудно убедиться, что аффинное многообразие, а следовательно и

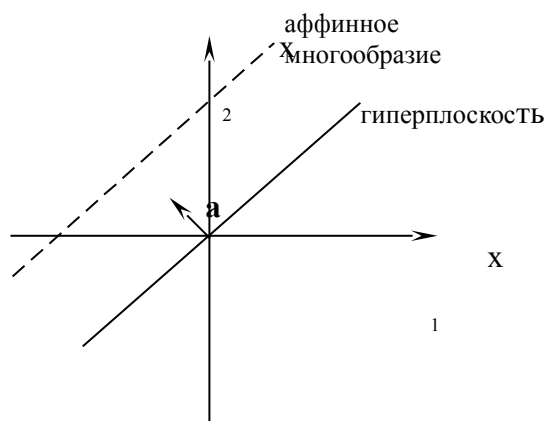


Рисунок 1.2.4 Гиперплоскость и аффинное многообразие

гиперплоскость разбивают все пространство на два непересекающихся подмножества $\mathbf{a}^T \mathbf{x} + b > 0$ и $\mathbf{a}^T \mathbf{x} + b \leq 0$. В дальнейшем мы будем называть множество $\mathbf{a}^T \mathbf{x} + b = 0$ гиперплоскостью и при $b \neq 0$.

В данных терминах линейная дискриминантная функция определяет некоторую гиперплоскость, разбивающую пространство на области принятия решений первого и второго класса соответственно

$$d(\mathbf{x}; \mathbf{a}, b) = \mathbf{a}^T \mathbf{x} + b \begin{cases} > 0 & k = 1 \\ \leq 0 & k = 2 \end{cases}.$$

Однако, во многих случаях возникает надобность использования в качестве границ между классами более сложные поверхности, чем гиперплоскости. Этого добиваются, используя в качестве дискриминантной функции полиномиальную функцию вида

$$d(\mathbf{x}; \mathbf{a}, n) = \sum_{i=1}^n a_i y_i(\mathbf{x}),$$

где $y_0(\mathbf{x}), y_1(\mathbf{x}) \dots y_n(\mathbf{x})$ - подходящее семейство базисных функций.

Нетрудно убедиться, что линейная дискриминантная функция есть частный случай полиномиальной

$$\begin{aligned} y_0(\mathbf{x}) &\equiv 1, i = 0 \\ y_i(\mathbf{x}) &= x_i, i = 1..n \end{aligned}$$

В теории обучения распознаванию образов достаточно ограничиться рассмотрением линейных дискриминантных функций, поскольку значения базисных функций всегда можно рассматривать как новые признаки объекта, обеспечивающие переход в новое признаковое пространство, в котором дискриминантные функции уже являются линейными.

В силу этого обстоятельства пространство, образованное значениями базисных функций называется спрямляющим базисным пространством.

Поскольку мы задали класс дискриминантных функций параметрически $d(\mathbf{x}; \mathbf{a})$, то и класс решающих правил распознавания также оказывается параметрически задан. Ранее мы определяли функционал, численно равный среднему риску ошибки распознавания $J[\hat{g}(\cdot)]$. Поскольку решающее правило зависит от параметра, то и средний риск зависит от параметра $J[\mathbf{a}]$.

Задачу обучения распознаванию образов естественно поставить как задачу определения такого параметра дискриминантной функции, при котором средний риск минимален

$$J[\hat{g}(\cdot)] = M[\lambda\{g(\omega), \hat{g}[\mathbf{x}(\omega); \mathbf{a}]\}] \rightarrow \min_{\mathbf{a} \in A} \quad (1.2.11).$$

Как и всякую функцию средний риск можно минимизировать, анализируя на каждом шаге величину градиента $\nabla_{\mathbf{a}} J[\mathbf{a}]$. необходимым условием минимума среднего риска является равенство нулю градиента

$$\nabla_{\mathbf{a}} M[\lambda\{g(\omega), \hat{g}[\mathbf{x}(\omega); \mathbf{a}]\}] = 0 \quad (1.2.12).$$

Однако для функции потерь $\lambda[g, \hat{g}]$, зависящей только от истинного и экспериментального значения класса объекта перестановка местами операций взятия градиента и математического ожидания

$$M[\nabla_{\mathbf{a}} \lambda\{g(\omega), \hat{g}[\mathbf{x}(\omega); \mathbf{a}]\}] = 0 \quad (1.2.13)$$

неправомерна. Это происходит в силу того скачкообразной зависимости решающего правила $\hat{g}[\mathbf{x}(\omega); \mathbf{a}]$ от параметра \mathbf{a} при любом $\mathbf{x} \in X$.

для преодоления этой трудности переходят к функциям потерь вида $\lambda[\mathbf{x}, \mathbf{a}, g]$, штрафую не просто факт принадлежности объекта к неверному классу, но и слишком близкой приближение вектора \mathbf{x} к разделяющей гиперплоскости в области своего класса и тем более расстояния, на которое вектор вторгается в чужую область.

В реальных условиях, так как вероятностные характеристики источника данных нам неизвестны, математические ожидания в формулах (3.6), (3.7), (3.8) не могут быть вычислены, и не может быть вычислено значение среднего риска. Тем не менее, нашей целью является приближение к оптимальному значению

$$\mathbf{a}^* = \arg \min_{\mathbf{a} \in A} J[\mathbf{a}].$$

Наблюдателю доступна лишь обучающая выборка (g_j, \mathbf{x}_j) , $j = 1, \dots, N$. Компромисс заключается в замене математического ожидания средним арифметическим

$$\hat{J}_N(\mathbf{a}) = \frac{1}{N} \sum_{j=1}^N \lambda(\hat{g}[\mathbf{x}_j; \mathbf{a}], g_j).$$

Такую оценку называют эмпирическим риском, т.е. риском, измеренным на выборке.

Минимизируя эмпирический риск, мы фактически заменим оптимальное значение

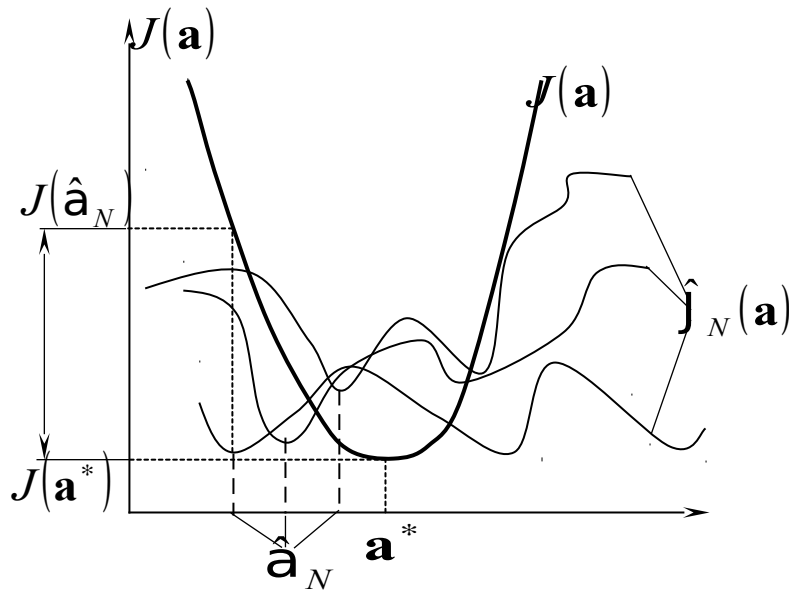


Рисунок 1.2.5 Средний и эмпирический риски ошибки распознавания

параметра \mathbf{a}^* его оценкой $\hat{\mathbf{a}}_N = \arg \min_{\mathbf{a} \in A} \hat{J}_N[\mathbf{a}]$. Очевидно, что эмпирический риск является случайной функцией $\hat{J}_N(\mathbf{a})$ параметра \mathbf{a} , определяемой размером N и конкретными значениями ее элементов (g_j, \mathbf{x}_j) , $j = 1, \dots, N$, где (g_j, \mathbf{x}_j) представляет собой двухкомпонентную случайную величину. Как следствие случайным оказывается и значение параметра $\hat{\mathbf{a}}_N = \arg \min_{\mathbf{a} \in A} \hat{J}_N(\mathbf{a})$, а также и значение среднего риска $J(\hat{\mathbf{a}}_N)$ (рис. 3.5). Для того, чтобы контролировать качество наблюдения необходимо контролировать величину случайного отклонения $|J(\hat{\mathbf{a}}_N) - J(\mathbf{a}^*)|$.

Теоретически осуществить такой контроль методами статистики очень трудно. Этот вопрос относится к так называемой статистике экстремальных значений [10,19]. Мы подменим вопрос об отклонении минимума случайной реализации от минимума ее математического ожидания более общим вопросом об отклонении реализации случайной функции от ее математического ожидания.

Говорят, что случайная функция $\hat{J}_N(\mathbf{a})$, зависящая от выборки размера N , равномерно сходится к некоторой неслучайной функции, если выполняется следующие условие

$$\lim_{N \rightarrow \infty} P\left(\left|\hat{J}_N(\mathbf{a}) - J(\mathbf{a})\right| > \varepsilon \text{ хотя бы для одного } \mathbf{a}\right) = 0 \text{ для всех } \varepsilon > 0.$$

Справедливо утверждение:

если для всех \mathbf{a} $|\hat{J}_N(\mathbf{a}) - J(\mathbf{a})| \leq \varepsilon$, то $|J(\hat{\mathbf{a}}_N) - J(\mathbf{a}^*)| \leq \varepsilon$, где $\hat{\mathbf{a}}_N = \arg \min_{\mathbf{a} \in A} \hat{J}_N(\mathbf{a})$, а

$\mathbf{a}^* = \arg \min_{\mathbf{a} \in A} J(\mathbf{a})$ (рис. 1.2.6).

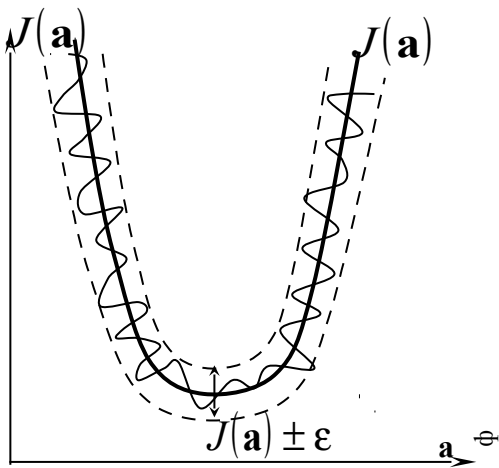


Рисунок 1.2.6 Иллюстрация сходимости случайной функции к ее математическому ожиданию

Т.о. из равномерной сходимости случайной функции $\hat{J}_N(a)$ к ее математическому ожиданию $J(a)$ непосредственно следует, что $\lim_{N \rightarrow \infty} P(|J(\hat{a}_N) - J(a^*)| > \epsilon) = 0$ для всех ϵ .

Иначе говоря, при $N \rightarrow \infty$ средний риск, обеспечиваемый эмпирически оптимальным решающим правилом неограниченно приближается по вероятности к минимально возможному среднему риску в данном семействе решающих правил.

Мы до сих пор вели разговор о среднем риске ошибки, основанном на произвольной

функции потерь $\lambda[g, \hat{g}]$. Однако функция потерь только тогда соответствует своему назначению, когда она удовлетворяет следующим условиям

- 1) $\lambda[g, \hat{g}] = 0$ $g = \hat{g}$,
- 2) $0 \leq \lambda[g, \hat{g}] \leq \lambda_{\max} < \infty$ $g \neq \hat{g}$.

В простейшем случае потери для любых видов ошибки распознавания могут быть назначены одинаковыми

$$\begin{aligned} \lambda[g, \hat{g}] &= 0 \quad g = \hat{g}, \\ \lambda[g, \hat{g}] &= \lambda_{\max} \quad g \neq \hat{g}. \end{aligned}$$

Тогда нет никакой необходимости брать λ_{\max} отличной от единицы. В этом случае матрица потерь будет называться антидиагональной

$$\lambda[g, \hat{g}] = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

Интуитивно понятно, что если равномерная сходимость эмпирического риска к среднему риску имеет место для антидиагональной матрицы потерь, то она имеет место и при более щадящей функции потерь.

Если матрица потерь антидиагональна, то средний риск ошибки распознавания есть ни что иное как вероятность оцененного класса с истинным

$$P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega]) = J(a)$$

с точностью до λ_{\max} .

Однако мы на обучающей выборке можем вычислять лишь частоту ошибки распознавания в пределах обучающей выборки

$$\frac{1}{N} \sum_{j=1}^N I(\hat{g}[\mathbf{x}_j; \mathbf{a}] \neq g_j).$$

Будем говорить, что частота появления события, зависящего от параметра, сходится по вероятности равномерно по всем значениям параметра к вероятности этого события, если выполняется условие

$$\lim_{N \rightarrow \infty} P \left\{ \frac{1}{N} \sum_{j=1}^N I(\hat{g}[\mathbf{x}_j; \mathbf{a}] \neq g_j) - P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega]) > \varepsilon \text{ хотя бы для одного } \mathbf{a} \in \mathbf{A} \right\} = 0, \text{ для } \forall \varepsilon > 0$$

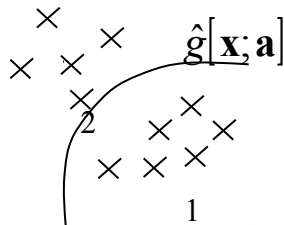
Интуитивно ясно, что сходимость частоты неверного определения класса объекта к вероятности этого события является достаточным условием равномерной сходимости эмпирического риска к среднему риску при любой функции потерь выделенного выше класса.

Этот факт имеет или не имеет места в зависимости, во-первых, от того, какое параметрическое семейство решающих правил выбрано, во-вторых, от того вероятностные характеристики источника данных. Мы сейчас займемся выработкой терминологии, которая позволит нам различать разные классы решающих правил с точки зрения их способности обеспечить равномерную сходимость частоты неверного распознавания к вероятности этого события.

Пусть $\hat{g}[\mathbf{x}; \mathbf{a}]$ - некоторое семейство решающих правил распознавания. Пусть $\mathbf{x}_1, \dots, \mathbf{x}_N$ - некоторое конечное множество точек в пространстве признаков. При каждом значении параметра \mathbf{a} решающее правило распознавания делит эти N точек на два класса (рис.1.2.7). Причем это деление зависит от выбора параметра. Вообще говоря, если не обращать внимание на класс решающих правил, N объектов можно разбить на два класса

$$\sum_{K=1}^N C_N^{N-K} = \sum_{K=1}^N \frac{N!(N-K)!}{K!}$$

способами. Однако в рамках принятого класса решающих правил не все из возможных способов можно реализовать. Кроме того число способов которыми



данное параметрическое семейство решающих правил можно разбить эти точки на два класса зависит еще и от расположения этих точек в признаковом пространстве.

Обозначим через $\Delta(\mathbf{x}_1, \dots, \mathbf{x}_N)$ число способов, которым данное параметрическое семейство разбивает на два класса данную совокупность точек за счет варьирования параметра \mathbf{a} . Будем рассматривать точки $\mathbf{x}_1, \dots, \mathbf{x}_N$ как

Рисунок 1.2.7 Решающее правило распознавания

случайные точки в составе обучающей выборке.

Плотность распределения такого случайного вектора — это полная функция распределения \mathbf{x} в признаковом пространстве по всем классам

$$f(\mathbf{x}) = \sum_{k=1}^m g^k \varphi^k(\mathbf{x}),$$

тогда $\Delta(\mathbf{x}_1, \dots, \mathbf{x}_N)$ -случайная величина.

Математическое ожидание логарифма этой случайной величины

$$H(N) = M\{\log_2 \Delta(\mathbf{x}_1, \dots, \mathbf{x}_N)\}$$

называют энтропией данного семейства решающих правил на выборке $\mathbf{x}_1, \dots, \mathbf{x}_N$.

Теорема. Для равномерной сходимости частоты появления событий $\hat{g}[\mathbf{x}(\omega)] \neq g[\omega]$ к их вероятности необходимо и достаточно, чтобы вероятностные меры на множестве Ω и семейство решающих правил удовлетворяли условию

$$\lim_{N \rightarrow \infty} \frac{H(N)}{N} = 0.$$

Однако на практике удобнее пользоваться значительно более конструктивным, хотя и только достаточным условием равномерной сходимости частот появления ошибок к их вероятностям, опирающимся только на решающее правило $\hat{g}[\mathbf{x}; \mathbf{a}]$ и инвариантное к конкретному виду распределения вероятностей на множестве Ω .

Емкостью семейства решающих правил называется максимальное число V точек $\mathbf{x}_1, \dots, \mathbf{x}_V$ в пространстве R^n , которые можно разбить на два класса всеми возможными способами за счет варьирования параметра \mathbf{a} .

В частности $v = n + 1$ для семейства линейных решающих правил

$$\hat{g}(\mathbf{x}; \mathbf{a}, b) = \begin{cases} 1; \mathbf{a}^T \mathbf{x} + b > 0 \\ 2; \mathbf{a}^T \mathbf{x} + b \leq 0 \end{cases}$$

Теорема. Справедлива оценка сверху для вероятности отклонения частоты ошибочной классификации от ее вероятности

$$P\left\{\sup_{\mathbf{a} \in A} \left| \frac{1}{N} \sum_{j=1}^N I(\hat{g}[\mathbf{x}_j; \mathbf{a}] \neq g_j) - P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega]) \right| > \varepsilon \right\} < 4,5 \frac{(2N)^v}{v!} e^{-\frac{\varepsilon^2(N-1)}{4}}$$

$$P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega])$$

Заметим, что величина $P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega])$ неслучайна и является характеристикой источника данных, а $\frac{1}{N} \sum_{j=1}^N I(\hat{g}[\mathbf{x}_j; \mathbf{a}] \neq g_j)$ - случайная величина, зависящая от выборки. Из этой теоремы непосредственно следует, что для равномерной сходимости частот к вероятности достаточно, чтобы емкость V семейства решающих правил $\hat{g}[\mathbf{x}; \mathbf{a}]$ была конечной.

Допустим, что по обучающей выборке (g_j, \mathbf{x}_j) , $j=1, \dots, N$ в рамках параметрического семейства $\hat{g}[\mathbf{x}; \mathbf{a}]$ проведено обучение распознаванию образов по методу минимизации эмпирического риска и получено значение параметра $\hat{\mathbf{a}}_N = \arg \min_{\mathbf{a} \in A} \hat{J}_N(\mathbf{a})$. фундаментальным вопросом теории распознавания образов является вопрос о величине среднего риска ошибки распознавания $J(\hat{\mathbf{a}}_N)$ для решающего правила $\hat{g}[\mathbf{x}; \hat{\mathbf{a}}_N]$.

Ограничимся рассмотрением случая антидиагональной матрицы потерь. тогда средний риск ошибки распознавания $J(\mathbf{a})$ - это вероятность ошибки распознавания, а эмпирический риск – это доля неверно классифицированных объектов, т.е. в сформулированной выше теореме под знаком модуля сумма $\frac{1}{N} \sum_{j=1}^N I(\hat{g}[\mathbf{x}_j; \mathbf{a}] \neq g_j)$ есть эмпирический риск, а вероятность $P(\hat{g}[\mathbf{x}(\omega)] \neq g[\omega])$ - средний риск. Поэтому теорему можно переписать следующим образом

$$P\left\{\sup_{\mathbf{a} \in A} |\hat{J}_N(\mathbf{a}) - J(\mathbf{a})| < \varepsilon\right\} > 1 - \eta_N,$$

где

$$\eta_N = 4,5 \frac{(2N)^v}{v!} e^{-\frac{\varepsilon^2(N-1)}{4}}.$$

Отсюда

$$P\left\{|\hat{J}_N(\mathbf{a}) - J(\mathbf{a})| < \varepsilon\right\} > 1 - \eta_N,$$

и следовательно

$$P\{J(\hat{\mathbf{a}}_N) < \hat{J}_N(\hat{\mathbf{a}}_N) + \varepsilon\} > 1 - \eta_N \quad (1.2.14)$$

В сущности это то неравенство, которое позволят судить о качестве решающего правила распознавания, полученного по обучающей выборке. Однако это неравенство является чрезвычайно осторожным и очень сильно занижает вероятность выполнения условия $J(\hat{\mathbf{a}}_N) < \hat{J}_N(\hat{\mathbf{a}}_N) + \varepsilon$. Осторожность этой оценки качества решающего правила вытекает по-видимому из того факта, что мы заменили понятие энтропии семейства решающих правил его емкостью. Грубость понятия емкости семейства решающих правил связана с тем, что оно сформулировано без учета вероятностных характеристик источника данных. В результате утверждения на основе этого понятия покрывают «самые плохие» распределения вероятности, даже те, которых никогда не бывает в реальности.

Обратим внимание, что теорема о связи равномерной сходимости частот ошибки к их вероятностям на основе свойств энтропии семейства решающих правил имеет вид

необходимого и достаточного условия, т.е. вовсе не является излишне осторожной гарантией.

В тоже время аналогичное условие в терминах емкости класса решающих правил уже имеет лишь достаточный характер.

1.2.6 Детерминистский вариант задачи обучения распознавания образов.

Оптимальная разделяющая гиперплоскость.

В качестве пространства признаков будем рассматривать n -мерное действительное пространство \mathbf{R}^n , понимая его точки как вектор-столбцы $\mathbf{x} = (x_1, \dots, x_n)^T$. Будем также рассматривать только случай двух классов, причем нам будет удобнее выбрать в качестве индексов классов не их номера не 1 и 2, а индексы $g = 1$ и $g = -1$.

Пусть Ω - гипотетическое множество всех мыслимых объектов распознавания $\omega \in \Omega$, каждый из которых характеризуется фактической принадлежностью к первому либо второму классу, выражаемой неизвестным, вообще говоря, значением индикаторной функции класса $g(\omega)$, соответственно, $g(\omega) = 1$ либо $g(\omega) = -1$, и наблюдаемым значением вектора признаков $\mathbf{x}(\omega)$.

Под задачей распознавания понимается задача построения некоторого решающего правила, которое позволило бы судить о скрытом классе $g(\omega)$ предъявленного объекта на основе анализа вектора его наблюдаемых признаков $\mathbf{x}(\omega) \in \mathbf{R}^n$, т.е. правила вида $\hat{g}(\mathbf{x})$, и “не слишком часто” ошибаться. Мы ограничимся здесь рассмотрением только решающих правил, опирающихся на линейные дискриминантные функции

$$\hat{g}(\mathbf{x}; \mathbf{a}, b) = \begin{cases} 1, & \text{если } \mathbf{a}^T \mathbf{x} + b > 0, \\ -1, & \text{если } \mathbf{a}^T \mathbf{x} + b < 0, \end{cases} \quad (1.2.15)$$

где вектор $\mathbf{a} \in \mathbf{R}^n$ и скаляр $b \in \mathbf{R}$ являются параметрами, полностью определяющими линейную дискриминантную функцию. Заметим, что уравнение $\mathbf{a}^T \mathbf{x} - b = 0$ определяет линейное многообразие размерности $n - 1$, разделяющее в пространстве признаков \mathbf{R}^n области принятия решений в пользу первого и второго класса. Мы, как и ранее, будем называть такое многообразие разделяющей гиперплоскостью, хотя, строго говоря, под гиперплоскостью принято понимать частный случай, когда линейное многообразие размерности на единицу меньше, чем все линейное пространство, является его подпространством, т.е. содержит нулевую точку, что имеет место только в случае $b = 0$.

Пусть (g_j, \mathbf{x}_j) , $j=1, \dots, N$ - обучающая выборка, где индексы классов принимают значения ± 1 . Исходя из принципа минимизации эмпирического риска, нам хотелось бы выбрать такие параметры линейной дискриминантной функции, которые бы обеспечивали бы наименьшее количество ошибок распознавания в пределах обучающей выборке.

Обратим внимание на тот факт, что если некоторая линейная дискриминантная функция с параметрами (\mathbf{a}, b) обеспечивает некоторое вполне определенное значение доли ошибок, то такую же долю ошибок будет обеспечивать целое множество дискриминантных функций с близкими параметрами, что для случая размерности признакового пространства $n=2$ показано на рис. 1.2.8. Тем не менее, очевидно, что значения среднего риска, обеспечиваемые этими, казалось бы эквивалентными с точки зрения обучающей выборки дискриминантными функциями, будут практически

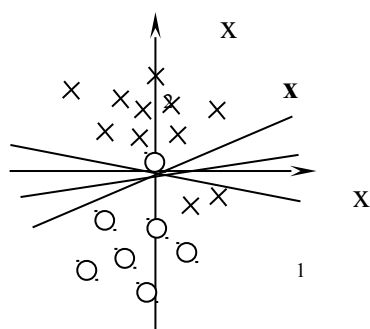
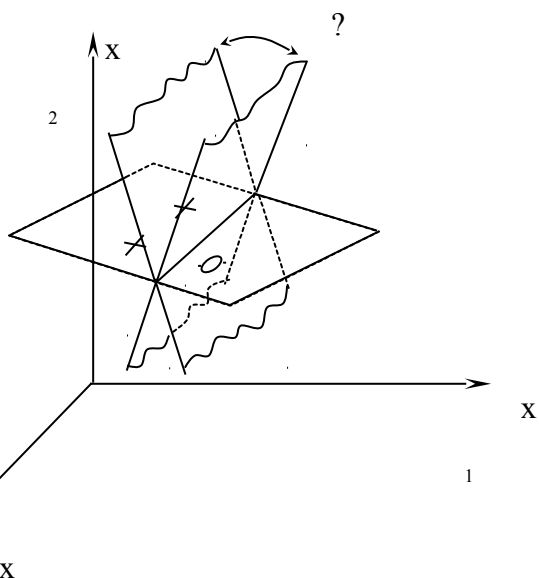


Рисунок 1.2.8 Иллюстрация неоднозначности решения задачи обучения распознаванию образов по обучающей выборке.

всегда различными по отношению к генеральной совокупности. Поскольку задача распознавания всегда решается на конечной выборке, то разговор о эмпирического риска к среднему при увеличении ее размера является малоутешительным. Вопрос о выборе одной дискриминантной функции из множества эквивалентных является очень актуальным.

Этот вопрос становится принципиальным, если размерности признакового пространства велика по сравнению с размером выборки.

Очень часто возникают ситуации, когда число элементов выборки меньше размерности пространства признаков. Тогда элементы выборки образуют в пространстве признаков подпространство (аффинное многообразие). Этот случай проиллюстрирован на рис 1.2.9.



Исходное предположение заключается в том, что в пространстве признаков существуют области не слишком сложной формы, пересекающиеся или непересекающиеся, в которых полностью сосредоточены распределения вероятности, связанные с первым и вторым классом объектов.

Если обучающая выборка имеет число элементов меньше, чем размерность признакового пространства, то такая выборка несет информацию лишь о форме сечения областей обоих классов в

подпространстве, образованном выборкой. Поэтому, строго говоря, такая выборка не позволяет нам принять обоснованное решение о линейной дискриминантной функции во всем признаковом пространстве. Мы можем лишь выбрать форму следа такой разделяющей гиперплоскости в подпространстве выборки. Выбрать же “пространственный наклон” разделяющей гиперплоскости мы не в состоянии, поскольку выборка не содержит никакой информации, на которую можно было бы опереться.

Для того, чтобы выбрать одну разделяющую гиперплоскость из целого пучка возможных, необходимо привлечь некоторую априорную информацию, содержащуюся в выборке.

В нашем случае, выбрав след разделяющей гиперплоскости в подпространстве выборки, мы фактически оценили форму сечений областей классов с тех сторон, которыми они обращены друг к другу. Что же касается формы этих областей вне пространства выборки, то поскольку нет никакой фактической информации, естественно принять, что эти формы такие же. В геометрических терминах такое предположение означает, что разделяющую гиперплоскость надо выбрать так, чтобы она была ортогональна подпространству выборки.

В метрических пространствах это означает, что мы выбираем разделяющую гиперплоскость так, чтобы точки первого и второго класса обучающей выборки были как можно дальше от нее, каждая со своей стороны.

Заметим, что такая идея представляется вполне разумной и для больших выборок. Ведь точки обучающей выборки несут лишь приблизительную информацию о формах областей классов. Нет гарантии, что область данного класса “заканчивается” прямо сразу за крайней точкой обучающей выборки. Поэтому представляется целесообразным на всякий случай отодвинуть разделяющую гиперплоскость от точек как первого, так и второго класса.

Для того, чтобы реализовать эту идею надо выбрать способ измерения удаленности гиперплоскости от выборки объектов определенного класса, т.е. способ измерения того, насколько вся выборка находится по нужную сторону от разделяющей гиперплоскости.

В данном курсе в качестве меры удаленности мы примем удаленность от гиперплоскости “самой плохой” точки выборки.

Сразу же заметим, что такой выбор представляется естественным далеко не всегда, поскольку исходит из того, что выборка не содержит “диких точек”, что на самом деле встречается довольно часто.

Изложенный принцип построения линейной дискриминантной функции реализуется понятием оптимальной разделяющей гиперплоскости для точек двух классов.

1.3. Особенности обучения в условиях малого относительного размера обучающей выборки по сравнению с размерностью пространства признаков

1.3.1. Селекция признаков (сокращение признакового пространства)

Одной из классических задач теории распознавания образов является понижение размерности вектора признаков X . Следует отметить тот факт, что интерес к этой процедуре сохраняется и в последнее время т.к. появление нового поколения быстродействующих ЭВМ, и как следствие, относительная независимость исследователей, разработчиков в области анализа данных, и распознавания образов в частности, от вычислительной сложности не явилась эволюционным решением этой проблемы. Дело в том, что помимо (а) сокращения объема вычислений, отбор признаков, в большей или меньшей степени направлен и на (б) сжатие объема данных, (в) сокращение стоимости сбора данных, (г) улучшение классификации, (д) возможность визуализации многомерных данных [1,13]. Приложения, требующие применения методов отбора признаков встречаются в следующих задачах: (1) Приложения, в которых объединены данные от большого числа датчиков. (2) Объединение многомерных моделей, когда все параметры различных моделей могут быть использованы для классификации; и (3) приложения по основанные на получении (извлечении) данных, где целью является определение скрытых зависимостей между признаками.

Таким образом, очевидно, что выбор признаков играет в распознавании важную роль. Выбор адекватного множества признаков, учитывающий трудности, которые связаны с реализацией процессов выделения или выбора признаков, и обеспечивающий в то же время необходимое качество классификации, представляет собой одну из наиболее трудных задач построения распознающих систем. Для того чтобы облегчить анализ этой задачи в [29] предлагается разделить признаки на три категории: (1) "физические", (2) структурные и (3) математические.

Физические и структурные признаки обычно используются людьми при распознавании образов, поскольку такие признаки легко обнаружить на ощупь, визуально, с помощью других органов чувств. Поскольку органы чувств обучены распознаванию физических и структурных признаков, человек, естественно

пользуется в основном такими признаками при классификации и распознавании. В случае же построения вычислительной системы распознавания образов эффективность таких признаков с точки зрения организации процесса распознавания может существенно снижаться, так как, вообще говоря, в большинстве практических ситуаций довольно сложно имитировать возможности органов чувств человека. С другой стороны, можно создать систему, обеспечивающую выделение математических признаков образов, что может оказаться затруднительным для человека при отсутствии "механической" помощи. Примерами признаков этого типа являются статистические средние, коэффициенты корреляций, характеристические числа и собственные векторы ковариационных матриц, и прочие инвариантные свойства объектов.

Предварительная обработка образов обычно включает решение двух основных задач: преобразование кластеризации и выбор признаков. Основной задачей распознавания образов является построение решающих функций, представляющих некоторые классы. Эти функции должны обеспечивать разделение пространства измерений на области, каждая из которых содержит точки, представляющие образы только одного из рассматриваемых классов. Данное положение приводит к идее преобразований кластеризации, реализуемого в пространстве измерений, для того чтобы обеспечить группировку точек, представляющих выборочные образы одного класса. В результате такого преобразования максимизируется расстояние между множествами и минимизируются внутримножественные расстояния.

Выбор наиболее эффективных признаков позволяет снизить размерность вектора измерений. Выбор признаков можно осуществлять вне связи с качеством схемы классификации. Оптимальный выбор признаков при этом определяется максимизацией или минимизацией некоторого критерия. Такой подход принято считать выбором признаков без учета ограничений. Другой подход связывает выбор признаков с качеством классификации, причем обычно эта связь выражается в терминах вероятности правильного распознавания.

Преобразование кластеризации и упорядочение признаков

Практически всегда измерения характеристик образа, соответствующие отдельным признакам x_j , $j = 1, \dots, L$, не важны в одинаковой степени для задачи классификации. Преобразование кластеризации признакам с меньшей значимостью принято приписывать меньшие веса. Назначение весов признаков можно осуществить посредством линейного преобразования, которое обеспечивает более благоприятную группировку точек в новом, т.н. вторичном пространстве.

Рассмотрим векторы образов \mathbf{a} и \mathbf{b} , который после применения к ним преобразования \mathbf{W} перешли в векторы \mathbf{a}^* и \mathbf{b}^* . Тогда справедливо $\mathbf{a}^* = \mathbf{W}\mathbf{a}$ и $\mathbf{b}^* = \mathbf{W}\mathbf{b}$, где

$$\mathbf{W} = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1L} \\ w_{21} & w_{22} & \dots & w_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ w_{L1} & w_{L2} & \dots & w_{LL} \end{pmatrix},$$

и w_{jk} - весовые коэффициенты.

Таким образом каждый элемент преобразованного вектора образа представляет собой линейную комбинацию элементов исходного вектора. В новом пространстве евклидово расстояние между векторами \mathbf{a}^* и \mathbf{b}^* определяется как

$$D(\mathbf{a}^*, \mathbf{b}^*) = \sqrt{\sum_{j=1}^L (a_j^* - b_j^*)^2} = \sqrt{\sum_{j=1}^L \left[\sum_{i=1}^L w_{ji} (a_i - b_i) \right]^2}. \quad (1.3.1)$$

В тех случаях, когда линейное преобразование сводится к изменению масштабных коэффициентов координатных осей, матрица \mathbf{W} является диагональной, и выражение для евклидова расстояния сводится к

$$D(\mathbf{a}^*, \mathbf{b}^*) = \sqrt{\sum_{j=1}^L w_{jj}^2 (a_j^* - b_j^*)^2}, \quad (1.3.2)$$

где элементы w_{jj} представляют собой весовые коэффициенты при признаках. Задача преобразования кластеризации заключается в том чтобы определить весовые коэффициенты признаков w_{jj} , минимизирующие внутримножественные расстояния внутри классов с учетом определенных ограничений, наложенных на коэффициенты w_{jj} . В качестве подобных ограничений обычно рассматриваются ограничения вида

$\sum_{j=1}^L w_{jj} = 1$ и $\prod_{j=1}^L w_{jj} = 1$. С учетом того факта, что в новом пространстве внутреннее расстояние для множества точек, представляющих образы определяется как

$$\overline{D^2} = 2 \sum_{j=1}^L (w_{jj} \sigma_j)^2, \quad (1.3.3)$$

где σ_j^2 - несмещенная оценка выборочной дисперсии компонент, соответствующих координат x_j , минимизация $\overline{D^2}$ с учетом первого ограничения дает значения весовых коэффициентов

$$w_{jj} = \frac{1}{\sigma_j^2 \sum_{j=1}^L (1/\sigma_j^2)}, \quad (1.3.4)$$

и с учетом второго ограничения

$$w_{jj} = \frac{1}{\sigma_j} \left(\prod_{j=1}^L \sigma_j \right)^{1/L} \quad (1.3.5)$$

Формулы (1.3.4) и (1.3.5) определяют матрицу преобразования \mathbf{W} с учетом введенных выше ограничений. Если векторы образов переводятся из пространства X в пространство X^* с помощью преобразования

$$\mathbf{x}^* = \mathbf{W}\mathbf{x}, \quad (1.3.6)$$

то внутреннее расстояние множества в пространстве X^* минимизируется. После этого требуется повести второе преобразование

$$\mathbf{x}^{**} = \mathbf{A}\mathbf{x}^*$$

с целью выделения компонент, имеющих заданную дисперсию, и обеспечения возможности провести упорядочение и выбор признаков. Это преобразование превращает ковариационную матрицу точек, представляющих образы в пространстве X^{**} в диагональную. Кроме того, для обеспечения неизменности расстояний необходимо наложить условие ортонормированности на матрицу \mathbf{A} .

Пусть $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_L$ - собственные векторы ковариационной матрицы \mathbf{C} и $\lambda_1, \lambda_2, \dots, \lambda_L$ - соответствующие характеристические числа. Элементы ортогональной матрицы преобразования \mathbf{A} выбираются так, что в преобразованном пространстве ковариационная матрица становится диагональной. Этого можно достичь используя l транспонированных собственных векторов ковариационной матрицы \mathbf{C} в качестве строк ортогональной матрицы \mathbf{A} :

$$\mathbf{A} = \begin{pmatrix} \mathbf{e}_1^T \\ \mathbf{e}_2^T \\ \vdots \\ \mathbf{e}_l^T \end{pmatrix}.$$

Таким образом, размерность исходного признакового пространства понижена до $l < L$. После этого можно определить матрицу весов \mathbf{W} так, чтобы расстояние $\overline{D^2}$ принимало при выполнении заданных ограничений экстремальное значение. Для ограничения вида $\prod_{j=1}^L w_{jj} = 1$ матрица \mathbf{W} определяется следующим образом:

$$\mathbf{W} = \left(\prod_{j=1}^l \lambda_j \right)^{1/2m} \begin{pmatrix} \lambda_1^{-1/2} & 0 & \dots & 0 \\ 0 & \lambda_2^{-1/2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_l^{-1/2} \end{pmatrix}. \quad (1.3.7)$$

Следовательно, если мы хотим минимизировать внутреннее расстояние множества, то в качестве векторов признаков следует выбирать собственные векторы,

соответствующие наименьшим характеристическим числам ковариационной матрицы \mathbf{C} .

При ограничении $\sum_{j=1}^L w_{jj} = 1$ матрица \mathbf{W} определяется как

$$\mathbf{W} = \left(\sum_{j=1}^l \frac{1}{\lambda_j} \right)^{-1} \begin{pmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0 \\ 0 & \frac{1}{\lambda_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \vdots & \frac{1}{\lambda_l} \end{pmatrix} \quad (1.3.8)$$

Таким образом, значение внутреннего расстояния множества достигнет глобального минимума, если в качестве характеристических чисел λ_j выбраны l наименьших из L характеристических чисел ковариационной матрицы и матрица преобразования \mathbf{A} составлена из l соответствующих собственных векторов.

Выбор признаков при помощи минимизации энтропии

В теории статистики статистическую неопределенность принято выражать таким понятием как энтропия. Энтропия представляет собой статистическую неопределенность. Хорошей мерой внутреннего разнообразия для заданного семейства объектов распознавания служит энтропия совокупности, определяемая как

$$H = -E\{\ln p\}, \quad (1.3.9)$$

где P - плотность вероятности совокупности образов, а E - оператор математического ожидания плотности P . Понятие энтропии удобно использовать в качестве критерия при организации оптимального выбора признаков. Признаки, уменьшающие неопределенность заданной ситуации, считаются более информативными, чем те, которые приводят к противоположному результату. Таким образом, если считать энтропию мерой неопределенности, то разумным правилом является выбор признаков, обеспечивающих минимизацию энтропии рассматриваемых классов. Поскольку это правило эквивалентно минимизации дисперсии в различных совокупностях образов, то вполне можно ожидать, что соответствующая процедура будет обладать кластеризационными свойствами.

Пусть существует m классов, характеризующихся плотностями распределения $p(\mathbf{x} | \omega_1)$, $p(\mathbf{x} | \omega_2)$, ..., $p(\mathbf{x} | \omega_m)$. В соответствии с (1.3.9) энтропия i -ой совокупности образов определяется как

$$H_i = - \int_{\mathbf{x}} p(\mathbf{x} | \omega_i) \ln p(\mathbf{x} | \omega_i) d\mathbf{x}. \quad (1.3.10)$$

Очевидно, что при $p(\mathbf{x}|\omega_i)=1$, т.е. при отсутствии неопределенности, имеем $H_i = 0$.

Данный метод предлагает, что каждая из M совокупностей образов характеризуется плотностью нормального распределения с математическим ожиданиями \mathbf{m}_i и ковариационными матрицами \mathbf{C}_i , соответственно для i -ой совокупности образов. Кроме того, предполагается, что ковариационные матрицы, описывающие статистические характеристики всех m классов идентичны.

Основная идея заключается в определении матрицы линейного преобразования \mathbf{A} , переводящей заданные векторы образов в новые векторы меньшей размерности. Это преобразование можно представить как

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \quad (1.3.11)$$

причем матрица преобразования отыскивается при помощи минимизации энтропий совокупности образов, входящих в рассматриваемые классы. Предполагается, что вектор \mathbf{X} - размерности L , \mathbf{Y} - отображенный вектор, размерности l , $l < L$ и \mathbf{A} - матрица размерности $l \times L$. Строками матрицы \mathbf{A} служат l выбранных векторов $\mathbf{a}_1^T, \mathbf{a}_2^T, \dots, \mathbf{a}_l^T$, представляющих собой вектор-строки. Таким образом, матрица \mathbf{A} имеет вид

$$\mathbf{A} = \begin{pmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_l^T \end{pmatrix} \quad (1.3.12)$$

Задача состоит в определении такого способа выбора l векторов признаков, чтобы вектор \mathbf{X} преобразовывался в изображение \mathbf{Y} и одновременно минимизировалась величина энтропии, определяемая (1.3.10).

Многомерное нормальное распределение полностью определяется вектором математического ожидания и ковариационной матрицей, которая в свою очередь характеризуется характеристическими числами и собственными векторами. Последние можно рассматривать как векторы, представляющие свойства рассматриваемых образов. Часть из этих векторов свойств содержит меньше информации, ценной для распознавания, чем другие векторы, и поэтому ими можно пренебречь. Это явление приводит к процедуре выбора признаков, предусматривающей использование наиболее важных свойств в качестве векторов-признаков. Такие векторы-признаки можно затем использовать для формирования матрицы преобразования \mathbf{A} . В [27, 57] показано, что функция энтропии H_i^* принимает минимальное значение, если матрица преобразования \mathbf{A} составлена из l нормированных собственных векторов, соответствующих наименьшим характеристическим числам ковариационной матрицы

С . Применяя этот результат, надо иметь в виду, что число векторов, используемых для формирования матрицы \mathbf{A} , должно быть достаточно большим, чтобы изображения несли достаточное количество различительной информации.

Преобразование Карунена-Лоэва

Основанием применения дискретного разложения Карунена-Лоэва [27] в качестве средства выбора признаков является наличие у него следующих оптимальных свойств.

Во-первых, оно минимизирует среднеквадратичную ошибку при использовании лишь конечного числа базисных функций в разложении

$$\mathbf{x}_i = \sum_{j=1}^L c_{ij} \boldsymbol{\phi}_j , \quad (1.3.13)$$

где

$$x_i = \begin{pmatrix} x_i(t_1) \\ x_i(t_2) \\ \vdots \\ x_i(t_L) \end{pmatrix}, \quad \boldsymbol{\phi}_j = \begin{pmatrix} \phi_j(t_1) \\ \phi_j(t_2) \\ \vdots \\ \phi_j(t_L) \end{pmatrix}, \quad (1.3.14)$$

L - количество наблюдений для функции $x_i(t)$, осуществленных в интервале $[T_1, T_2]$, $\phi_j(t)$ - базисные функции, в качестве которых используется множество детерминированных ортонормированных функций, заданных на интервале $[T_1, T_2]$. Причем относительно коэффициентов предполагается, что они удовлетворяют условию $E\{c_{ij}\} = 0$.

Во-вторых, данное преобразование минимизирует функцию энтропии, выраженную через дисперсии коэффициентов разложения.

Принцип минимизации среднеквадратичной ошибки предполагает, что разложение Карунена-Лоэва минимизирует ошибку аппроксимации при использовании в приведенном разложении числа базисных векторов, меньшего L . Принцип минимизации энтропии обеспечивает искомые эффекты кластеризации, описанные выше.

Применение дискретного разложения Карунена-Лоэва при выборе признаков можно рассматривать как линейное преобразование. Если

$$\boldsymbol{\Phi} = (\boldsymbol{\phi}_1 \boldsymbol{\phi}_2 \dots \boldsymbol{\phi}_l), \quad l < L \quad (1.3.15)$$

- матрица преобразования, то преобразованные образы являются коэффициентами разложения Карунена-Лоэва, т.е. для любого образа \mathbf{x}_i , принадлежащего классу ω_i , выполняется

$$\mathbf{c}_i = \Phi^T \mathbf{x}_i. \quad (1.3.16)$$

Поскольку Φ^T - матрица размера $l \times L$ и \mathbf{X} - L -мерный вектор, то \mathbf{c}_i при $l < L$ представляют собой изображения, имеющие размерность, меньшую чем L .

Условия оптимальности разложения Карунена-Лоэва выполняются, если в качестве столбцов матрицы преобразования Φ выбраны l нормированных собственных векторов, соответствующих наибольшим характеристическим числам корреляционной матрицы \mathbf{R} . В таком случае для любого вектора \mathbf{X} его изображения меньшей размерности определяются как

$$\mathbf{y} = \mathbf{A}\mathbf{x},$$

где \mathbf{A} - матрица преобразования, строками которой служат нормированные собственные векторы, соответствующие наибольшим характеристическим числам корреляционной матрицы \mathbf{R} .

Для того, чтобы применение разложения Карунена-Лоэва приводило к получению оптимальных результатов, необходимо выполнение условия $E\{\mathbf{x}_i\} = 0$, которое выполняется автоматически, если отдельные классы характеризуются нулевыми математическими ожиданиями. Однако надо иметь в виду, что за исключением непосредственно этапа обучения, вообще говоря, отсутствуют сведения о принадлежности образа к определенному классу.

Хотя предположение об идентичности математических ожиданий всех совокупностей образов ограничивает возможности применения разложения Карунена-Лоэва, не следует считать, что этот подход к выбору признаков не имеет достоинств. Допущения такого рода характерны для большинства статистических методов анализа.

1.3.2. Стабилизация классификаторов

В дискриминантном анализе часто встречаются задачи с малыми размерами обучающих выборок. В практических задачах ситуация складывается так, что должно быть рассмотрено большое число различных измерений при обучении на объектах реального мира. Как результат, во многих областях применения массивы данных могут иметь большое число признаков, например 100-200 и более. Статистическая зависимость между отдельными компонентами в таких данных часто имеет нелинейный характер. Следовательно, необходимо использовать дополнительные полиномиальные признаки для четкого оценивания упомянутых нелинейных зависимостей [52]. В этих случаях размерность признакового пространства становится крайне большой. Если число объектов, по которым необходимо провести

обучение, все же ограничено, может возникнуть большие трудности при построении дискриминантной функции [38,47,48].

Стандартные классические статистические методы требуют обращения ковариационной матрицы, например, линейная дискриминантная функция Фишера (ЛДФ) [28,35]:

$$g_F(\mathbf{x}) = [x - \frac{1}{2}(\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)})]S^{-1}(\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)}) \quad (1.3.17)$$

в которой S представляет собой оценку максимума правдоподобия ковариационной матрицы \mathbf{C} размером $n \times n$, \mathbf{x} представляет собой P -размерный вектор, который необходимо классифицировать, $\bar{\mathbf{x}}^{(i)}$ - вектор среднего по выборке i -го класса. Необходимо определить такое решающее правило (\mathbf{a}, a_0) , чтобы выполнялось

$$g_F(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + a_0 = d, \quad (1.3.18)$$

где $\mathbf{x} \in X$ и d принимает положительные значения для объектов первого класса и отрицательные для второго.

Прямые вычисления невозможны в случае, когда число признаков n превышает число объектов N [46]. Для больших размеров признаков при уменьшении n ожидаемая вероятность ошибок классификаций сильно возрастает [47]. Для преодоления этих проблем существует несколько различных методов.

Один заключается в снижении числа признаков до $n < N$, используя знания нескольких экспертов или с помощью методов отбора и экстракции наиболее информативных признаков, например, как это было описано в предыдущем разделе. Второй подход заключается в модификации стандартной ЛДФ каким либо способом. Целью настоящего раздела и является обзор некоторых таких возможностей. Главные из них – это применение функции псевдо-линейного разделения Фишера, которая рассматривает классификаторы в подпространстве, определенном некоторыми доступными объектами [34] и классификатор ближайших средних, который игнорирует ковариации.

Третий подход состоит в способе стабилизации, использующем различные методы регуляризации, такие как ридж-оценка ковариационной матрицы и бэггинг (бутстрэп, совмещенный с агрегированием [37]). Часто оптимальная ридж-оценка ковариационной матрицы или бэггинг могут улучшить работу линейных классификаторов. Иногда подобные методы стабилизации не только не помогают, но и более того даже увеличивают ошибку классификации. Это четко связано со стабильностью классификаторов на специфических данных.

Одной из простейших модификаций процедуры ЛДФ является классификатор ближайших средних

$$g_{NM}(\mathbf{x}) = [x - \frac{1}{2}(\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)})]^T (\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)}) \quad (1.3.19)$$

Следовательно, классификатор ближайших средних генерирует перпендикулярную биссектрису между средними по классам, и таким образом, строит оптимальный линейный классификатор для классов с одинаковыми нормальными сферическими распределениями. Преимущество этого классификатора заключается в относительной нечувствительности к размеру выборки [47]. Однако такой классификатор не учитывает разницы между дисперсиями и ковариациями.

Другая модификация линейной дискриминантной функции Фишера (1.3.17), позволяет преодолеть обращение вырожденной ковариационной матрицы для выборок малого размера $n < N$, это так называемый псевдо линейный дискриминатор Фишера (ПЛДФ) [34]. Здесь непосредственное решение для (1.3.18) получается как (используя расширенный вектор):

$$g_{PF}(\mathbf{x}) = (\mathbf{a}, a_0)^T (\mathbf{x}, 1) = (\mathbf{x}, 1)(\mathbf{X}, \mathbf{I})^{-1} \mathbf{d} \quad (1.3.20)$$

где, $(\mathbf{x}, 1)$ - расширенный вектор, который необходимо классифицировать и (\mathbf{X}, \mathbf{I}) - расширенная матрица данных. Процедура обращения матрицы $(\mathbf{X}, \mathbf{I})^{-1}$ представляет собой псевдо обращение Мур-Пенроса, которое дает минимальное нормированное решение. Перед обращением данные необходимо сдвинуть таким образом, чтобы средние значения по признакам были равны нулю. Этот метод близок к методу декомпозиции вырожденных значений.

Для значений $N \geq n$ ПЛДФ, максимизируя расстояния между выборками разных классов, идентичен ЛДФ (1.3.17). Однако для значений $N < n$, ПЛДФ находит линейное подпространство, где располагаются все данные, и в этом подпространстве оцениваются средние значения и ковариационные матрицы и строит линейное разделяющее правило, которое ортогонально этому подпространству во всех других направлениях, в которых нет заданных объектов.

Постоянное расстояние для всех обучающих выборок, найденное ПЛДФ может быть увеличено коррекцией обучающей выборки, позволяющей некоторым выборкам иметь большее взаимное расстояние. Такая процедура, названная классификатором малых выборок предложена в [34].

Ридж-оценка ковариационной матрицы

Хорошо известный метод обращения вырожденной ковариационной матрицы, использованный при построении стандартного ЛДФ (1.3.17), заключается в добавлении некоторых постоянных значений к диагональным элементам, оцениваемой ковариационной матрицы

$$\mathbf{C}_R = \mathbf{C} + \lambda \mathbf{I}, \quad (1.3.21)$$

и \mathbf{I} - есть единичная матрица размером $n \times n$, и λ - параметр регуляризации.

Новая оценка \mathbf{C}_R называется "реберной" оценкой ковариационной матрицы, термин, который был заимствован из регрессионного анализа [1]. Этот подход называется регуляризованным дискриминантным анализом. Модификация (1.3.21) дает нам "реберную" или регуляризованную дискриминантную функцию

$$g_F(\mathbf{x}) = [x - \frac{1}{2}(\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)})](\mathbf{S} + \lambda \mathbf{I})^{-1}(\bar{\mathbf{x}}^{(1)} - \bar{\mathbf{x}}^{(2)}) \quad (1.3.22)$$

Очевидно, что при $\lambda \rightarrow \infty$, теряются значения дисперсий. В этом случае классификатор (1.3.22) является приближением к процедуре ближайших средних (1.3.19). Одновременно обобщенная ошибка может быть существенным образом уменьшена. Малые значения параметра регуляризации λ могут быть полезны для стабилизации решения. Очень малые значения λ могут быть не достаточно эффективны. Обычно для выбора приемлемого параметра регуляризации используется два метода: cross validation [40] и бутстрэпа. Они оба требуют большого объема вычислений, и оба не достаточно точны при малом размере обучающей выборки. Смотри, например, Фридман [36], Раудис и Скурихина [50].

Бэггинг

Другим методом стабилизации является бэггинг, основанный на бутстрэпе, и соединяющий концепции предложенные Бриманом. Предлагается строить бэггинг-классификатор посредством усреднения параметров линейного классификатора, построенного на нескольких повторениях процедуры бутстрэпа. Случайный выбор с замещением N образцов из набора из N штук называется дублированием бутстрэпа. Из каждой такой копии и строится "бутстрэпная" версия классификатора. Усреднение этих версий как раз и дает бэггинг-классификатор. Бримэн показал что бэггинг может уменьшить ошибку линейной регрессии и классификации. Отмечается, что такой метод полезен только для нестабильных процедур. Для стабильных методов он может даже ухудшить результат работы классификатора.

Таким образом, вопрос стабильности и нестабильности классификатора на специфических данных очень важен. Для классификатора можно предсказать меру нестабильности на разных данных, необходимость использования бэггинг или ридж-оценки ковариационной матрицы.

1.4.1 Основные задачи исследования

Для реализации предлагаемого принципа необходимости учета априорной информации об исследуемых данных при построении решающих правил распознавания образов в данной работе ставятся следующие основные задачи:

1. Разработать эффективный алгоритм обучения в признаковом пространстве большой размерности по сравнению с объемом выборки.
2. Разработать эффективный алгоритм обучения для классов задач, в которых легко удастся непосредственно вычислить степень «непохожести» любых двух объектов, но трудно указать набор осмысленных характеристик объектов, которые могли бы служить координатными осями пространства признаков.
3. Разработать комплекс вспомогательных процедур, направленных на отображение многомерных данных и решающего правила распознавания.
4. Исследовать работоспособность предложенных алгоритмов на модельных и реальных данных.
5. Создать программно-алгоритмический комплекс, реализующий разработанные алгоритмы и обеспечивающий наглядное представление данных и результатов обучения и распознавания.

Конструируемые процедуры должны реализовывать схему обучения распознаванию образов с учителем.

2. Обучение распознавания образов в линейных пространствах признаков.

2.1 Концепция оптимальной разделяющей гиперплоскости.

Рассмотрим теперь концепцию оптимальной разделяющей гиперплоскости. Пусть предъявлена обучающая выборка (\mathbf{x}_j, g_j) , $\mathbf{x}_j \in \mathbb{R}^{n+1}$, $g_j \in \{1, -1\}$ $j = 1, \dots, N$. По своей идее искомая разделяющая гиперплоскость $\mathbf{a}^T \mathbf{x} + a^{(0)} = 0$ призвана как можно лучше отделять друг от друга точки первого и второго класса. Поскольку единственное, что известно о границах областей классов в пространстве признаков, полностью содержится в обучающей выборке, то представляется естественным выбрать вектор $\mathbf{a} \in \mathbb{R}^n$ и скаляр $a^{(0)}$ так, чтобы

$$\mathbf{a}^T \mathbf{x}_j + a^{(0)} \begin{cases} > 0, & \text{если } g_j = 1, \\ < 0, & \text{если } g_j = -1. \end{cases}$$

Такая пара $(\mathbf{a}, a^{(0)})$ существует, если выпуклые оболочки подвыборок первого и второго класса в обучающей совокупности не пересекаются, причем в этом случае существует множество таких пар. Среди них естественно выбрать ту, которая определяет гиперплоскость, наиболее удаленную от краев подвыборок. Заметим, что нетривиальна лишь задача поиска направляющего вектора $\mathbf{a} \in \mathbb{R}^n$, качество которого естественно оценивать величиной остаточного “зазора”:

$$J(\mathbf{a}) = \min_{j: g_j=1} \mathbf{a}^T \mathbf{x}_j - \max_{j: g_j=-1} \mathbf{a}^T \mathbf{x}_j. \quad (2.1.1)$$

Нам будет удобно ввести специальное обозначение для скалярного произведения $\mathbf{a}^T \mathbf{x}_j$, рассматривая его как промежуточный критерий в составе полного критерия (1.2.16), количественно характеризующий определенную точку первого или второго класса относительно всей подвыборки этого класса. Пока мы примем оба критерия одинаковыми

$$Q^{(1)}(j, \mathbf{a}) = Q^{(-1)}(j, \mathbf{a}) = \mathbf{a}^T \mathbf{x}_j, \quad (2.1.2)$$

однако ниже мы обобщим эти понятия.

С учетом принятых обозначений критерий качества направляющего вектора примет вид

$$J(\mathbf{a}) = \min_{j: g_j=1} Q^{(1)}(j, \mathbf{a}) - \max_{j: g_j=-1} Q^{(-1)}(j, \mathbf{a}). \quad (2.1.3)$$

После того, как вектор \mathbf{a} выбран, оптимальное значение скаляра $a^{(0)}$ определяется как среднее значение

$$a^{(0)} = \frac{1}{2} \left(\min_{j: g_j=1} Q^{(1)}(j, \mathbf{a}) + \max_{j: g_j=-1} Q^{(-1)}(j, \mathbf{a}) \right). \quad (2.1.4)$$

Заметим, что существенно только соотношение между значениями элементов вектора \mathbf{a} , но не их величины, поэтому направляющий вектор разделяющей гиперплоскости достаточно выбирать среди векторов единичной нормы $\|\mathbf{a}\| = (\mathbf{a}^T \mathbf{a})^{1/2} = 1$. Таким образом, обучение сводится к решению задачи

$$\hat{\mathbf{a}} = \operatorname{argmax}_{\mathbf{a} \in R^n, \|\mathbf{a}\|=1} J(\mathbf{a}) \quad (2.1.5).$$

Такая задача остается полностью корректной с точки зрения конечной цели обучения и в случае пересекающихся выпуклых оболочек подвыборок первого и второго класса. При этом, наибольшее возможное значение критерия будет отрицательным $J(\hat{\mathbf{a}}) < 0$, и направляющий вектор, найденный согласно (2.1.2), (2.1.3) и (2.1.1), будет определять гиперплоскость, обеспечивающую наименьший по абсолютной величине остаточный пространственный дефицит разделения подвыборок (рис. 2.1.1). Такая гиперплоскость называется оптимальной. В.Н. Вапник, опираясь на выпуклость критерия $J(\mathbf{a})$, показывает, что оптимальная гиперплоскость единственна.

Заметим, что задача (2.1.5) представляет собой задачу максимизации кусочно-линейной функции при квадратичном ограничении типа равенства. Впрочем, возможны и другие эквивалентные представления этой задачи. В частности, она может быть переформулирована как задача минимизации квадратичной функции при совокупности ограничений в виде линейных неравенств [55].

Можно показать, что каковы бы ни были обучающие подвыборки первого и второго класса, из них всегда можно удалить часть точек так, что оптимальной решение $\hat{\mathbf{a}}$ для оставшихся точек будет в точности таким же, как и для выборки в целом. Минимальное число точек, которое надо оставить, чтобы получающаяся гиперплоскость не изменилась, зависит от конкретной конфигурации подвыборок, но оно всегда не меньше двух, по одной точке первого и второго класса, и не больше $n+1$, т.е. на единицу больше исходной размерности n пространства признаков. Именно эти точки подвыборок и определяют оптимальную разделяющую гиперплоскость, она как бы “опирается” на них, в силу чего такие точки называют опорными. Это самые крайние точки подвыборок с тех сторон, которыми они обращены друг к другу в R^n .

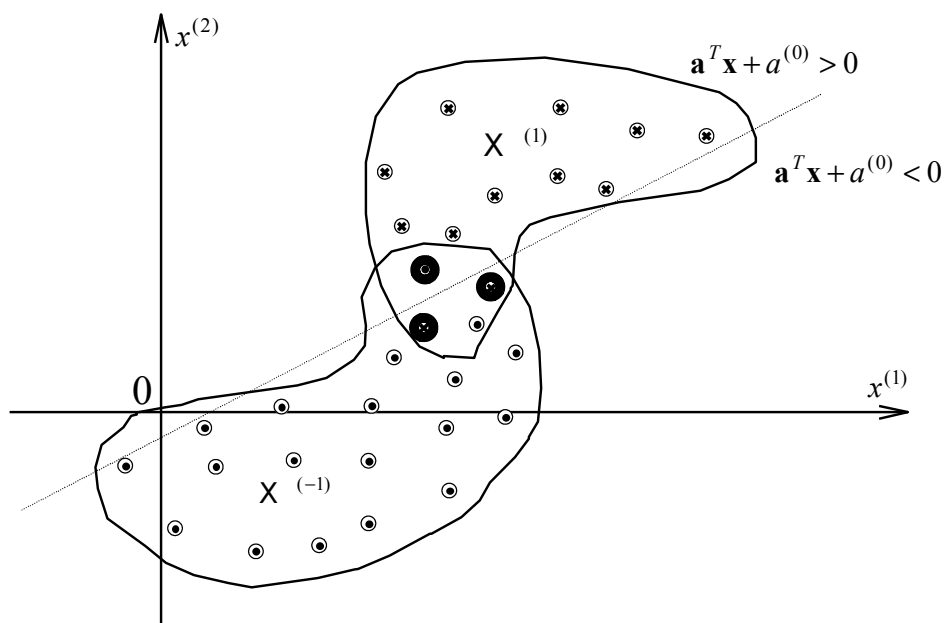


Рисунок. 2.1.1. Гиперплоскость, оптимальная в смысле наилучшего разделения точек первого и второго класса в обучающей выборке; выделены т.н. опорные точки, только на которые фактически и опирается оптимальная гиперплоскость.

Таким образом, неотъемлемой особенностью концепции оптимальной разделяющей гиперплоскости как общей стратегии обучения является то обстоятельство, что обучение опирается только на очень малую часть выборки. Если повторить обучение еще раз, то при малом размере подвыборок первого и второго класса их крайние точки могут “лечь” существенно по-другому, и оптимальная разделяющая гиперплоскость приобретет другой “наклон”. Эта особенность оптимальной разделяющей гиперплоскости наглядно иллюстрируется примером на рис. 2.1.1.

Можно дать и другую интерпретацию эффекту высокой чувствительности оптимальной разделяющей гиперплоскости к конфигурации точек в обучающей выборке. В конечном итоге, мы хотели бы построить гиперплоскость, оптимальную по отношению к истинной форме областей классов $X^{(1)}$ и $X^{(-1)}$. Абсолютно вся информация, которая для этого нужна, содержится в функции

$$J^*(\mathbf{a}) = \min_{\mathbf{x} \in X^{(1)}} \mathbf{a}^T \mathbf{x} - \max_{\mathbf{x} \in X^{(-1)}} \mathbf{a}^T \mathbf{x},$$

которую естественно назвать функцией линейной разделимости классов. Эта функция нам недоступна, и мы пользуемся ее кусочно-линейной оценкой $J(\mathbf{a})$ (2.1.2), (2.1.3), измеренной только в очень редких точках в пространстве направляющих векторов $\mathbf{a} \in \mathbb{R}^n$, число и расположение которых определяются конфигурацией опорных векторов выборки в пространстве признаков. Максимизацию этой функции мы также проводим только среди этих самых опорных точек, как это схематически иллюстрирует рис. 2.1.2. Как следствие, оцененная точка максимума в

пространстве направляющих векторов, как правило, будет существенно отличаться от искомой “истинной” точки \mathbf{a}^* .

В следующем разделе мы рассмотрим способ компенсации высокой чувствительности оптимальной разделяющей гиперплоскости к вариабельности конфигурации опорных точек обучающей выборки, основанный на более общей версии понятий оптимальной разделяющей гиперплоскости и множества опорных точек, позволяющей вовлечь в процесс формирования разделяющей гиперплоскости, значительно большую часть точек выборки.

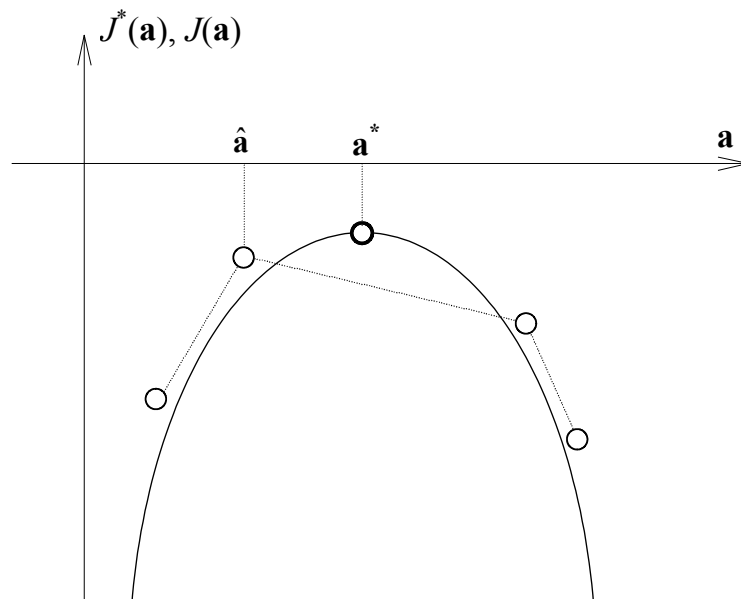


Рисунок. 2.1.2. Схематическое изображение функции линейной разделимости классов $J^*(\mathbf{a})$ и ее кусочно-линейной оценки $J(\mathbf{a})$.

Следует отметить, что алгоритм построения оптимальной разделяющей гиперплоскости был предложен еще Б.Н. Козинцом [3]. Однако, его процедура работает лишь с непересекающимися выпуклыми оболочками двух множеств. Причем итерационный процесс поиска такой гиперплоскости останавливается на основании достаточно эвристического критерия, определяющего расстояние (зазор) между точками, определяющими искомую границу. Хотя надо добавить, что в случае непересекающихся множеств не составляет труда на основании результатов работы процедуры Б.Н. Козинца явно указать опорные векторы.

2.2 Метод опорных векторов и алгоритм обучения распознаванию для двух классов

Пусть обучающая совокупность содержит N объектов двух классов, представленных векторами их действительных признаков $\mathbf{x}_j \in \mathbf{R}^n$ и индексами классов $g_j \in \{1, -1\}$, $j = 1, \dots, N$.

1.2.3 Первая форма задачи построения оптимальной разделяющей гиперплоскости (общая для разделимых и неразделимых объектов двух классов)

Предположим, что объекты классов 1 и -1 линейно разделимы. Тогда существует гиперплоскость $\mathbf{a}^T \mathbf{x} + b = 0$, такая, что

$$\mathbf{a}^T \mathbf{x}_j + b \geq \xi \text{ при } g_j = 1 \text{ и } \mathbf{a}^T \mathbf{x}_j + b \leq -\xi \text{ при } g_j = -1, \quad j = 1, \dots, N, \quad (2.2.1)$$

где $\xi > 0$.

Оптимальной называется такая гиперплоскость, для которой зазор ξ является наибольшим среди всех гиперплоскостей с направляющими векторами единичной нормы:

$$\xi \rightarrow \max \text{ при ограничениях (1) и } \mathbf{a}^T \mathbf{a} = 1$$

или, в более конструктивном виде,

$$J(\mathbf{a}) = \min_{j: g_j=1} \mathbf{a}^T \mathbf{x}_j - \max_{j: g_j=-1} \mathbf{a}^T \mathbf{x}_j \rightarrow \max \text{ при ограничении } \mathbf{a}^T \mathbf{a} = 1. \quad (2.2.2)$$

Предположим теперь, что объекты классов 1 и -1 линейно неразделимы. В этом случае для любой гиперплоскости $\mathbf{a}^T \mathbf{x} + b = 0$ существуют точки класса 1, в которых $\mathbf{a}^T \mathbf{x}_j + b < 0$, и точки класса -1, в которых $\mathbf{a}^T \mathbf{x}_j + b > 0$, так что не существует гиперплоскости, удовлетворяющей условиям (2.2.1) с каким бы то ни было положительным зазором ξ . Какую бы гиперплоскость мы ни выбрали, условия (2.2.1) будут выполняться лишь для некоторой величины $\xi < 0$.

Естественно оптимальной называть такую гиперплоскость, которая обеспечивает выполнение таких неравенств с наименьшим по абсолютной величине “обратным” зазором, т.е. с наибольшим значением ξ , только это наибольшее значение уже не сможет стать положительным. Таким образом, задача поиска оптимальной разделяющей гиперплоскости по-прежнему формально выражается условием (2.2.2).

Задача (2.2.2) представляет собой задачу минимизации кусочно линейной функции $J(\mathbf{a})$ на сфере $\mathbf{a}^T \mathbf{a} = 1$. Хотя сама целевая функция выпукла, область поиска выпуклой не является, поэтому задача неудобна для численного решения.

2.2.2. Вторая форма задачи построения оптимальной разделяющей гиперплоскости (разная для разделимых и неразделимых объектов двух классов)

Пусть объекты классов 1 и -1 линейно разделимы. Очевидно, что изменением масштаба оси \mathbf{a} всегда можно сделать правые части неравенств (2.2.1) равными, соответственно, 1 и -1. Для этого достаточно разделить оба неравенства на ξ

$$\frac{1}{\xi} \mathbf{a}^T \mathbf{x}_j + \frac{1}{\xi} b \geq 1, \quad \frac{1}{\xi} \mathbf{a}^T \mathbf{x}_j + \frac{1}{\xi} b \leq -1$$

и принять $\frac{1}{\xi} \mathbf{a}$ и $\frac{1}{\xi} b$ в качестве новых вектора \mathbf{a} и порога b :

$$\mathbf{a}^T \mathbf{x}_j + b \geq 1 \text{ при } g_j = 1 \text{ и } \mathbf{a}^T \mathbf{x} + b \leq -1 \text{ при } g_j = -1. \quad (2.2.3)$$

Эти неравенства можно записать в более компактной форме, умножив обе части второго неравенства на -1, поменяв при этом, соответственно, его знак:

$-\mathbf{a}^T \mathbf{x} - b \geq 1$ при $g_j = -1$. Тогда оба неравенства можно заменить одним общим неравенством:

$$g_j (\mathbf{a}^T \mathbf{x}_j + b) \geq 1, \quad j = 1, \dots, N.$$

После деления на ξ новый вектор \mathbf{a} будет в ξ раз короче прежнего, норма которого была равна 1. Чем больше ξ , т.е. чем лучше была прежняя гиперплоскость, тем меньше будет норма нового вектора \mathbf{a} . Таким образом, задача построения оптимальной разделяющей гиперплоскости для линейно разделимых объектов классов 1 и -1 принимает следующий вид:

$$\mathbf{a}^T \mathbf{a} \rightarrow \min \text{ при ограничениях } g_j (\mathbf{a}^T \mathbf{x}_j + b) \geq 1, \quad j = 1, \dots, N. \quad (2.2.5)$$

Это задача минимизации квадратичной функции при линейных ограничениях типа неравенств, т.е. классическая задача квадратичного программирования. Минимальное значение $1/\xi^2 = \mathbf{a}^T \mathbf{a}$ указывает максимальную возможную величину ξ зазора между гиперплоскостью и векторами первого и второго классов, безошибочно разделяемыми этой гиперплоскостью, если вернуться к исходным параметрам $\mathbf{a} = \xi \mathbf{a}$ и $b = \xi b$, удовлетворяющим условию $\mathbf{a}^T \mathbf{a} = 1$.

При неразделимых совокупностях объектов классов 1 и -1 множество, определяемое ограничениями (2.2.4), будет пустым, т.е. задача (2.2.5) не будет иметь решения. В этом случае неравенства (1) могут быть выполнены, как мы уже говорили, только при отрицательном значении ξ . Если по-прежнему считать ξ положительной величиной, то их надо заменить неравенствами

$$\mathbf{a}^T \mathbf{x}_j + b \geq -\xi \text{ при } g_j = 1 \text{ и } \mathbf{a}^T \mathbf{x} + b \leq \xi \text{ при } g_j = -1, \quad j = 1, \dots, N. \quad (2.2.6)$$

Гиперплоскость тем лучше, чем меньше значение ξ . Разделив оба неравенства на ξ и изменив тем самым масштаб оси \mathbf{a} , всегда можно сделать правые части неравенств (6) равными, соответственно, -1 и 1

$$\frac{1}{\xi} \mathbf{a}^T \mathbf{x}_j + \frac{1}{\xi} b \geq -1, \quad \frac{1}{\xi} \mathbf{a}^T \mathbf{x}_j + \frac{1}{\xi} b \leq 1,$$

или, приняв $\frac{1}{\xi} \mathbf{a}$ и $\frac{1}{\xi} b$ в качестве новых вектора \mathbf{a} и порога b ,

$$\mathbf{a}^T \mathbf{x}_j + b \geq -1 \text{ при } g_j = 1 \text{ и } \mathbf{a}^T \mathbf{x} + b \leq 1 \text{ при } g_j = -1.$$

Эти два неравенства эквивалентны одному неравенству

$$g_j(\mathbf{a}^T \mathbf{x}_j + b) \geq -1, \quad j = 1, \dots, N.$$

Чем меньше значение ξ и, соответственно, больше значение \mathbf{a} , тем лучше гиперплоскость, поэтому задача построения оптимальной разделяющей гиперплоскости для линейно неразделимых объектов классов 1 и -1 может быть записана в виде:

$$\mathbf{a}^T \mathbf{a} \rightarrow \max \text{ при ограничениях } g_j(\mathbf{a}^T \mathbf{x}_j + b) \geq -1, \quad j = 1, \dots, N. \quad (2.2.8)$$

Максимальное значение $1/\xi^2 = \mathbf{a}^T \mathbf{a}$ дает минимальную величину остаточного дефекта ξ , с которым гиперплоскость с исходными параметрами $\mathbf{a} = \xi \mathbf{a}$ и $b = \xi b$, удовлетворяющими условию $\mathbf{a}^T \mathbf{a} = 1$, разделяет векторы первого и второго классов, не разделимые никакой гиперплоскостью без ошибки.

2.2.3. Третья форма задачи построения оптимальной разделяющей гиперплоскости (разная для разделимых и неразделимых объектов двух классов)

Пусть объекты двух классов линейно разделимы. Задача (2.2.5) есть задача минимизации квадратичной функции при линейных ограничениях типа неравенств, т.е. классическая задача квадратичного программирования. Ей соответствует функция Лагранжа

$$L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N) = \frac{1}{2} \mathbf{a}^T \mathbf{a} - \sum_{j=1}^N \lambda_j [g_j(\mathbf{a}^T \mathbf{x}_j + b) - 1], \quad (2.2.9)$$

где для удобства дальнейших выкладок принят коэффициент $1/2$ перед целевой функцией. $\lambda_j \geq 0$, $j = 1, \dots, N$ – неотрицательные множители Лагранжа. Решением задачи является седловая точка функции Лагранжа:

$$L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N) \rightarrow \min \text{ по } \mathbf{a}, b,$$

$$L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N) \rightarrow \max \text{ по } \lambda_1, \dots, \lambda_N,$$

при ограничениях

$$\lambda_j \geq 0, \quad j = 1, \dots, N.$$

Первое из этих условий (2.2.10) дает

$$\nabla_{\mathbf{a}} \left\{ \frac{1}{2} \mathbf{a}^T \mathbf{a} - \sum_{j=1}^N \lambda_j [g_j(\mathbf{a}^T \mathbf{x}_j + b) - 1] \right\} = 0 \text{ и } \frac{\partial}{\partial b} \left\{ \frac{1}{2} \mathbf{a}^T \mathbf{a} - \sum_{j=1}^N \lambda_j [g_j(\mathbf{a}^T \mathbf{x}_j + b) - 1] \right\} = 0,$$

откуда получим

$$\mathbf{a} = \sum_{j=1}^N \lambda_j g_j \mathbf{x}_j,$$

$$\sum_{j=1}^N \lambda_j g_j = 0.$$

Подстановка (2.2.13) во второе условие (2.2.11) превращает его в целевую функцию, вообще говоря, относительно множителей Лагранжа $\lambda_1, \dots, \lambda_N$ и порога b

$$W(b, \lambda_1, \dots, \lambda_N) = \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k - \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k - \left(\sum_{j=1}^N \lambda_j g_j \right) b + \sum_{j=1}^N \lambda_j,$$

однако в силу равенства (2.2.14) активными аргументами являются только множители Лагранжа. Таким образом, мы приходим к следующей формулировке задачи построения оптимальной разделяющей гиперплоскости в виде задачи квадратичного программирования, называемой в литературе двойственной по Вульффу [R. Fletcher. Practical Methods of Optimizations. John Wiley and Sons Inc., 2nd edition, 1987.] или по Лагранжу [Базара М., Шетти К. Нелинейное программирование. Теория и алгоритмы. М.: Мир, 1982] по отношению к задаче (2.2.5):

$$\begin{aligned} W(\lambda_1, \dots, \lambda_N) &= \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k \rightarrow \max, \\ \sum_{j=1}^N \lambda_j g_j &= 0, \quad \lambda_j \geq 0, \quad j = 1, \dots, N. \end{aligned} \quad (2.2.15)$$

$W(\lambda_1, \dots, \lambda_N)$ - двойственная функция Лагранжа - является вогнутой, следовательно, всякий ее локальный максимум является и глобальным [Базара М., Шетти К. Нелинейное программирование. Теория и алгоритмы. М.: Мир, 1982].

После того, как множители Лагранжа найдены, направляющий вектор оптимальной разделяющей гиперплоскости определяется по формуле (2.2.13) как линейная комбинация векторов обучающей совокупности. Те векторы, для которых $\lambda_j \neq 0$, т.е. $\lambda_j > 0$ с учетом ограничения (2.2.12), называются опорными векторами оптимальной разделяющей гиперплоскости.

Для определения значения порога b используем тот факт, что в оптимальной точке из совокупности ограничений $g_j(\mathbf{a}^T \mathbf{x}_j + b) \geq 1$ (2.2.5) активными, т.е. превращающимися в равенства $g_j(\mathbf{a}^T \mathbf{x}_j + b) = 1$, являются те, для которых множители Лагранжа положительны $\lambda_j > 0$. Для этих ограничений имеем

$$(\lambda_j g_j) g_j(\mathbf{a}^T \mathbf{x}_j + b) = \lambda_j(\mathbf{a}^T \mathbf{x}_j + b) = \lambda_j g_j$$

Но эти же равенства выполняются и для всех остальных j в силу того, что для них $\lambda_j = 0$. Сложив все эти равенства, получим

$$\sum_{j=1}^N \lambda_j(\mathbf{a}^T \mathbf{x}_j + b) = \sum_{j=1}^N \lambda_j g_j = 0, \text{ т.е. } \sum_{j=1}^N \lambda_j \mathbf{a}^T \mathbf{x}_j + \left(\sum_{j=1}^N \lambda_j \right) b = 0, \text{ откуда следует}$$

$$b = -\frac{\sum_{j=1}^N \lambda_j \mathbf{a}^T \mathbf{x}_j}{\sum_{j=1}^N \lambda_j}.$$

Заметим, что мы по-прежнему решаем задачу (2.2.5), и значение $1/\xi^2 = \mathbf{a}^T \mathbf{a}$ для направляющего вектора (2.2.12), вычисленного при оптимальных значениях весов, дает максимальную величину ξ остаточного зазора, с которым гиперплоскость разделяет точки первого и второго классов. С учетом (2.2.13) эта величина выражается непосредственно через значения весов

$$1/\xi^2 = \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k.$$

К тому же, аналогично (16), $\lambda_j g_j (\mathbf{a}^T \mathbf{x}_j + b) = \lambda_j$, что при суммировании по всем j дает

$$\sum_{j=1}^N \lambda_j = \sum_{j=1}^N \lambda_j g_j (\mathbf{a}^T \mathbf{x}_j + b) = \sum_{j=1}^N \lambda_j g_j \mathbf{a}^T \mathbf{x}_j + \left(\sum_{j=1}^N \lambda_j g_j \right) b,$$

откуда с учетом (13) и ограничения в виде равенства в (2.2.15) получим

$$\sum_{j=1}^N \lambda_j = \sum_{j=1}^N \lambda_j g_j \left(\sum_{k=1}^N \lambda_k g_k \mathbf{x}_k \right)^T \mathbf{x}_j = \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k = 1/\xi^2.$$

Таким образом, $\max_{\lambda_1, \dots, \lambda_N} W(\lambda_1, \dots, \lambda_N) = 1/2\xi^2$, и при любых других значениях весов выполняется неравенство $W(\lambda_1, \dots, \lambda_N) \leq 1/2\xi^2$.

Пусть $\varepsilon > 0$ – некоторое достаточно малое число, такое, что множества точек первого и второго класса можно считать линейно неразделимыми, если никакая гиперплоскость не может обеспечить величину остаточного зазора, большую ε . Если в процессе решения задачи квадратичного программирования (2.2.15) наступит ситуация $W(\lambda_1, \dots, \lambda_N) > 1/2\varepsilon^2$, то точки первого и второго класса линейно неразделимы, и процесс должен быть остановлен.

Рассмотрим теперь третью постановку задачи для случая линейной неразделимости объектов классов 1 и -1 . Следует отметить тот факт, что для задачи (8) система ограничений образует замкнутую область в пространстве варьируемых параметров, а критерий представляет собой **максимизацию выпуклой** квадратичной целевой функции. Это приводит к наличию нескольких локальных экстремумов, и как следствие - трудности в построении эффективной процедуры направляющего вектора оптимальной разделяющей гиперплоскости. Этот факт наглядно демонстрируется на примере, приведенном на рис. 2.2.1.

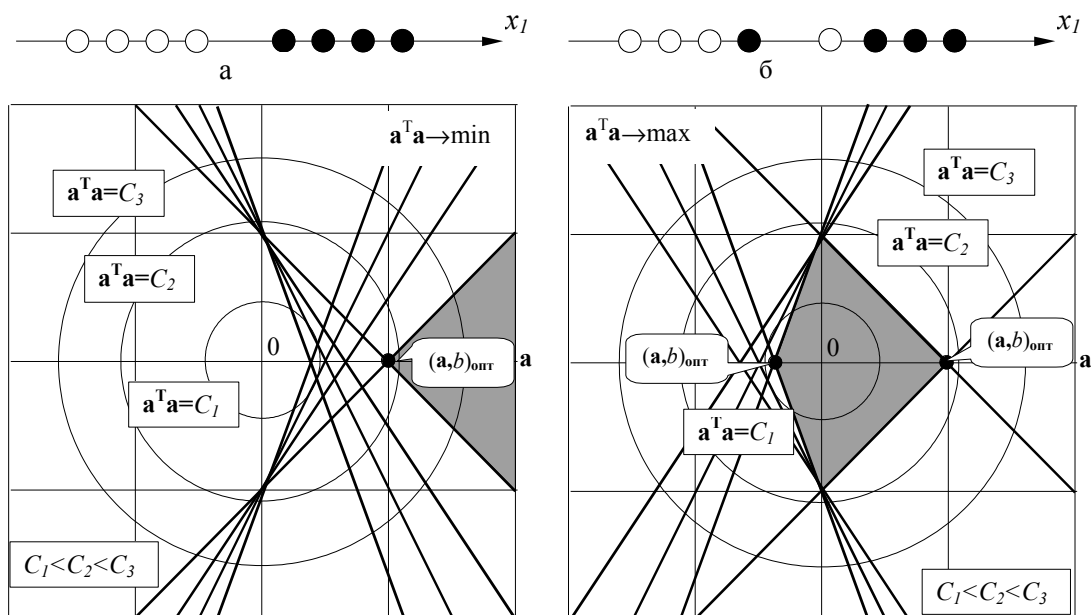
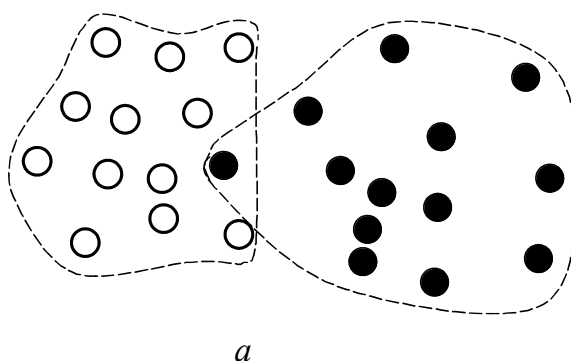


Рисунок. 2.2.1 Максимизация вогнутой (а) и выпуклой (б) функции при системе ограничений.

Это обстоятельство заставило Вапника и Кортес отказаться от записи задачи (2.2.8) в виде, эквивалентном задаче (2.2.9). В работе [C.Cortes and V.Vapnik, 1995] они предложили отличную схему, суть которой заключается в следующем.

Вернемся ко второй форме задачи построения оптимальной разделяющей гиперплоскости. Как очевидно то, что для случая линейно разделимых классов изменением масштаба оси \mathbf{a} , всегда можно сделать правые части неравенств (2.2.1) равными соответственно 1 и -1, очевидно и то, что смещением объектов 1-го класса в положительном направлении вектора \mathbf{a} и -1-го класса в отрицательном направлении удастся линейно неразделимые классы представить как разделимые. Предлагается применить такое "центробежное" преобразование по-разному для различных объектов выборки, а именно, для объектов, попавших в область чужого класса, необходимо ощутимо отодвинуть их в "свою" сторону, в то время как объекты из задних областей вообще не нуждаются в подобном смещении. Понятно преимущество такого подхода (рис.2.2.2 в) по сравнению с добавлением одинаковой константы ко всем объектам выборки (рис.2.2.2 б).



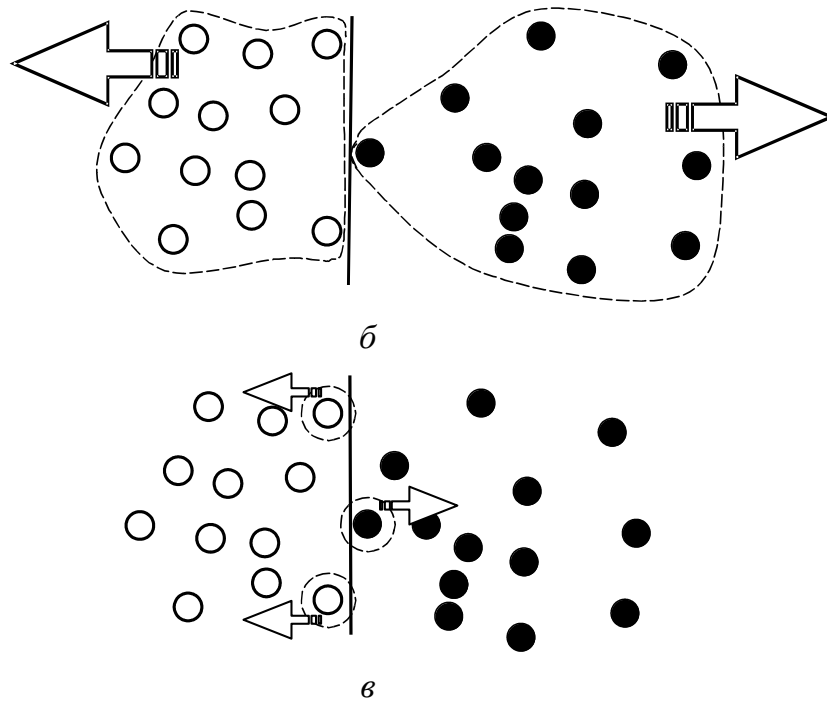


Рисунок 2.2.2 Смещение объектов для случая линейно неразделимых классов.

Он позволяет оценить число ошибок в обучающей выборке $R = \frac{n_1 + n_{-1}}{N_1 + N_{-1}}$, где n_1 и n_{-1} - минимальное число объектов в классе 1 и -1, соответственно, сместив которые, удастся добиться линейной разделимости классов.

Таким образом, для неразделимого случая перепишем неравенства (2.2.3) в виде

$$\mathbf{a}^T \mathbf{x}_j + b \geq +1 - \delta_j \text{ при } g_j = 1 \text{ и } \mathbf{a}^T \mathbf{x}_j + b \leq -1 + \delta_j \text{ при } g_j = -1,$$

где $\delta_j \geq 0$, $j = 1, \dots, N$ - неотрицательные константы, на которые необходимо сместить объекты обучающей выборки, чтобы добиться линейной разделимости. В более компактной форме это может быть записано следующим образом

$$g_j(\mathbf{a}^T \mathbf{x}_j + b) \geq +1 - \delta_j.$$

Тогда ошибка распознавания на обучающей выборке может быть представлена в следующем виде

$$R = \frac{\sum_{j=1}^N \theta(\delta_j)}{N}, \text{ где } \theta(\delta) = \begin{cases} 1, & \text{если } \delta > 0 \\ 0, & \text{если } \delta = 0 \end{cases}.$$

Задача заключается в выборе δ_j , $j = 1, \dots, N$ таким образом, чтобы $\sum_{j=1}^N \delta_j \rightarrow \min$. В

таком случае общий критерий поиска можно представить в одном из следующих видов:

$$\frac{1}{2}(\mathbf{a}^T \mathbf{a} + C \sum_{j=1}^N \delta_j) \rightarrow \min,$$

$$\frac{1}{2}[\mathbf{a}^T \mathbf{a} + C \sum_{j=1}^N (\delta_j)^2] \rightarrow \min$$

$$\frac{1}{2}[\mathbf{a}^T \mathbf{a} + C(\sum_{j=1}^N \delta_j)^k] \rightarrow \min, \quad k > 1,$$

где C - положительная константа - параметр пользователя, при ограничениях

$$\begin{aligned} g_j(\mathbf{a}^T \mathbf{x}_j + b) &\geq +1 - \delta_j, \\ \delta_j &\geq 0, \quad j = 1, \dots, N. \end{aligned}$$

Таким образом, получена задача минимизации квадратичной функции при линейных ограничениях типа неравенств. Осталось получить двойственную по отношению к ней.

Функция Лагранжа, соответствующая такой задаче имеет вид

$$\begin{aligned} L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N) &= \frac{1}{2} \mathbf{a}^T \mathbf{a} + \frac{C}{2} \sum_{j=1}^N \delta_j - \\ &- \sum_{j=1}^N \lambda_j [g_j(\mathbf{a}^T \mathbf{x}_j + b) - 1 + \delta_j] - \sum_{j=1}^N \mu_j \delta_j, \end{aligned} \quad (2.2.20)$$

где $\lambda_j \geq 0, \mu_j \geq 0, j = 1, \dots, N$ - неотрицательные множители Лагранжа.

Решением задачи является седловая точка функции Лагранжа:

$$L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N) \rightarrow \min \text{ по } \mathbf{a}, b, \delta_1, \dots, \delta_N, \quad (2.2.21)$$

$$L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N) \rightarrow \max \text{ по } \lambda_1, \dots, \lambda_N, \mu_1, \dots, \mu_N, \quad (2.2.22)$$

при ограничениях

$$\lambda_j \geq 0, \mu_j \geq 0, \quad j = 1, \dots, N.$$

Первое из этих условий дает

$$\begin{aligned} \nabla_{\mathbf{a}} L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N) &= 0, \\ \frac{\partial L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N)}{\partial b} &= 0, \\ \frac{\partial L(\mathbf{a}, b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N)}{\partial \delta_j} &= 0, \quad j = 1, \dots, N. \end{aligned}$$

откуда получим

$$\mathbf{a} = \sum_{j=1}^N \lambda_j g_j \mathbf{x}_j, \quad j = 1, \dots, N,$$

$$\sum_{j=1}^N \lambda_j g_j = 0, \quad j = 1, \dots, N,$$

$$\lambda_j + \mu_j = \frac{C}{2}, \quad j = 1, \dots, N.$$

Отметим что, так как $\lambda_j \geq 0, \mu_j \geq 0$ и $\lambda_j + \mu_j = \frac{C}{2}$, то $0 \leq \lambda_j \leq \frac{C}{2}$ и $0 \leq \mu_j \leq \frac{C}{2}$.

Подстановка (2.2.23) в условие (2.2.22) превращает его в целевую функцию

$$W(b, \lambda_1, \dots, \lambda_N, \delta_1, \dots, \delta_N, \mu_1, \dots, \mu_N) = \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k + \frac{C}{2} \sum_{j=1}^N \delta_j - \\ - \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k - \left(\sum_{j=1}^N \lambda_j g_j \right) b + \sum_{j=1}^N \lambda_j - \sum_{j=1}^N \lambda_j \delta_j - \sum_{j=1}^N \mu_j \delta_j$$

однако в силу равенств (2.2.24) и (2.2.25) активными аргументами являются только множители Лагранжа $\lambda_j \geq 0$, $j = 1, \dots, N$.

Таким образом, мы приходим к следующей формулировке задачи построения оптимальной разделяющей гиперплоскости в виде задачи квадратичного программирования

$$W(\lambda_1, \dots, \lambda_N) = \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k \rightarrow \max, \\ \sum_{j=1}^N \lambda_j g_j = 0, \quad 0 \leq \lambda_j \leq \frac{1}{2} C, \quad j = 1, \dots, N. \quad (2.2.26)$$

По-прежнему значения $\lambda_j > 0$ указывают опорные элементы обучающей выборки, определяющие параметры оптимальной разделяющей гиперплоскости, вычисляемые в данном случае по формуле (2.2.26). Максимальное значение критерия

(26) $\max_{\lambda_1, \dots, \lambda_N} W(\lambda_1, \dots, \lambda_N) = -1/2\xi^2$ укажет минимальную величину ξ дефицита, с которым точки первого и второго классов могут быть разделены гиперплоскостью, а

направляющий вектор разделяющей гиперплоскости определяется как $\mathbf{a} = \sum_{j=1}^N \lambda_j g_j \mathbf{x}_j$,

то есть так же как и для случая линейно разделимых выборок. Константа же b в этом случае будет другой. Определим ее.

Смещения δ_j равны нулю тогда и только тогда, когда соответствующие им суть множителей Лагранжа $\mu_j > 0$, то есть, когда $\lambda_j < \frac{1}{2} C$, $j = 1, \dots, N$. В других терминах:

$\delta_j > 0$ тогда и только тогда, когда $\lambda_j = \frac{1}{2} C$.

Для положительных λ_j выполняется условие $g_j(\mathbf{a}^T \mathbf{x}_j + b) = 1 - \delta_j$. Если $0 < \lambda_j < \frac{1}{2} C$, то $\delta_j = 0$, и, следовательно, $g_j(\mathbf{a}^T \mathbf{x}_j + b) = 1$. Учитывая, тот факт что $g_j^2 = 1$, перепишем последнее уравнение в виде $\lambda_j(\mathbf{a}^T \mathbf{x}_j + b) = \lambda_j g_j$, откуда следует

$\sum_{j: 0 < \lambda_j < \frac{1}{2} C} \lambda_j(\mathbf{a}^T \mathbf{x}_j + b) = \sum_{j: 0 < \lambda_j < \frac{1}{2} C} \lambda_j g_j$. Выразив отсюда b получим:

$$b = - \frac{\sum_{j: 0 \leq \lambda_j < \frac{1}{2}C} \lambda_j \mathbf{a}^T \mathbf{x}_j + \frac{1}{2}C \sum_{j: \lambda_j = \frac{1}{2}C} g_j}{\sum_{j: 0 \leq \lambda_j < \frac{1}{2}C} \lambda_j}.$$

Для задачи (2.2.19b) аналогичным образом можно получить соответствующую ей двойственную задачу вида

$$W(\lambda_1, \dots, \lambda_N) = \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k - \frac{1}{2C} \sum_{j=1}^N \lambda_j^2 \rightarrow \max, \quad (2.2.27a)$$

$$\sum_{j=1}^N \lambda_j g_j = 0, \quad \lambda_j > 0, \quad j = 1, \dots, N.$$

Или введя новые обозначения $\mathbf{D} = \{(g_j g_k \mathbf{x}_j^T \mathbf{x}_k)\}$, $\mathbf{1} = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ и $\Lambda = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_N \end{pmatrix}$ в виде

$$W(\Lambda) = \mathbf{1}^T \Lambda - \frac{1}{2} \Lambda^T (\mathbf{D} + \frac{1}{C} \mathbf{I}) \Lambda \rightarrow \max,$$

$$\sum_{j=1}^N \lambda_j g_j = 0, \quad \lambda_j > 0, \quad j = 1, \dots, N.$$

Направляющий вектор разделяющей гиперплоскости будет определяться следующим образом.

$$\mathbf{a} = \sum_{j=1}^N \lambda_j g_j \mathbf{x}_j, \quad b = - \frac{\sum_{j=1}^N \lambda_j \mathbf{a}^T \mathbf{x}_j + \frac{1}{C} \sum_{j=1}^N \lambda_j^2 g_j}{\sum_{j=1}^N \lambda_j}.$$

Для задачи выпуклого программирования (19с) получим двойственную задачу

$$W(\lambda_1, \dots, \lambda_N, d) = \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k - \frac{d^{k/k-1}}{(kC)^{k/k-1}} \left(1 - \frac{1}{k}\right) \rightarrow \max, \quad (2.2.28)$$

$$\sum_{j=1}^N \lambda_j g_j = 0, \quad 0 \leq \lambda_j \leq d, \quad j = 1, \dots, N, \quad k > 1.$$

где произведена замена $\sum_{j=1}^N \delta_j = \left(\frac{2d}{Ck}\right)^{\frac{1}{k-1}}$. В частности при $k = 2$

$$W(\lambda_1, \dots, \lambda_N, d) = \sum_{j=1}^N \lambda_j - \frac{1}{2} \left[\sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k + \frac{d^2}{C} \right] \rightarrow \max, \quad (2.2.28a)$$

$$\sum_{j=1}^N \lambda_j g_j = 0, \quad 0 \leq \lambda_j \leq d, \quad j = 1, \dots, N.$$

Направляющий вектор разделяющей гиперплоскости, определяемый таким набором опорных векторов, будет иметь вид.

$$\mathbf{a} = \sum_{j=1}^N \lambda_j g_j \mathbf{x}_j, b = - \frac{\sum_{j: 0 \leq \lambda_j \leq d} \lambda_j a^T \mathbf{x}_j + d \sum_{j: \lambda_j = d} g_j}{\sum_{j: 0 \leq \lambda_j < d} \lambda_j}.$$

2.3 Метод опорных векторов и алгоритм обучения распознаванию для случая многих классов

Пусть обучающая совокупность содержит N объектов k классов, представленных векторами их действительных признаков $\mathbf{x}_j \in \mathbf{R}^n$ и индексами классов $g_j \in \{1, \dots, k\}$, $j = 1, \dots, N$.

Для случая двух классов решение задачи распознавания образов хорошо известно []. Главная идея обучения состоит в построении разделяющей гиперплоскости таким образом, чтобы максимизировать расстояние между гиперплоскостью и крайними точками обучающих подвыборок. Это дает следующую задачу оптимизации:

$$\phi(\mathbf{a}, \delta) = \frac{1}{2} \left(\mathbf{a}^T \mathbf{a} + C \sum_{j=1}^N \delta_j \right) \rightarrow \min, \quad (2.4.1)$$

при ограничениях

$$\begin{aligned} g_j (\mathbf{a}^T \mathbf{x}_j + b) &\geq +1 - \delta_j, \\ \delta_j &\geq 0, \quad j = 1, \dots, N, \\ g_j &\in \{-1, 1\}. \end{aligned} \quad (2.4.2)$$

Двойственная к ней задача, имеющая вид

$$\begin{aligned} W(\lambda_1, \dots, \lambda_N) &= \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N (g_j g_k \mathbf{x}_j^T \mathbf{x}_k) \lambda_j \lambda_k \rightarrow \max, \\ \sum_{j=1}^N \lambda_j g_j &= 0, \quad 0 \leq \lambda_j \leq \frac{1}{2} C, \quad j = 1, \dots, N. \end{aligned} \quad (2.4.3)$$

дает следующее решающее правило

$$f(x) = \text{sign} \left[\sum_{j=1}^N \lambda_j g_j \mathbf{x}_j + b \right].$$

Обучение распознаванию образов в случае многих классов может производиться несколькими способами. Во-первых, для каждой пары классов может строиться своя разделяющая гиперплоскость. Т.о. общее количество гиперплоскостей будет $\frac{k(k-1)}{2}$. Во-вторых, можно построить k гиперплоскостей, каждая из которых отделяет класс m от $k-1$ других классов. В обоих этих случаях общий алгоритм обучения строится как совокупность задач обучения для двух классов.

Более естественным кажется решать задачу обучения для многих классов, рассматривая все k классов одновременно. Отвлечемся от понятия оптимальной разделяющей гиперплоскости. Для каждого класса будем строить свою гиперплоскость, минимизирующую функционал, аналогичный (2.4.1):

$$\phi(\mathbf{a}, \delta) = \frac{1}{2} \left(\sum_{m=1}^k \mathbf{a}_m^T \mathbf{a}_m + C \sum_{j=1}^N \sum_{m \neq g_j} \delta_j^m \right) \quad (2.4.4)$$

при ограничениях

$$\mathbf{a}_{g_j}^T \mathbf{x}_j + b_{g_j} \geq \mathbf{a}_m^T \mathbf{x}_j + b_m + 2 - \delta_j^m, \quad (2.4.5)$$

$$\delta_j^m \geq 0, j = 1, \dots, N, m \in \{1, \dots, k\} \setminus g_j$$

Расстояния до этой гиперплоскости от всех точек родного класса больше, чем от всех остальных точек обучающей совокупности. Еще раз отметим, что это есть не разделяющая, а неким хитрым образом организованная гиперплоскость, хотя для случая двух классов она совпадает с оптимальной разделяющей гиперплоскостью.

Эта задача дает решающее правило

$$f(\mathbf{x}) = \arg \max_m [\mathbf{a}_m^T \mathbf{x} + b_m], m = 1, \dots, k \quad (2.4.6)$$

Таким образом, получена задача минимизации квадратичной функции при ограничениях типа неравенств. Осталось получить двойственную к ней.

Функция Лагранжа, соответствующая этой задаче имеет вид

$$L(\mathbf{a}, b, \delta, \lambda, \mu) = \frac{1}{2} \sum_{m=1}^k \mathbf{a}_m^T \mathbf{a}_m + \frac{C}{2} \sum_{j=1}^N \sum_{m=1}^k \delta_j^m - \sum_{j=1}^N \sum_{m=1}^k \lambda_j^m [\mathbf{a}_{g_j}^T \mathbf{x}_j + b_{g_j} - \mathbf{a}_m^T \mathbf{x}_j - b_m - 2 + \delta_j^m] - \sum_{j=1}^N \sum_{m=1}^k \mu_j^m \delta_j^m \quad (2.4.7)$$

с фиктивными переменными

$$\lambda_j^{g_j} = 0, \delta_j^{g_j} = 2, \mu_j^{g_j} = 0, j = 1, \dots, N$$

где $\lambda_j^m \geq 0, \mu_j^m \geq 0, j = 1, \dots, N, m = 1, \dots, k$ - неотрицательные множители Лагранжа.

Решением задачи является седловая точка функции Лагранжа:

$$L(\mathbf{a}, b, \delta, \lambda, \mu) \rightarrow \min \text{ по } \mathbf{a}, b, \delta$$

$$L(\mathbf{a}, b, \delta, \lambda, \mu) \rightarrow \max \text{ по } \lambda, \mu$$

Введем следующие обозначения

$$c_j^n = \begin{cases} 1, & \text{если } g_j = n \\ 0, & \text{если } g_j \neq n \end{cases}$$

и

$$\Lambda_j = \sum_{m=1}^k \lambda_j^m$$

Дифференцируя (7) по \mathbf{a}_n , b_n и δ_j^n , получим

$$\frac{\partial L}{\partial \mathbf{a}_n} = \mathbf{a}_n + \sum_{j=1}^N \lambda_j^n \mathbf{x}_j - \sum_{j=1}^N \Lambda_j c_j^n \mathbf{x}_j$$

$$\frac{\partial L}{\partial b_n} = \sum_{j=1}^N \lambda_j^n - \sum_{j=1}^N \Lambda_j c_j^n$$

$$\frac{\partial L}{\partial \delta_j^n} = -\lambda_j^n + \frac{C}{2} - \mu_j^n$$

Седловая точка функции Лагранжа удовлетворяет следующим условиям

$$\frac{\partial L}{\partial \mathbf{a}_n} = 0 \Rightarrow \mathbf{a}_n = \sum_{j=1}^N (\lambda_j^n - \Lambda_j c_j^n) \mathbf{x}_j \quad (2.4.8)$$

$$\frac{\partial L}{\partial b_n} = 0 \Rightarrow \sum_{j=1}^N \lambda_j^n = \sum_{j=1}^N \Lambda_j c_j^n \quad (2.4.9)$$

$$\frac{\partial L}{\partial \delta_j^n} = 0 \Rightarrow \lambda_j^n + \mu_j^n = \frac{C}{2} \text{ или } 0 \leq \lambda_j^n \leq \frac{C}{2}. \quad (2.4.10)$$

Подставляя (2.4.8) в (2.4.7), получим

$$\begin{aligned} W(b, \delta, \lambda, \mu) = & \frac{1}{2} \sum_{m=1}^k \sum_{i=1}^N \sum_{j=1}^N (c_i^m \Lambda_i - \lambda_i^m) (c_j^m \Lambda_j - \lambda_j^m) \mathbf{x}_i^T \mathbf{x}_j - \\ & - \sum_{m=1}^k \sum_{i=1}^N \lambda_i^m \left[\sum_{j=1}^N (c_j^{g_i} \Lambda_j - \lambda_j^{y_i}) \mathbf{x}_i^T \mathbf{x}_j - \sum_{j=1}^N (c_j^m \Lambda_j - \lambda_j^m) \mathbf{x}_i^T \mathbf{x}_j + b_{y_i} - b_m - 2 \right] - \\ & - \sum_{m=1}^k \sum_{j=1}^N \lambda_j^m \delta_j^m + \frac{C}{2} \sum_{j=1}^N \sum_{m=1}^k \delta_j^m - \sum_{j=1}^N \sum_{m=1}^k \mu_j^m \delta_j^m \end{aligned} \quad (2.4.11)$$

Добавив ограничение (2.4.10), исключим в (11) δ .

Заметим, что

$$B_1 = \sum_{i,m} \lambda_i^m b_{y_i} = \sum_m b_m \left(\sum_i c_i^m \Lambda_i \right),$$

$$B_2 = \sum_{i,m} \lambda_i^m b_m = \sum_m b_m \left(\sum_i \lambda_i^m \right),$$

но, принимая во внимание (2.4.9), $B_1 = B_2$, что дает

$$\begin{aligned} W(\lambda) = & 2 \sum_{i,m} \lambda_i^m + \sum_{i,j,m} \left(\frac{1}{2} c_i^m c_j^m \Lambda_i \Lambda_j - \frac{1}{2} c_i^m \Lambda_i \lambda_j^m - \frac{1}{2} c_j^m \Lambda_j \lambda_i^m + \frac{1}{2} \lambda_j^m \lambda_i^m - c_j^{g_i} \Lambda_j \lambda_i^m + \right. \\ & \left. + \lambda_j^m \lambda_i^{g_i} + c_j^m \Lambda_j \lambda_i^m - \lambda_j^m \lambda_i^m \right) \cdot \mathbf{x}_i^T \mathbf{x}_j \end{aligned}$$

но $\sum_{i,j,m} c_i^m \Lambda_i \lambda_j^m = \sum_{i,j,m} c_j^m \Lambda_j \lambda_i^m$, поэтому

$$W(\lambda) = 2 \sum_{i,m} \lambda_i^m + \sum_{i,j,m} \left(\frac{1}{2} c_i^m c_j^m \Lambda_i \Lambda_j - c_j^{g_i} \Lambda_i \lambda_i^m - \frac{1}{2} \lambda_j^m \lambda_i^m + \lambda_j^m \lambda_i^{g_i} \right) \cdot \mathbf{x}_i^T \mathbf{x}_j.$$

Т.к. $\sum_m c_i^m c_j^m = c_i^{g_i} = c_j^{g_i}$, окончательно получим двойственную задачу

$$W(\lambda) = 2 \sum_{i,m} \lambda_i^m + \sum_{i,j,m} \left(-\frac{1}{2} c_i^{g_i} \Lambda_i \Lambda_j - \frac{1}{2} \lambda_j^m \lambda_i^m + \lambda_j^m \lambda_i^{g_i} \right) \cdot \mathbf{x}_i^T \mathbf{x}_j \rightarrow \max, \quad (2.4.11a)$$

которая является квадратичной функцией в терминах множителей Лагранжа с линейными ограничениями

$$\sum_{j=1}^N \lambda_j^n = \sum_{j=1}^N c_j^n \Lambda_j, \quad n = 1, \dots, k$$

и

$$0 \leq \lambda_j^m \leq \frac{C}{2}, \quad \lambda_j^{g_j} = 0, \quad j = 1, \dots, N, \quad m \in \{1, \dots, k\} \setminus g_j$$

Решив эту задачу, получим решающее правило

$$f(\mathbf{x}, \lambda) = \arg \max_m \left[\sum_{j=1}^N (c_j^m \Lambda_j - \lambda_i^m) \mathbf{x}_j^T \mathbf{x} + b_m \right].$$

По-прежнему значения $\lambda_j > 0$ указывают опорные элементы обучающей выборки, определяющие параметры оптимальной разделяющей гиперплоскости, вычисляемые в данном случае по формуле

3. Учет априорных предпочтений о классе решающих правил.

3.1. Обучение распознаванию сигналов с учетом критерия гладкости решающего правила

Гиперплоскость, оптимальная в смысле наилучшего разделения точек первого и второго класса в обучающей выборке по критерию В.Н. Вапника показана на рис. 3.1.1; выделены т.н. опорные точки, только на которые фактически и опирается оптимальная гиперплоскость

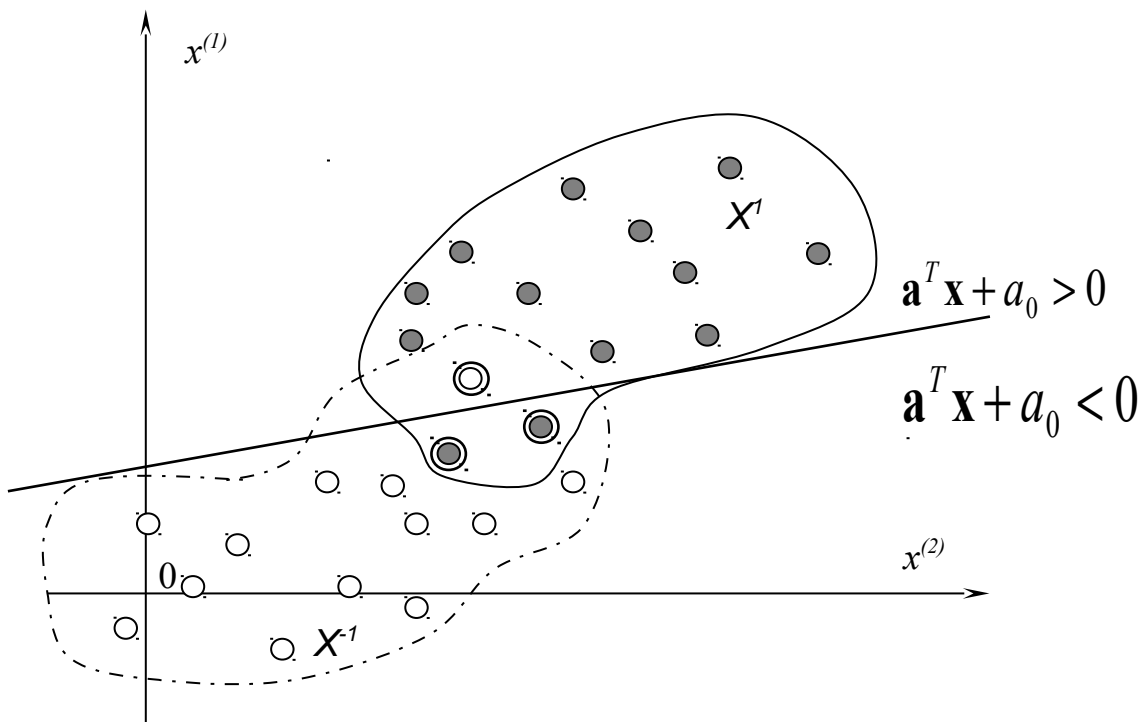


Рисунок 3.1.1 Гиперплоскость, оптимальная в смысле наилучшего разделения точек первого и второго класса в обучающей выборке по критерию В.Н. Вапника.

Можно показать, что каковы бы ни были обучающие подвыборки первого и второго класса, из них всегда можно удалить часть объектов так, что оптимальное решение $\hat{\mathbf{a}}$ для оставшихся будет в точности таким же, как и для выборки в целом. Минимальное число объектов, которое надо оставить, чтобы получающаяся гиперплоскость не изменилась, зависит от конкретной конфигурации подвыборок, но оно всегда не меньше двух, по одному объекту первого и второго класса, и не больше $n+1$, т.е. на единицу больше исходной размерности n пространства признаков. Именно эти объекты и определяют оптимальную разделяющую гиперплоскость, она как бы “опирается” на них. Такие объекты называют опорными. Это самые крайние точки подвыборок с тех сторон, которыми они обращены друг к другу в пространстве \mathbf{R}^n . При малых

размерах выборки N обучающая совокупность, показывая, где в основном сосредоточены объекты разных классов, будет содержать весьма скудную информацию о форме границ областей. Поэтому доверять этой информации надо с осторожностью.

Кроме того, если число объектов N в обучающей совокупности недостаточно велико и сравнимо с размерностью пространства признаков n , то при любой комбинации объектов и их классов удастся найти решающее правило, которое правильно классифицирует объекты данной выборки. Но в результате может оказаться, что оптимальное для данной выборки решающее правило будет очень плохо узнавать классы объектов в других выборках. Иначе говоря, при малых размерах обучающей выборки свобода выбора параметров решающего правила, в данном случае, оказывается слишком большой, и для устойчивости обучения ее надо как-то ограничить [17,18].

Предлагается новый подход к регуляризации решающего правила распознавания многомерных объектов, когда совокупность описывающих их признаков представляет собой результат упорядоченных вдоль оси некоторого аргумента измерений одной и той же характеристики. Типичным примером таких объектов являются некоторые виды сигналов, для которых естественно предположение о невозможности произвольно резких скачков значений соседних отсчетов. Требование гладкости, сужая область допустимых значений решающего правила, позволяет улучшить качество распознавания на генеральной совокупности, пусть даже ценой некоторого снижения его качества на обучающей выборке. Эффективность требования гладкости решающего правила для малонаполненных выборок подтверждена экспериментами.

Предположим, что отдельные признаки $(x_i, i = 1, \dots, n)$ в составе вектора \mathbf{X} представляют собой результат упорядоченного измерения некоторого свойства объекта вдоль координаты той или иной природы, причем есть основания полагать, что соседние признаки несут почти идентичную информацию о принадлежности объекта к определенному классу. Такое предположение эквивалентно принятию тезиса о существовании априорной информации о значениях коэффициентов $(a_i, i = 1, \dots, n)$ в составе вектора параметров \mathbf{a} , заключающееся в том, что соседние коэффициенты, скорее всего, не слишком сильно отличаются друг от друга, т.е. плавно изменяются при увеличении индекса i .

Для того, чтобы в процессе обучения предпочтение отдавалось решающим правилам с плавным изменением коэффициентов линейной части, можно, например, внести в критерий дополнительную аддитивную составляющую

$$J'(\mathbf{a}) = \sum_{i=2}^n (a_i - a_{i-1})^2. \quad (2.4.1.)$$

Нетрудно убедиться, что такая квадратичная функция может быть записана в виде

$$J'(\mathbf{a}) = \mathbf{a}^T \tilde{\mathbf{B}} \mathbf{a},$$

$$\tilde{\mathbf{B}}(n \times n) = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 2 & -1 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 2 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 2 & -1 & 0 \\ 0 & 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{pmatrix} \quad (2.4.2)$$

Тогда целевая функция будет включать в себя еще одно слагаемое

$$\frac{1}{2}(\mathbf{a}^T \mathbf{a} + \alpha \mathbf{a}^T \tilde{\mathbf{B}} \mathbf{a} + C \sum_{j=1}^N \delta_j) \rightarrow \min$$

или более компактно

$$\frac{1}{2} \mathbf{a}^T \mathbf{B} \mathbf{a} + C \sum_{j=1}^N \delta_j \rightarrow \min, \text{ где } \mathbf{B} = \mathbf{I} + \alpha \tilde{\mathbf{B}}, \quad (2.4.3)$$

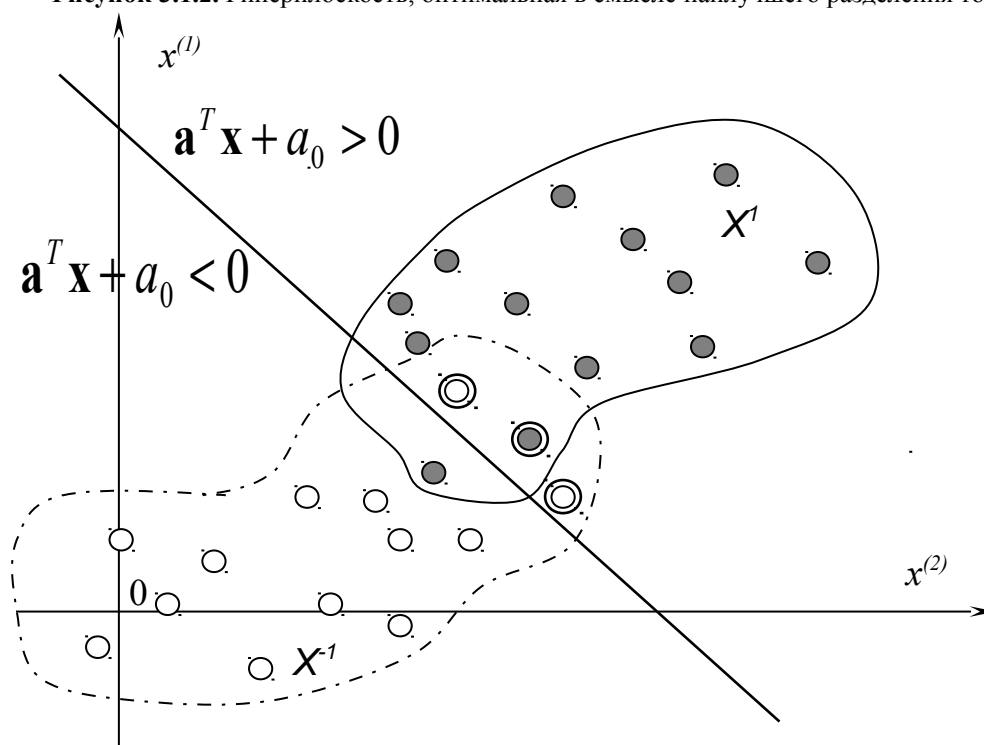
при ограничениях

$$\begin{aligned} g_j(\mathbf{a}^T \mathbf{x}_j + b) &\geq +1 - \delta_j, \\ \delta_j &\geq 0, \quad j = 1, \dots, N. \end{aligned}$$

Здесь коэффициент $\alpha \geq 0$ определяет степень, с которой штраф на негладкость последовательности коэффициентов участвует в процессе обучения. Очевидно, что матрица \mathbf{B} положительно определена [30].

Гиперплоскость, оптимальная в смысле наилучшего разделения точек первого и второго класса в обучающей выборке по критерию В.Н. Вапника с учетом критерия гладкости решающего правила, показана на рисунок. 3.1.1.

Рисунок 3.1.2. Гиперплоскость, оптимальная в смысле наилучшего разделения точек первого и



второго класса в обучающей выборке по критерию В.Н. Вапника с учетом критерия гладкости решающего правила

3.2. Оптимальное линейное решающее правило в метрическом пространстве.

Метод опорных векторов, как следует из его названия, разработан для линейного пространства векторов признаков. Несмотря на это существует широкий класс приложений, в которых трудно или невозможно выбрать фиксированное множество признаков переменных, формирующих линейное пространство, в котором задача распознавания образов может быть решена как задача нахождения разделяющей гиперплоскости. Часто случается, что степень различия может быть измерена только для двух объектов, или, другими словами, может быть сформировано только метрическое пространство признаков. Эта ситуация и обсуждается в этой главе.

Пусть Ω , как и ранее, представляет собой гипотетическое множество объектов распознавания, элемент которого будем обозначать $\omega \in \Omega$. Определим на множестве Ω неотрицательную функцию $r(\omega', \omega'') \geq 0$, которая обладает свойствами метрики, т.е. $r(\omega, \omega) = 0$, $r(\omega', \omega'') = r(\omega'', \omega')$, $r(\omega', \omega''') = r(\omega'', \omega') + r(\omega'', \omega''')$. Как и ранее индикаторная функция $g(\omega)$, определенная на Ω , принимающая значения из двуэлементного множества $\{1, -1\}$ и определяющее конкретное действительное разбиение множества всех объектов на два класса.

Пусть $((\omega_j, g_j), j=1, \dots, N)$ – обучающая выборка, в которой «учитель» указал класс $g_j = g(\omega_j)$ каждого объекта. Если появляется новый объект $\omega \in \Omega$, то принять решение о его классе $\hat{g}(\omega)$ можно лишь на основании измерения расстояний этого объекта до всех объектов обучающей выборки, классы которых известны. В этой ситуации задача обучения распознаванию образов сводится к формированию решающего правила вида $\hat{g}(\omega) = \hat{g}(r(\omega, \omega_1), \dots, r(\omega, \omega_N))$ на основании гипотезы компактности, согласно которой два объекта, характеризующиеся малым расстоянием между ними $r(\omega', \omega'')$ скорее всего принадлежат к одному и тому же классу.

Например, естественно ввести в рассмотрение линейную дискриминантную функцию

$$d(\omega | a_1, \dots, a_N) = \sum_{k: g(\omega_k)=-1} a_k r(\omega, \omega_k) - \sum_{l: g(\omega_l)=1} a_l r(\omega, \omega_l) = \sum_{j=1}^N (-g_j r(\omega, \omega_j)) a_j, \quad (3.2.1)$$

построенную как разность средневзвешенных расстояний вновь поступившего объекта до объектов второго и первого класса, где $a_j \geq 0$ – некоторые весовые коэффициенты, и принять решающее правило распознавания в виде

$$\hat{g}(\omega | a_1, \dots, a_N) = \begin{cases} 1, & d(\omega | a_1, \dots, a_N) \geq 0, \\ -1, & d(\omega | a_1, \dots, a_N) < 0. \end{cases} \quad (3.2.2)$$

Такая дискриминантная функция и, соответственно, решающее правило, полностью характеризуются значениями неотрицательных весовых коэффициентов при объектах обучающей выборки $a_j \geq 0$, $j = 1, \dots, N$, играющих роль параметров.

Будем рассматривать совокупность расстояний произвольного объекта ω до всех элементов обучающей выборки ω_j с учетом их классов g_j как N -мерный вектор метрических признаков $\mathbf{x} = (x_1 \dots x_N)^T \in \mathbb{R}^N$, $x_j = -g_j r(\omega, \omega_j)$, число которых совпадает с числом элементов выборки. Точно так же удобно ввести в рассмотрение N -мерный вектор весовых коэффициентов $\mathbf{a} = (a_1 \dots a_N)^T \in \mathbb{R}^N$. Тогда каждая линейная дискриминантная функция из параметрического семейства (3.2.1)

$$d(\omega | a_1, \dots, a_N) = d(\mathbf{x} | \mathbf{a}) = \mathbf{a}^T \mathbf{x} = \sum_{j=1}^N a_j x_j,$$

$$\hat{g}(\omega | a_1, \dots, a_N) = \hat{g}(\mathbf{x} | \mathbf{a}) = \begin{cases} 1, & d(\mathbf{x} | \mathbf{a}) > 0, \\ -1, & d(\mathbf{x} | \mathbf{a}) < 0, \end{cases}$$

определяет дискриминантную гиперплоскость $d(\mathbf{x} | \mathbf{a}) = \mathbf{a}^T \mathbf{x} = 0$, и соответственно решающее правило распознавания $\hat{g}(\omega | a_1, \dots, a_N) = \hat{g}(\mathbf{x} | \mathbf{a})$, в пространстве метрических признаков объекта относительно фиксированной совокупности объектов аналогично классическому случаю, когда признаки представляли собой результаты измерений произвольных свойств объекта. Специфика линейного пространства метрических признаков, образованного самой обучающей выборкой, заключается в том, что компоненты вектора параметров гиперплоскости выбираются лишь из множества неотрицательных значений $a_j \geq 0$, $j = 1, \dots, N$.

На этапе обучения нет другой информации о связи класса объекта с его расстояниями до других объектов, кроме обучающей выборки, поэтому, в

качестве критерия обучения следует принять правильность определения классов объектов в составе обучающей выборки $i = 1, \dots, N$

$$d(\omega_i | a) = d(x_i | a) = a^T x_i = \sum_{j=1}^N a_j x_{ij} = \sum_{j=1}^N a_j (-g_j r(\omega_i, \omega_j)) \begin{cases} > 0 \text{ if } g_i = 1, \\ < 0 \text{ if } g_i = -1, \end{cases}$$

Такой принцип обучения в пространстве метрических признаков полностью аналогичен принципу обучения в произвольном линейном пространстве с тем лишь отличием, что направляющего вектора разделяющей гиперплоскости ищется лишь среди векторов с неотрицательными компонентами и среди параметров разделяющей гиперплоскости нет константы b . Даже если предположить, что гиперплоскость, в точности удовлетворяющая критерию обучения, не существует, ничто не мешает использовать тот же прием, который был применен в задаче обучения распознаванию образов в линейном пространстве общего вида, заключающийся в поиске таких минимальных сдвигов «мешающих» точек выборки в направлении «своего» класса $\delta_i \geq 0$, которые обеспечат существование разделяющей гиперплоскости. Повторяя рассуждения, приведенные в главе 2, мы придем к общей математической постановке задачи построения оптимального решающего правила распознавания в пространстве метрических признаков, аналогичной (2.2.19) с дополнительными ограничениями на неотрицательность компонент направляющего вектора разделяющей гиперплоскости $a_1 \geq 0, \dots, a_N \geq 0$:

$$\begin{aligned} \frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j &\rightarrow \min, C > 0, \\ \sum_{k=1}^N (-g_j g_k r_{jk}) a_k &\geq 1 - \delta_j, \delta_j \geq 0, a_j \geq 0, j = 1, \dots, N. \end{aligned} \quad (3.2.3)$$

Для решения этой задачи составим функцию Лагранжа:

$$\begin{aligned} L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; \nu_1, \dots, \nu_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) = \\ \frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j - \sum_{j=1}^N \lambda_j \left[\sum_{k=1}^N (-g_j g_k r_{jk}) a_k - 1 + \delta_j \right] - \sum_{j=1}^N \mu_j \delta_j - \sum_{j=1}^N \nu_j a_j. \end{aligned} \quad (3.2.4)$$

Преобразуем функцию Лагранжа к более удобному виду

$$\begin{aligned}
L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; v_1, \dots, v_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) = \\
\frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j - \sum_{j=1}^N \lambda_j \left[\sum_{k=1}^N (-g_j g_k r_{jk}) a_k - 1 + \delta_j \right] - \sum_{j=1}^N \mu_j \delta_j - \sum_{j=1}^N v_j a_j = \\
\frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j - \sum_{j=1}^N \lambda_j \sum_{k=1}^N (-g_j g_k r_{jk}) a_k + \sum_{j=1}^N \lambda_j - \sum_{j=1}^N \lambda_j \delta_j - \sum_{j=1}^N \mu_j \delta_j - \sum_{j=1}^N v_j a_j = \\
\frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j - \sum_{k=1}^N \lambda_k \sum_{j=1}^N (-g_j g_k r_{jk}) a_j + \sum_{j=1}^N \lambda_j - \sum_{j=1}^N \lambda_j \delta_j - \sum_{j=1}^N \mu_j \delta_j - \sum_{j=1}^N v_j a_j = \\
\frac{1}{2} \sum_{j=1}^N a_j^2 + C \sum_{j=1}^N \delta_j - \sum_{j=1}^N \left(\sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) \right) a_j + \sum_{j=1}^N \lambda_j - \sum_{j=1}^N \lambda_j \delta_j - \sum_{j=1}^N \mu_j \delta_j - \sum_{j=1}^N v_j a_j = \\
\sum_{j=1}^N \left(\frac{1}{2} a_j^2 - v_j a_j - \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) a_j \right) + \sum_{j=1}^N \lambda_j - \sum_{j=1}^N (\lambda_j + \mu_j - C) \delta_j
\end{aligned}$$

Решением является седловая точка функции Лагранжа

$$\begin{aligned}
L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; v_1, \dots, v_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) \rightarrow \min_{a_j, \delta_j, j=1, \dots, N} \\
L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; v_1, \dots, v_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) \rightarrow \max_{v_j \geq 0, \mu_j \geq 0, \lambda_j \geq 0, j=1, \dots, N}
\end{aligned}$$

Первое из этих условий дает

$$\frac{\partial}{\partial a_j} L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; v_1, \dots, v_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) = a_j - v_j - \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) = 0$$

$$\boxed{a_j = v_j + \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk})} \quad (3.2.5)$$

$$\frac{\partial}{\partial \delta_j} L(a_1, \dots, a_N; \delta_1, \dots, \delta_N; v_1, \dots, v_N; \mu_1, \dots, \mu_N; \lambda_1, \dots, \lambda_N) =$$

$$-(\lambda_j + \mu_j - C) = 0, \quad j = 1, \dots, N$$

$$\boxed{\lambda_j + \mu_j = C, \quad j = 1, \dots, N} \quad (3.2.6)$$

Подстановка (3.2.5) и (3.2.6) в (3.2.4) дает целевую функцию при условно оптимальных значениях $\hat{a}_j, \hat{\delta}_j, j = 1, \dots, N$

$$\begin{aligned}
W(v_1, \dots, v_N; \lambda_1, \dots, \lambda_N) &= -\frac{1}{2} \sum_{j=1}^N \hat{a}_j^2(v_j, \lambda_1, \dots, \lambda_N) + \sum_{j=1}^N \lambda_j = \\
&= -\frac{1}{2} \sum_{j=1}^N \left(v_j + \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) \right)^2 + \sum_{j=1}^N \lambda_j = \\
&= -\frac{1}{2} \sum_{j=1}^N \left[v_j^2 + 2v_j \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) + \left(\sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) \right)^2 \right] + \sum_{j=1}^N \lambda_j = \\
&= -\frac{1}{2} \sum_{j=1}^N v_j^2 - \sum_{j=1}^N \sum_{k=1}^N (-g_j g_k r_{jk}) v_j \lambda_k - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N \left(\sum_{l=1}^N (-g_j g_l r_{jl}) (-g_k g_l r_{kl}) \right) \lambda_j \lambda_k + \sum_{j=1}^N \lambda_j
\end{aligned}$$

Т.о. мы приходим к двойственной задаче квадратичного программирования

$$W(v_1, \dots, v_N; \lambda_1, \dots, \lambda_N) = \sum_{j=1}^N \lambda_j - \frac{1}{2} \sum_{j=1}^N v_j^2 - \sum_{j=1}^N \sum_{k=1}^N (-g_j g_k r_{jk}) v_j \lambda_k - \frac{1}{2} \sum_{j=1}^N \sum_{k=1}^N \left(\sum_{l=1}^N (-g_j g_l r_{jl}) (-g_k g_l r_{kl}) \right) \lambda_j \lambda_k \rightarrow \max, \\ v_j \geq 0, 0 \leq \lambda_j \leq C$$

если $v_j > 0$, то $a_j = 0$ и $v_j = -\sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) > 0$,

если $v_j = 0$, то $a_j = \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) > 0$.

Или другой вид двойственной задачи:

$$W(a_1, \dots, a_N; \lambda_1, \dots, \lambda_N) = \frac{1}{2} \sum_{j=1}^N a_j^2 - \sum_{j=1}^N \lambda_j \rightarrow \min, \\ a_j - \sum_{k=1}^N \lambda_k (-g_j g_k r_{jk}) \geq 0, a_j \geq 0, 0 \leq \lambda_j \leq C, j = 1, \dots, N$$

По-прежнему значения $\lambda_j > 0$ указывают опорные элементы обучающей выборки

Правда, в отличие от линейного пространства общего вида обучающая выборка в пространстве метрических признаков всегда линейно разделима в классе гиперплоскостей $\mathbf{a}^T \mathbf{x} = 0$ с неотрицательными компонентами направляющего вектора в силу того, что число элементов в обучающей выборке совпадает с размерностью пространства, и что по определению метрических признаков $x_{ij} = -g_j r(\omega_i, \omega_j) = -r(\omega_i, \omega_j) \leq 0$ при определении расстояния данного объекта ω_i до объектов первого класса $g_j = 1$ и $x_{ij} = -g_j r(\omega_i, \omega_j) = r(\omega_i, \omega_j) \geq 0$ при определении расстояния до объектов второго класса $g_j = -1$. В силу этого обстоятельства при решении задачи квадратичного программирования, всегда будут получаться нулевые значения оптимальных сдвигов $\delta_i = 0$ для всех объектов $i = 1, \dots, N$.

Однако линейная разделимость обучающей выборки в пространстве метрических признаков обманчива, и в реальных задачах так строить процесс обучения нельзя. Дело в том, что решающее правило распознавания, построенное по обучающей выборке, будет хорошо работать на новых объектах лишь при условии, что число элементов выборки по крайней мере на порядок превосходит размерность пространства признаков [58]. В то же время, при обучении в пространстве метрических признаков число элементов выборки совпадает с размерностью пространства, и разделяющая гиперплоскость, оптимальная в смысле обеспечения максимального «зазора» между классами при ограничениях (3.2.3), будет слишком чувствительна к случайным аспектам пространственной формы обучающей выборки, в которых «утонет» информация

о фактической связи класса объекта распознавания с его расстояниями до объектов выборки. Для повышения статистической стабильности процесса обучения, или, как говорят, для его регуляризации, необходимо привлечение некоторой дополнительной априорной информации об ожидаемом направлении разделяющей гиперплоскости [18].

Заметим, что требование минимизации квадрата нормы направляющего вектора искомой разделяющей гиперплоскости $\mathbf{a}^T \mathbf{a} \rightarrow \min$ выражается присутствием в целевой функции задачи квадратичного программирования квадратичной формы с единичной матрицей $\mathbf{a}^T \mathbf{a} = \sum_{j=1}^N a_j^2$. Часто оказывается, что априорная информация, существенно снижающая свободу выбора разделяющей гиперплоскости, может быть формализована путем использования квадратичной функции с некоторой неединичной положительно определенной матрицей $\mathbf{a}^T \mathbf{Q} \mathbf{a}$. Мы приходим к формальной постановке задачи обучения в пространстве метрических признаков как задачи квадратичного программирования

$$\begin{aligned} \frac{1}{2} \mathbf{a}^T \mathbf{B} \mathbf{a} + C \sum_{i=1}^N \delta_i = \sum_{j=1}^N \sum_{k=1}^N \beta_{jk} a_j a_k + C \sum_{i=1}^N \delta_i \rightarrow \min, \\ g_i(\mathbf{a}^T \mathbf{x}_i) = g_i \left(\sum_{j=1}^N a_j x_{ij} \right) \geq 1 - \delta_i, \quad \delta_i \geq 0, \quad i = 1, \dots, N, \quad a_j \geq 0, \quad j = 1, \dots, N. \end{aligned} \quad (3.2.4)$$

При такой постановке задачи обучения гиперплоскость, в точности разделяющая объекты разных классов в обучающей выборке, может оказаться невыгодной с точки зрения априорных предпочтений, выражаемых матрицей \mathbf{B} . В результате оптимальная гиперплоскость пожертвует, если понадобится, правильной классификацией некоторых наиболее «диких» объектов выборки, что выразится в положительных значениях их сдвигов $\delta_i > 0$. При правильном выборе матрицы \mathbf{B} в учет имеющейся априорной информации доля таких объектов будет более лучше соответствовать фактической разрешимости задачи распознавания, и при применении решающего правила (3.2.2.) к новым объектам \mathbf{Q} будет существенно повышена надежность распознавания.

ω_j

Специфической особенностью метрических признаков $x_j = r(\omega, \omega_j)$ всякого объекта \mathbf{Q} является то обстоятельство что они образованы некоторой совокупностью реальных базовых объектов $\omega_1, \dots, \omega_N$ которые сами характеризуются взаимными расстояниями согласно той же метрики. Если расстояние $r(\omega_j, \omega_k)$ мало, т.е. они близко расположены друг к другу в метрическом пространстве то соответствующие признаки x_j и x_k не несут

существенно разной информации об объекте распознавания ω . Линейная дискриминантная функция имеет вид линейной комбинации расстояний объекта

до базовых объектов $\mathbf{a}^T \mathbf{x} = \sum_{j=1}^N a_j x_{ij} = \sum_{j=1}^N a_j (-g_j r(\omega_i, \omega_j))$, и разные значения

коэффициентов a_j и a_k имеют смысл только в том случае, если помогают учесть разную роль этих двух признаков в определении класса объекта. Матрица расстояний между базовыми признаками является, в сущности, матрицей расстояний между признаками, и при малом значении $r(\omega_j, \omega_k)$ для некоторой пары признаков соответствующие коэффициенты a_j и a_k должны мало отличаться друг от друга. В этом и заключается априорная информация о направляющем векторе разделяющей гиперплоскости, которую предлагается учитывать в процессе обучения в пространстве метрических признаков.

Эту дополнительную информацию можно внести в критерий обучения (3.2.3) в виде дополнительного квадратичного члена

$$J'(\mathbf{a}) = \frac{\alpha}{2} \sum_{j=1}^N \sum_{k=1}^N \tilde{\beta}_{jk} (a_j - a_k)^2 = \alpha \mathbf{a}^T \tilde{\mathbf{B}} \mathbf{a}, \quad \tilde{\mathbf{B}} = \begin{bmatrix} -\tilde{\beta}_{11} + \sum_{i=1}^N \tilde{\beta}_{1i} & \cdots & -\tilde{\beta}_{1N} \\ \vdots & \ddots & \vdots \\ -\tilde{\beta}_{N1} & \cdots & -\tilde{\beta}_{NN} + \sum_{i=1}^N \tilde{\beta}_{Ni} \end{bmatrix},$$

слагаемые которого имеют смысл штрафов на попарные различия коэффициентов a_j и a_k , общая величина которого регулируется выбором значения коэффициента $\alpha \geq 0$. Штрафные коэффициенты $\tilde{\beta}_{jk}$ должны возрастать с уменьшением расстояния $r(\omega_j, \omega_k)$ между признаками, т.е. объектами обучающей выборки, стремясь к бесконечно большим $\tilde{\beta}_{jk} \rightarrow \infty$ при $r(\omega_j, \omega_k) \rightarrow 0$. Например такому условию удовлетворяет соотношение $\tilde{\beta}_{jk} = 1/[r(\omega_j, \omega_k)]^\gamma$, где $\gamma > 0$ - параметр определяющий скорость возрастания штрафа при сближении признаков.

Тогда целевая функция (3.2.3.) будет включать в себя еще одно слагаемое

$$\frac{1}{2} (\mathbf{a}^T \mathbf{a} + \alpha \mathbf{a}^T \tilde{\mathbf{B}} \mathbf{a}) + C \sum_{j=1}^N \delta_j \rightarrow \min$$

или более компактно

$$\frac{1}{2} \mathbf{a}^T \mathbf{B} \mathbf{a} + C \sum_{j=1}^N \delta_j \rightarrow \min, \text{ где } \mathbf{B} = \mathbf{I} + \alpha \tilde{\mathbf{B}},$$

при ограничениях

$$g_i(\mathbf{a}^T \mathbf{x}_i) = g_i \left(\sum_{j=1}^N a_j x_{ij} \right) \geq 1 - \delta_i, \quad \delta_i \geq 0, \quad i = 1, \dots, N, \quad a_j \geq 0, \quad j = 1, \dots, N.$$

Служить мерой сходства и различия между объектами, а следовательно и между признаками, может не сама метрика $r(\omega', \omega'')$, а некоторая функция от нее $\mu(\omega', \omega'') = \varphi[r(\omega', \omega'')]$. Если используется убывающая функция, такая что $\varphi(r) \rightarrow 0$ при $r \rightarrow \infty$ и $\varphi(0) = \mu^0$ то характеристика симметричного парного отношения между объектами $\mu(\omega', \omega'') = \mu(\omega'', \omega')$ является мерой близости между ними принимающей минимальное значение $\mu(\omega, \omega) = \mu^0$ при совпадении двух объектов.

В частности в задачах молекулярной биологии сходства и различия между парами белков часто характеризуют близостью, определение которой строится на понятии гипотетической общей последовательности, из которой обе сравниваемые последовательности могут быть получены небольшим числом единичных пропусков вставок или замен аминокислот. При таком определении величина близости оказывается зависящей от длин сравниваемых последовательностей, а разные последовательности характеризуются разными значениями близости к самим себе.

В таком случае матрица близости между объектами обучающей выборки $\mu(\omega_j, \omega_k)$, $j, k = 1, \dots, N$ должна быть подвергнута предварительному преобразованию так, что бы диагональные элементы имели одинаковое значение. Это можно сделать, например, разделив каждое значение $\mu(\omega_j, \omega_k)$ на величину $\sqrt{\mu(\omega_j, \omega_j)}\sqrt{\mu(\omega_k, \omega_k)}$. После такой операции матрица останется симметричной, а все ее диагональные элементы будут иметь единичное значение, соответствующее максимальному значению близости при полном совпадении объектов.

4. Программный комплекс обучения распознаванию образов.

4.1. Общая характеристика программного комплекса

Предложенные алгоритмы обучению распознаванию образов были реализованы в виде программного комплекса под названием Space 5.3. Комплекс может работать как с тестовыми примерами для отладки предложенных алгоритмов, так и для обработки реальных данных из различных областей знаний, таких как, техника, медицина, социология и т.п. Также, описываемый программный комплекс может быть использован для проведения курса “Методы распознавания образов”, читаемого на кафедре автоматики и телемеханики ТулГУ.

Программный комплекс состоит из программной оболочки, позволяющей в диалоговом режиме ввести исходные данные, проверить их корректность, выполнить указанный алгоритм обработки и отобразить результат на экране дисплея, а так же подготовленные заготовки-функции, позволяющие на основе принципа наследования данных ООП легко встраивать новые задачи обработки

Программа-оболочка представляет собой 32-разрядное приложение Windows написанное на языке C++ (компилятор Visual C++ 5.0) в соответствии с принципами объектно-ориентированного подхода (ООП) и принципом свободной последовательности программ, управляемых событиями.

Space 5.3 позволяет:

- ввод и редактирование данных в наглядном текстовом виде
- просмотр многомерных данных в псевдо двумерном (2D) пространстве
- просмотр результатов работы в 2D-пространстве
- возможность работы с данными больших размеров (например, 1000 объектов в 200-мерном признаковом пространстве)
- отсутствие программных ограничений на размер данных
- возможность внедрения новых задач обработки на основе объектно-ориентированного подхода
- просмотр справки по системе в виде стандартного гипертекста Windows

Программный комплекс состоит из следующих модулей:

SPACE53.EXE
SPACE5.HLP
SPACE53.LGO

Исполнимый файл
Файл справки
Логотип программы

Входные данные

Стандартизованный файл данных представляет собой ASCII текстовый файл со строками различной длины и расширением .DAT. Этот файл содержит в себе информацию о размерности данных, информацию эксперта и непосредственно матрицу данных.

Технические требования к аппаратному обеспечению:

- процессор 80486 и выше
- ОЗУ – 8 МБ
- SVGA адаптер с разрешением 800x600 и 256 цветов
- устройство ввода типа «мышь»
- операционная система Windows'95 или Windows NT
- по крайней мере 2,5 МБ свободного дискового пространства

4.2. Структура файлов данных

Все алгоритмы, работающие в программно алгоритмическом комплексе, обращаются к данным, представленным в стандартизованном формате. Пользователю, при создании файла, содержащего новые данные, необходимо заботиться о соответствии этого файла существующему формату.

Стандартизованный файл данных представляет собой ASCII текстовый файл со строками различной длины и расширением .DAT. Этот файл содержит в себе информацию о размерности данных, информацию эксперта и непосредственно матрицу данных. Порядок следования данных изменять нельзя.

Входные последовательности разделены символом ":". В строке возможно наличие не более одного такого символа. Последовательность символов слева от ":" предназначена для комментариев, более наглядного представления, и может быть опущена. Сканируемое значение обычно представляет собой либо единственное число, либо последовательность чисел, разделенных пробелами. В качестве примера приводится часть файла A.DAT с данными, полученными при написании латинской буквы *a* различными людьми.

```
Number of objects: 6
Number of features: 6
Number of classes: 1
Cluster number: 6
Feature group number: 6
Actual features (>0 - used feature): 1 1 1 1 1 1
```



```

1 - a2-Seredin(#1): 0.000000 0.436457 0.579186 0.725564
1.049975 0.887381 1
2 - a3-Seredin(#1): 0.436457 0.000000 0.566639 0.461734
1.052930 0.975661 1
3 - a1-Dmitriev(#1): 0.579186 0.566639 0.000000 0.363853
1.156121 1.232744 1
4 - a2-Dmitriev(#1): 0.725564 0.461734 0.363853 0.000000
0.897810 1.135118 1
5 - a1-Dolgova(#1): 1.049975 1.052930 1.156121 0.897810
0.000000 0.542393 1
6 - a2-Dolgova(#1): 0.887381 0.975661 1.232744 1.135118 0.542393 0.000000 1

```

4.3. Загрузка и редактирование данных

Загрузка данных осуществляется в стандартном диалоге Windows посредством выбора соответствующего файла .DAT, для этого пользователю необходимо из главного окна программы (см. рис. 4.3.1) выбрать пункт меню “File|OpenTask”.

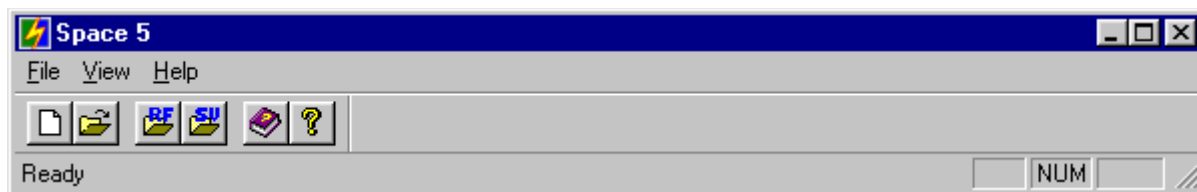


Рис. 4.3.1. Главное окно программного комплекса Space 5.3

В программном комплексе возможно редактирование данных с помощью встроенного многофункционального редактора. Для этого необходимо выбрать пункт меню “File|Open”. Необходимо помнить, что изменения, вносимые в файл данных .DAT, реально загружаются в динамические массивы памяти лишь после выбора задачи из меню и создания проблемного окна. Следовательно, после редактирования данных, они должны быть сохранены и проблемное окно открыто вновь. Алгоритмы обучения распознаванию образов имеют некоторые рабочие параметры. Начальные значения параметров хранятся в файле с тем же именем что и файл данных, но имеют расширение .DS5.

4.4. Отображение данных и решающего правила распознавания

4.4.1. Построение плоскости проецирования в многомерном пространстве

В процессе создания программного комплекса, реализующего алгоритм обучения распознаванию образов, реально встала проблема отображения результатов работы на дисплее ЭВМ. Если размерность признакового пространства

равна трем, уже приходится проводить визуализацию объектов распознавания с помощью громоздких изометрических построений. В случае же когда эта размерность больше трех, то вообще трудно себе представить геометрические аналогии. Реально же вектор состояния может включать в себя десятки, и даже сотни компонент. Вопрос: как отобразить многомерное признаковое пространство на плоскость экрана вычислительной машины? Разумеется, в прикладных задачах вряд ли возникнет такая необходимость, в них основой является непосредственно алгоритм распознавания, а способ интерпретации и показа результатов будет зависеть от конкретной решаемой задачи. В нашем случае программный комплекс выполняет демонстрационные функции, и было бы крайне полезно отобразить каким либо образом непосредственно объекты в многомерном пространстве.

Предлагается следующий способ визуализации. Пусть в признаковом пространстве существует разделяющая два класса гиперплоскость. Расположим плоскость экрана так, чтобы она была перпендикулярна гиперплоскости, и проекции объектов на нее располагались наименее плотно. В этом случае гиперплоскость будет отображаться на плоскость экрана в виде вертикальной линии, и основная задача будет заключаться в расчете проекций, которые будут отбрасывать объекты на плоскость экрана.

Пусть в пространстве \mathbf{R}^n найдена граничная гиперплоскость, например, как предложено в разделе 2, определяющая границу области ненулевого совместного распределения двух классов. Такая гиперплоскость задается ее направляющим вектором (вектор-нормаль к ней единичной длины) $\mathbf{c} = (c_1, \dots, c_n)^T$ и смещением c_0 , где $\|\mathbf{c}\| = 1$.

Будем полагать, что все векторы-строки \mathbf{x}^T матрицы данных $\mathbf{X}(N \times n)$, где N - число объектов, центрированы, то есть начало координат пространства \mathbf{R}^n перенесено в точку центра тяжести всей совокупности данных $\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_i$. Тогда каждый объект будет представлен своим вектором $\mathbf{x} - \bar{\mathbf{x}}$. В дальнейшем будем полагать, что векторы $\mathbf{x} \in \mathbf{R}^n$ всегда центрированы, если это специально не оговорено.

Будем полагать, что плоскость проецирования (плоскость экрана) проходит через начало координат так, что вектор \mathbf{c} лежит в ней. Пусть вектор \mathbf{b} - некоторый вектор единичной длины $\|\mathbf{b}\| = 1$, проведенный из начала координат, лежащий в плоскости экрана и ортогональный вектору \mathbf{c} . Тогда векторы \mathbf{c} и \mathbf{b} образуют в плоскости экрана прямоугольную систему координат $y_1 O y_2$, где ось y_1 образована вектором \mathbf{c} , а ось y_2 образована вектором \mathbf{b} . Тогда в плоскости

экрана некоторый вектор \mathbf{X} будет представлен вектором $\mathbf{y} = (y_1, y_2)^T$, координаты которого определены как $y_1 = \mathbf{c}^T \mathbf{x} + a_0$, $y_2 = \mathbf{b}^T \mathbf{x}$.

Очевидно, что в плоскости экрана гиперплоскость вида $\mathbf{c}^T \mathbf{x} + a_0 = 0$ будет представлена вертикальной линией параллельной оси y_2 , пересекающей ось y_1 в точке $-c_0$.

Так как вектор \mathbf{c} уже найден, и представляет собой направляющий вектор разделяющей гиперплоскости, то найдем вектор \mathbf{b} . Еще раз заметим, что ориентация в пространстве вектора \mathbf{b} , который ортогонален вектору \mathbf{c} и лежит в плоскости экрана, определяет ориентацию плоскости экрана. Пусть вектор \mathbf{b} ориентирован так, что все проекции объектов на него располагаются наименее плотно. Другими словами, пусть координаты y_2 объектов в плоскости экрана занимают как можно больший диапазон, то есть различаются между собой как можно сильнее. Назовем такой вектор \mathbf{b} оптимальным в указанном здесь смысле.

Наше интуитивное представление об оптимальном векторе \mathbf{b} следует уточнить, то есть формализовать. Это можно сделать различными способами.

Пусть, например, среди всей совокупности объектов в исходном пространстве найдены два самых далеких объекта, то есть найден соединяющий их вектор \mathbf{z} , координатами которого являются разности координат данных объектов. Тогда оптимальный вектор \mathbf{b} ориентирован так, проекция вектора \mathbf{z} на него имеет наибольшую длину.

Следовательно, найдем вектор \mathbf{b} из условия $\mathbf{z}^T \mathbf{b} \rightarrow \max$ при ограничениях $\mathbf{c}^T \mathbf{b} = 0$, $\mathbf{b}^T \mathbf{b} = 1$, где первое из них означает ортогональность, а второе означает $\|\mathbf{b}\| = 1$.

Составим функцию Лагранжа $L(\mathbf{b}, \lambda_1, \lambda_2) = \mathbf{z}^T \mathbf{b} + \lambda_1 \mathbf{c}^T \mathbf{b} + \lambda_2 (\mathbf{b}^T \mathbf{b} - 1)$, и найдем ее минимум из условия равенства нулю ее производных по неизвестным координатам вектора \mathbf{b} :

$$\nabla_{\mathbf{b}} L(\mathbf{b}, \lambda_1, \lambda_2) = \mathbf{z} + \lambda_1 \mathbf{c} + 2\lambda_2 \mathbf{b} = 0.$$

Отсюда

$$\mathbf{b} = -\frac{\lambda_1}{2\lambda_2} \mathbf{c} - \frac{1}{2\lambda_2} \mathbf{z}.$$

Пусть $\lambda'_1 = -\frac{\lambda_1}{2\lambda_2}$, $\lambda'_2 = -\frac{1}{2\lambda_2}$. Подставив эти значения и переобозначив $\lambda'_1 = \lambda_1$, $\lambda'_2 = \lambda_2$, окончательно получим

$$\mathbf{b} = \lambda_1 \mathbf{c} + \lambda_2 \mathbf{z}.$$

Используем первое ограничение и найдем λ_1 :

$$\mathbf{c}^T \mathbf{b} = \mathbf{c}^T (\lambda_1 \mathbf{c} + \lambda_2 \mathbf{z}) = \lambda_1 \mathbf{c}^T \mathbf{c} + \lambda_2 \mathbf{c}^T \mathbf{z} = \lambda_1 + \lambda_2 \mathbf{c}^T \mathbf{z} = 0, \quad \lambda_1 = -\lambda_2 \mathbf{c}^T \mathbf{z}.$$

Подставим найденное в выражение для \mathbf{b} . Заметим, что в соответствии с правилом согласования размерностей в матричных уравнениях, следует записать $\mathbf{b} = c\lambda_1 + z\lambda_2$, считая коэффициенты матрицами размера (1×1) . Тогда получим

$$\mathbf{b} = -\mathbf{c}\mathbf{c}^T z\lambda_2 + z\lambda_2 = (\mathbf{I} - \mathbf{c}\mathbf{c}^T)z\lambda_2,$$

где $\mathbf{I}(n \times n)$ - единичная матрица. Обозначим $\mathbf{C} = \mathbf{I} - \mathbf{c}\mathbf{c}^T$ и получим $\mathbf{b} = \mathbf{C}z\lambda_2$.

Найдем λ_2 из второго ограничения

$$(\mathbf{C}z\lambda_2)^T \mathbf{C}z\lambda_2 = \mathbf{z}^T \mathbf{C}^T \mathbf{C}z\lambda_2^2 = 1, \quad \lambda_2 = 1/\sqrt{\mathbf{z}^T \mathbf{C}^T \mathbf{C}z},$$

взяв положительное значение. Окончательно получим


$$\mathbf{b} = \mathbf{C}z/\sqrt{\mathbf{z}^T \mathbf{C}^T \mathbf{C}z}, \quad \text{где } \mathbf{C} = \mathbf{I} - \mathbf{c}\mathbf{c}^T.$$

4.4.2. Оптимальное проецирование данных на плоскость экрана

При проецировании возможны три случая. В первом случае исходное признаковое пространство является двухмерным. Тогда плоскость экрана $y_1 O y_2$ является просто исходным двухмерным пространством $x_1 O x_2$, в котором при пошаговом отображении работы алгоритма изменяется положение (наклон и смещения) трех параллельных границ (края и середина области совместного распределения данной пары классов), а объекты не изменяют своего положения. Положение границ определяется направляющим вектором $\tilde{\mathbf{a}}$ неединичной длины и соответствующими смещениями вдоль него от начала координат: $-a_0$, $0.5 - a_0$ и $1 - a_0$.

Во втором случае исходное признаковое пространство также является двухмерным, но плоскость экрана $y_1 O y_2$ образована вектором \mathbf{C} , где $\mathbf{c} = \tilde{\mathbf{a}}/\|\tilde{\mathbf{a}}\|$, $c_0 = a_0/\|\tilde{\mathbf{a}}\|$, $\|\mathbf{c}\| = 1$, и ортогональным вектором \mathbf{b} , где $\mathbf{b} = (-c_2, c_1)^T$, $\|\mathbf{b}\| = 1$. Тогда при пошаговом отображении работы алгоритма три границы изменяют только смещения, оставаясь взаимно параллельными и параллельными вертикальной оси $O y_2$. Объекты в плоскости экрана могут изменять свое положение, так как сам экран поворачивается в исходном пространстве соответственно поворотам вектора $\tilde{\mathbf{a}}$. Положение границ на оси $O y_1$ определяется смещениями $-a_0/\|\tilde{\mathbf{a}}\|$, $(0.5 - a_0)/\|\tilde{\mathbf{a}}\|$, $(1 - a_0)/\|\tilde{\mathbf{a}}\|$.

В третьем случае пространство является многомерным. Плоскость экрана находится как описано выше для многомерного пространства. Поэтому, все остается также как и во втором случае, за исключением того, что вектор \mathbf{b} находится специальным образом.

В случае, когда число классов обучения больше двух пользователь может наблюдать проекцию данных лишь относительно одной из пар классов. Выбор соответствующей пары осуществляется посредством диалога "Mapping Style" из окна задачи, нажав кнопку .

4.5. Обучение и распознавание

Для запуска алгоритма обучения распознаванию образов пользователю необходимо из меню главного окна программы (рис. 4.3.1) выбрать пункт меню "File|OpenTask". После этого на экран будет выведено окно задачи (рис. 4.5.1) и произойдет загрузка динамических массивов памяти из файлов данных.

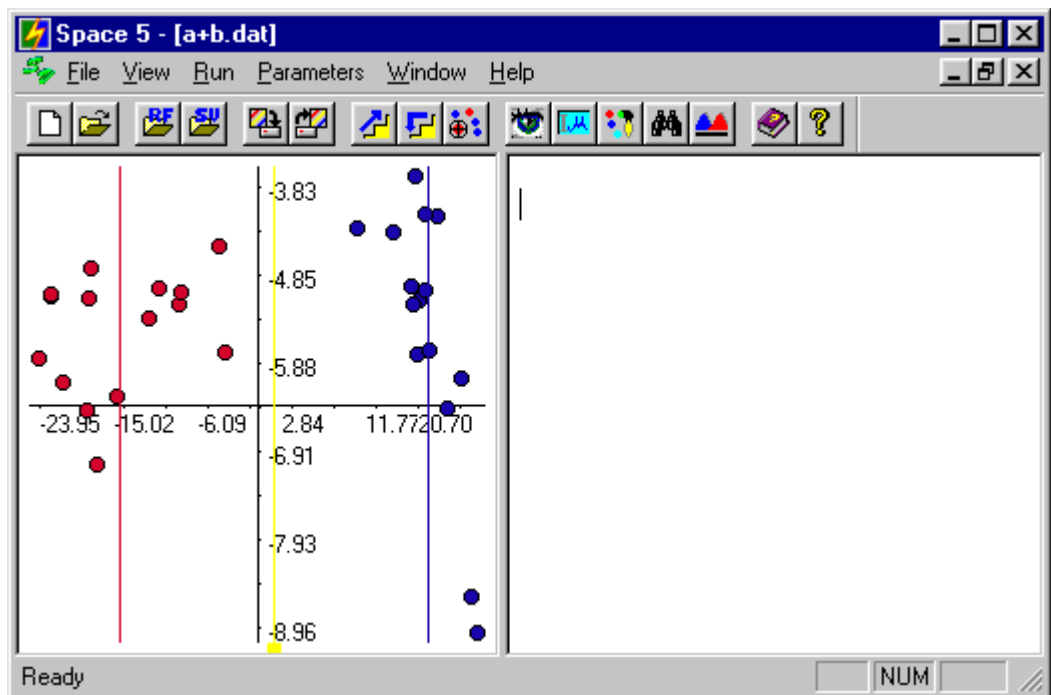


Рис. 4.5.1. Рабочее окно программы при выборе задачи обучения

Меню "Run" позволяет производить запуск итерационного алгоритма по шагам, до останова и возврат на начальное приближение. Кроме того, эти же функции можно выбрать посредством кнопок ускорителей:



- выполнение алгоритма до останова,



- возврат к начальному приближению.

Алгоритм, выполняемый в режиме "до останова", может быть остановлен нажатием кнопки "возврат".

Графическое соответствие реализуемого процесса выводится в виде двумерной проекции многомерного признакового пространства. Принцип

построения такой проекции описан в разделе 4.4. Направляющий вектор разделяющей гиперплоскости коллинеарен оси X экрана, и пользователь наблюдает след разделяющей гиперплоскости как вертикальную линию.

Одновременно с изменением графического представления решающих правил и данных на этапе обучения возможно отображение результатов в текстовом виде. Информация подобного рода содержится в текстовом буфере, который отображается в окне "Problem Status", и периодически обновляется (см. рис.4.5.1). Для просмотра более детальных сведений о решающем правиле пользователь может вывести подобную информацию в текстовое окно нажатием правой кнопки мыши на желтый квадрат внизу линии, отображающей разделяющую гиперплоскость (см. рис. 4.5.2).

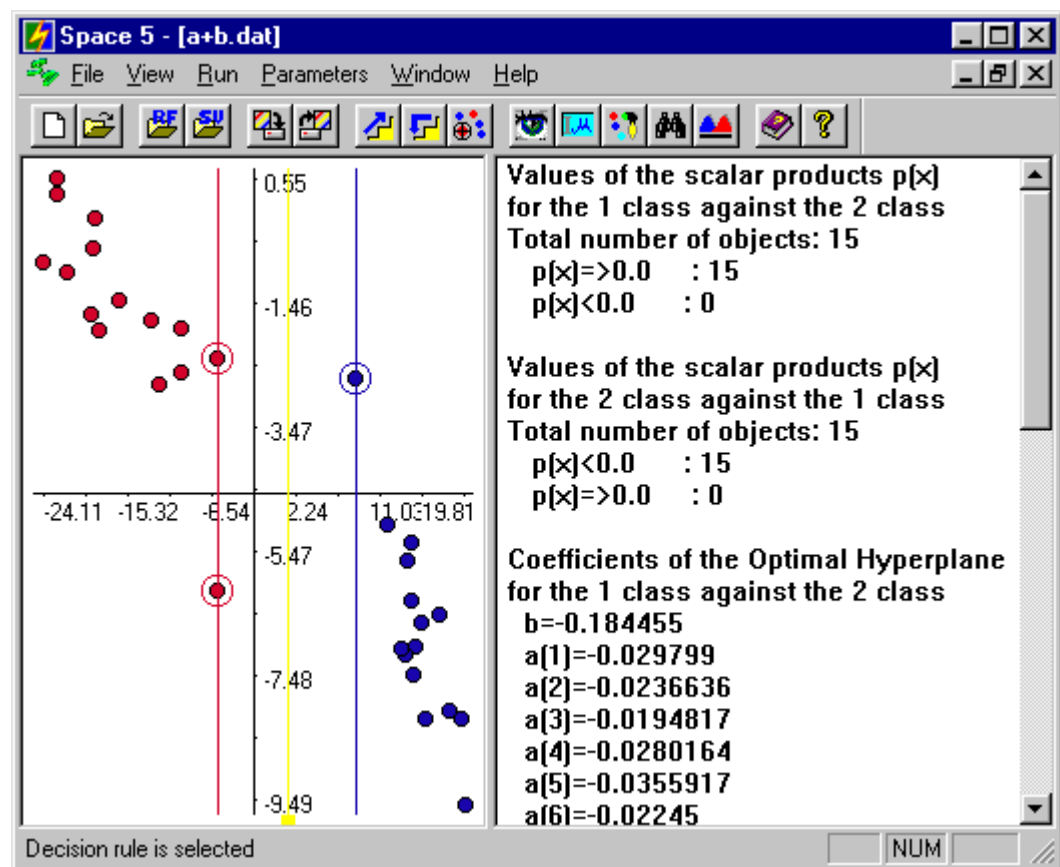




Рисунок 4.5.2 Информация о решающем правиле распознавания

Изменение параметров алгоритма осуществляется в диалоговом окне, выпадающем при нажатии кнопки . Начальные значения параметров загружаются из файла с тем же именем что и файл данных, и имеющего расширение .DS5. После изменения параметров в окне "Parameters of Algorithm" и завершения работы программы изменения не сохраняются. Для изменения начальных параметров в файле .DS4, пользователю необходимо воспользоваться каким либо внешним редактором, например NotePad.

Изменение вида отображения осуществляется нажатием кнопок  и .

После завершения процесса обучения пользователь может провести контроль качества решающего правила с помощью процедуры «скользящий контроль» (см. п.5.1.4., нажав кнопку ). Результаты процедуры скользящий контроль выводятся кратко в окне "Problem Status", и, полностью, в файле с расширением .loc. (см. рис. 4.5.3).

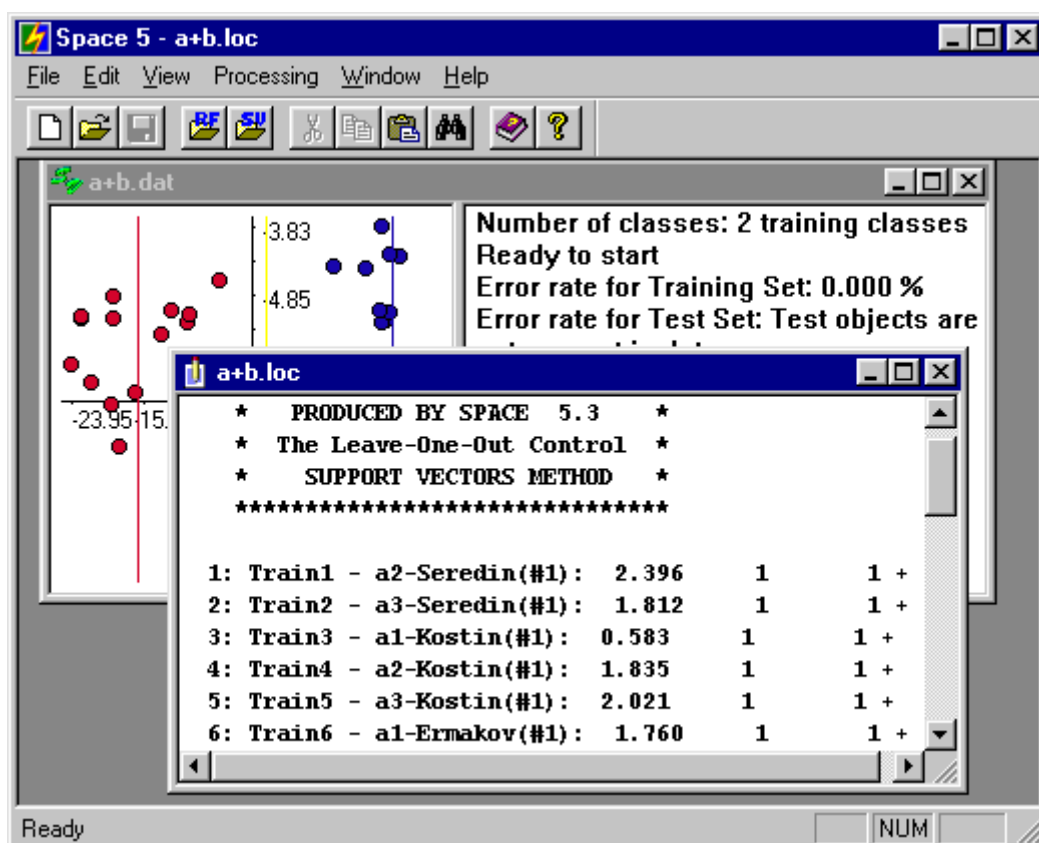



Рис. 4.5.3. Результат процедуры «скользящий контроль»


4.6. Сохранение и загрузка решающего правила

В программном комплексе Space 4.5 реализованы следующие функции:

- возможность сохранения текущего состояния задачи без выхода из программы,
- возможность загрузки предварительно сохраненных результатов и параметров,
- возможность применения полученного решающего правила к другим данным такой же природы.

Для обеспечения выполнения перечисленных функций в программный комплекс введено такое понятие как файл состояния задачи .STA. Пользователь

имеет возможность сохранить в этот файл полученное решающее правило, размерность данных, которым оно соответствует и параметры алгоритма. Для этого необходимо нажать кнопку  из окна задачи. Пользователю ни в коем случае не стоит редактировать этот файл, он является служебным. Кроме файла .STA на диске создается файл отчета с расширением .TXT. Именно этот файл, пользователь может использовать по своему усмотрению - редактировать, переименовывать, удалить.

Для вызова ранее сохраненного файла состояния для данных, по которым он был получен, либо для других данных такой же природы, пользователю необходимо выбрать кнопку . После этого будут загружены решающее правило и рабочие параметры алгоритма, и обновлено графическое отображение. В случае, если загружаемое решающее правило не соответствует размерности данных, будет выведено сообщение об ошибке.

5. Экспериментальное исследование алгоритмов обучения

5.1 Структура экспериментального материала

Модельные данные.

Для проверки качества решающего правила использованы следующие данные. Выборки образованы двумя нормальными распределениями с одинаковыми ковариационными матрицами в 25-мерном пространстве. Каждый объект выборки представляет собой последовательность из 25 отсчетов гладкой функции с наложенными возмущениями. Гладкость понимается в обычном смысле, т.е. функция является гладкой, если она плавно меняется при плавном изменении аргумента. В качестве таких функций выбраны синусоидальные гармоники одинаковой амплитуды, но разной частоты. Такие данные представляют собой сферические распределения, координаты центров которых определены значениям первых 25 отсчетов гладкой функции. Сами классы линейно неразделимы.

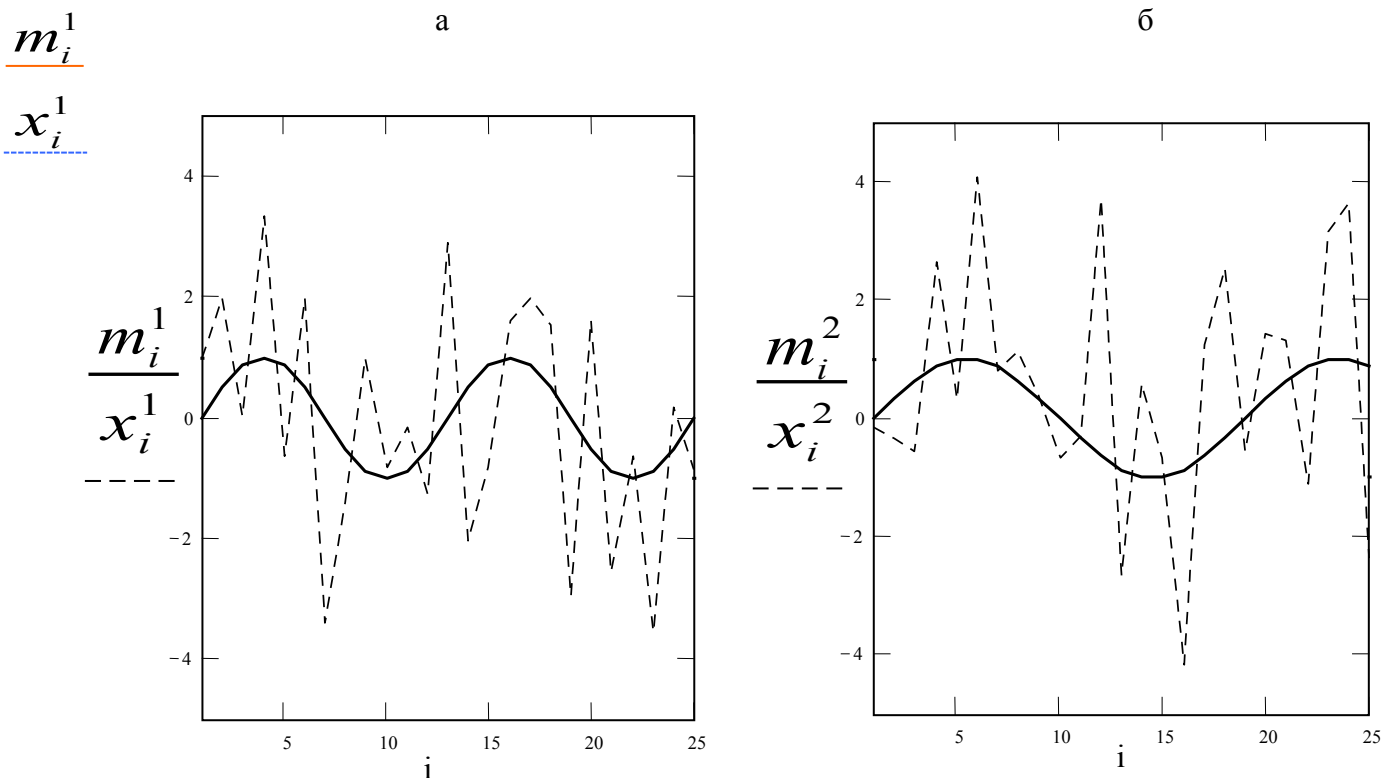


Рис.5 1.1 Отдельные признаки в составе векторов центров классов и отдельных объектов

На каждом рисунке сплошной линией показаны значения признаков центра класса, а пунктиром - значения признаков одного из объектов выборки класса.

Рукописные символы

Наглядным примером задачи распознавания образов в метрическом пространстве является задача распознавания рукописных символов, например, букв, при их вводе в ЭВМ непосредственно в процессе написания с помощью специального пера. При таком способе ввода каждый символ первоначально оказывается представленным сигналом, состоящим из двух компонент, а именно, текущих координат пера по вертикали и горизонтали, однако, может оказаться целесообразным использовать и дополнительные локальные характеристики процесса написания, например, угловой азимут мгновенного направления движения пера, его скорость, силу прижатия к бумаге, временные отрывы от нее, наклон и т.п. В качестве аргумента сигнала может выступать либо время, либо длина пути, пройденного пером от точки первого касания бумаги. На рис.5.1.2 представлен трехкомпонентный сигнал в виде функций координат и азимута движения пера от пройденного пути вдоль его траектории, полученный при написании рукописной буквы “d”.

В этой задаче трудно указать заранее фиксированное число признаков сигнала, которые могли бы сформировать пространство, удовлетворяющее гипотезе компактности. Нельзя использовать в качестве признаков и отсчеты сигнала, взятые с некоторым шагом вдоль оси аргумента, поскольку сигналы, полученные от разных написаний даже одного и того же символа неизбежно будут иметь разную длину, и, следовательно, не существует единого линейного пространства, в котором могли бы быть представлены написания распознаваемых символов [43].

Заметим, что разные варианты написания одного и того же символа естественно представить как результат некоторого нелинейного преобразования оси аргумента, приводящего к ее «короблению». Эти различия между разными написаниями, несущественные с точки зрения распознавания символов, легко компенсировать с помощью процедуры так называемого парного выравнивания (рис. 5.1.2), тогда остающееся несовпадение сигналов будет нести информацию об «истинной» непохожести сигналов, которую естественно принять в качестве рабочей метрики при построении процедуры обучения распознаванию символов.

На рис.5.1.3 приведен кадр программы, иллюстрирующий различие между введенной буквой *a* и эталонами букв *a* и *b*.

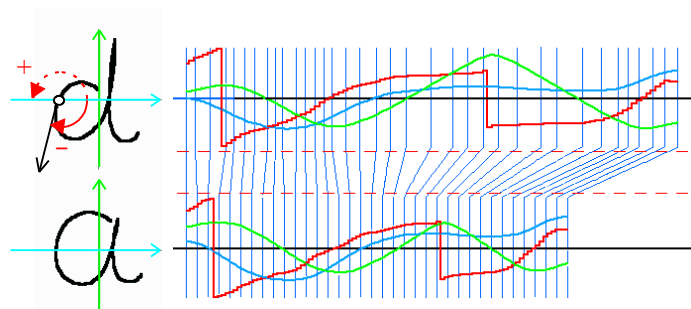


Рисунок. 5.1.2 Два сигнала в виде функций координат и азимута движения пера от пройденного пути вдоль его траектории, полученные при вводе рукописных символов в компьютер непосредственно в процессе написания. Совмещение произведено по значениям азимута.

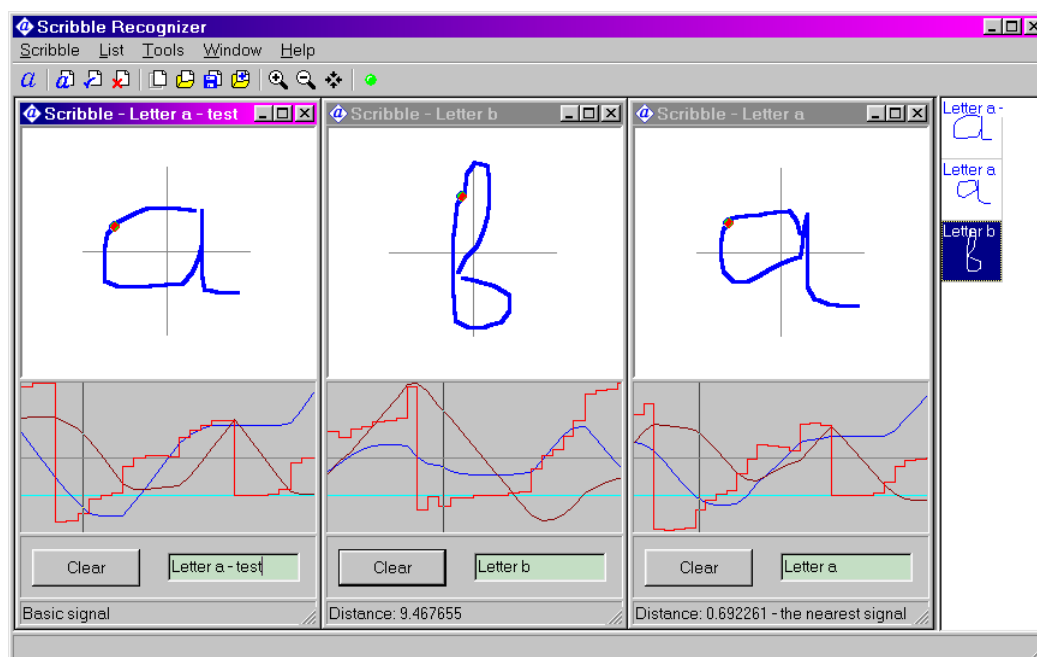


Рисунок 5.1.3. Экран работы программы “Scribble recognizer”. Нижняя часть окна показывает численное значение близости вновь введенного символа “a” к шаблонам символов “a” и “b”

Другим примером задач, требующих введения метрического пространства распознавания является задача классификации пространственной структуры белков опираясь на знание лишь первичной структуры (последовательности аминокислот). Информация о пространственной организации белка (третичная структура) является очень важной для понимания механизмов работы макромолекул и их функций. Третичную структуру макромолекул определяют экспериментальными методами (рентгеноструктурный анализ, ядерный магнитный резонанс). Эти методы являются чрезвычайно трудоемкими и требуют больших затрат времени, но позволяют получить достоверные сведения о пространственной организации молекул. Характерно, что для больших групп эволюционно родственных белков, подчас очень значительно отличающихся по первичной структуре и, значит, по распределению всех атомов в пространстве, способ укладки полипептидной цепи остается в главных чертах неизменным. С

другой стороны, при всем разнообразии пространственных структур белков удается выделить относительно небольшое число типов укладки полипептидной цепи. Налицо задача классификации – выделение групп белков, достаточно близких друг к другу по пространственной структуре. Один из фундаментальных принципов молекулярной биологии говорит о том, что последовательность аминокислотных остатков полипептидной цепи белка несет в себе всю информацию, необходимую и достаточную для формирования однозначной пространственной структуры. Учитывая это положение, в настоящее время большие усилия прилагаются для разработки методов предсказания третичной структуры молекул на основе известной первичной структуры. Разумеется, абсолютно точно произвести такой прогноз невозможно, остается надежда на то, чтобы правильно «угадать» группу к которой относится исследуемый белок.

В качестве примера, на рис. 5.1.4 представлено схематичное трехмерное изображение белка Cytochrome C4 и его первичная структура. На сегодняшний момент биологи выделяют определенные типы (семейства, фолды) известных пространственных структур [44]. Положительным результатом процедуры распознавания считается достоверное отнесение белка к одному из таких классов. Налицо задача распознавания образов.

Пространственная структура



Первичная структура:

| | |
|-----|------------|
| 1 | AGDAEAGQGK |
| 11 | VAVCGACHGV |
| 21 | DGNSPAPNFP |
| 31 | KLAGQGERYL |
| 41 | LKQLQDIKAG |
| 51 | STPGAPEGVG |
| 61 | RKVLEMTGML |
| 71 | DPLSDQDLED |
| 81 | IAAYFSSQKG |
| 91 | SVGYADPALA |
| 101 | KQGEKLFRRG |
| 111 | KLDQGMPACT |
| 121 | GCHAPNGVGN |
| 131 | DLAGFPKLGG |
| 141 | QHAAYTAKQL |
| 151 | TDFREGNRTN |
| 161 | DGDTMIMRGV |
| 171 | AAKLSNKDIE |
| 181 | ALSSYIQGLH |

Рисунок. 5.1..4. 3D представление и первичная структура белка Cytochrome C4

Имея в наличии информацию только о первичной структуре белков входящих в изучаемые данные, первым шагом представляется попытка получить некоторые количественные характеристики, которые бы отражали существо пространственной классификации. В настоящее время открыто более четырехсот признаков аминокислот (наиболее важные - гидрофобность, степень поляризации, размер и др.), однако, прямое использование этих признаков затруднено тем, что различные протеины имеют разную длину, и, следовательно, непосредственное представление первичных структур как векторных "сигналов" их свойств потребует учета специфики работы с задачами такого типа. Другой простейший подход получения количественных признаков из аминокислотной последовательности заключается в подсчете относительного числа остатков каждой из аминокислот к общей длине последовательности. В таком случае каждый протеин будет представлен точкой в двадцатимерном пространстве. Однако наиболее разумной представляется схема, использующая знания о взаимной близости аминокислотных последовательностей. Существуют априорные объективные данные о близости в химико-биологическом смысле всех пар аминокислот (210 – пар, включая близость «самой с собой»), которые обычно выражаются в виде матрицы соответствия 20×20 . Для двух аминокислотных последовательностей пытаются найти такое их взаимное соответствие, чтобы величина «невязки» близостей аминокислот была по возможности минимальной. При этом, для белков различной длины более короткий приходится искусственно «вытягивать» за счет введения в определенные позиции делеций, т.е. разрывать исходную структуру. Результат такой процедуры обычно выражается величиной несходства выравниваемых последовательностей. Опираясь на какую либо процедуру выравнивания последовательностей, например Fasta3 (<ftp://ftp.virginia.edu/pub/fasta>), строится матрица всех взаимных расстояний между первичными структурами. Такая матрица и рассматривается как метрическое пространство.

5.2 Схема эксперимента

Оценка качества решающего правила

В настоящей работе мы не затрагивали проблему оценивания классификаторов и выбор такой оценки. Обычно, о надежности алгоритма судят по качеству распознавания, определяемому как отношение числа верно распознанных объектов к их общему числу. Достаточно часто, особенно в

зарубежной литературе, применяется термин противоположный по смыслу качеству распознавания, а именно величина (вероятность) ошибки (error rate) распознавания. Интуитивно понятно, что разработчики процедур обучения распознаванию образов стремятся улучшить качество распознавания, или, соответственно, снизить степень ошибки предлагаемого ими классификатора. Необходимо понимать, что определяемое таким образом качество распознавания зависит от выборки, на которой проводится обучение. Например, если в последовательности много раз встречается ситуация, которую машина классифицирует не так, как учитель, то процент несовпадений будет велик, в то время как при другом составе последовательности он может оказаться мал. Поэтому необходимо заранее условиться, как будет определяться качество решающего правила, т.е. по какой последовательности будет исчисляться процент несовпадений.

Наименьшая, теоретически возможная вероятность ошибки распознавания определяется т.н. ошибкой байесовского классификатора, или просто байесовской ошибкой [20].

В реальных задачах распознавания образов, часто возникает ситуация когда классы не разделимы полностью, тогда одной из проблем статистической теории распознавания является проблема определения наилучшего возможного качества распознавания. Именно байесовский классификатор и дает такую оценку. Пусть для каждого класса Ω_k существует некоторая априорная вероятность его появления $p(\Omega_k)$, $k = 1, 2$ и в n -мерном пространстве \mathbb{R}^n задана условная плотность распределения $p(\mathbf{x} | \Omega_k)$ вектора \mathbf{X} относительно каждого класса Ω_k . Тогда байесовское решающее правило определяется как

$$p(\Omega_k | \mathbf{x}) = \frac{p(\Omega_k)p(\mathbf{x} | \Omega_k)}{p(\mathbf{x})}, \quad p(\mathbf{x}) = \sum_{j=1}^2 p(\Omega_j)p(\mathbf{x} | \Omega_j), \quad (5.2.1)$$

а классификатор, который относит вектор \mathbf{X} к классу с наибольшей апостериорной вероятностью, называемый байесовским, имеет вид

$$p(\Omega_1)p(\mathbf{x} | \Omega_1) \begin{matrix} > \\ < \end{matrix} p(\Omega_2)p(\mathbf{x} | \Omega_2) \rightarrow \mathbf{x} \in \begin{cases} \Omega_1, \\ \Omega_2 \end{cases} \quad (5.2.2)$$

и соответствующая ему ошибка называется байесовской ошибкой классификации:

$$E_b = 1 - \sum_{i=1}^2 \int_{\Omega_i} p(\Omega_i)p(\mathbf{x} | \Omega_i) d\mathbf{x}. \quad (5.2.3)$$

Следует отметить, что байесовская ошибка, определяемая выражением (5.2.3) предполагает интегрирование по многомерной, не всегда точно определенной условной плотности распределения, поэтому в явном виде байесовская ошибка классификации может быть вычислена непосредственно лишь для небольшого круга задач. В частности это удастся сделать, когда распределения являются нормальными с одинаковыми ковариационными матрицами. Покажем это. Решающее правило (5.2.2) можно переписать в виде

$$l(\mathbf{x}) = \frac{p(\mathbf{x}|\Omega_1)}{p(\mathbf{x}|\Omega_2)} - \frac{p(\Omega_2)}{p(\Omega_1)} \rightarrow \mathbf{x} \in \begin{cases} \Omega_1, \\ \Omega_2 \end{cases} \quad (5.2.4)$$

Величину $l(\mathbf{x})$ называют отношением правдоподобия. Величину $p(\Omega_2)/p(\Omega_1)$ называют пороговым значением отношения правдоподобия для данного решающего правила. Нам будет удобнее вместо отношения правдоподобия $l(\mathbf{x})$ удобно использовать величину $-\ln l(\mathbf{x})$. В этом случае решающее правило (5.2.4) примет вид

$$h(\mathbf{x}) = -\ln l(\mathbf{x}) = -\ln p(\mathbf{x}|\Omega_1) + \ln p(\mathbf{x}|\Omega_2) \underset{>}{\overset{<}{-}} \ln \{p(\Omega_1)/p(\Omega_2)\} \rightarrow \mathbf{x} \in \begin{cases} \Omega_1 \\ \Omega_2 \end{cases} \quad (5.2.5)$$

Направление неравенства изменилось, потому что использовалось отрицательное значение логарифма. Уравнения (5.2.4) и (5.2.5) называют байесовским критерием, минимизирующим ошибку решения.

Если $p(\mathbf{x}|\Omega_k)$ $k = 1, 2$ - нормальная случайная величина с вектором математического ожидания \mathbf{M}_k и ковариационной матрицей Σ_k , то решающее правило (8) приобретает вид

$$h(\mathbf{x}) = -\ln l(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{M}_1)^T \Sigma_1^{-1}(\mathbf{x} - \mathbf{M}_1) - \frac{1}{2}(\mathbf{x} - \mathbf{M}_2)^T \Sigma_2^{-1}(\mathbf{x} - \mathbf{M}_2) + \frac{1}{2} \ln \frac{|\Sigma_1|}{|\Sigma_2|} \underset{>}{\overset{<}{-}} \ln \{p(\Omega_1)/p(\Omega_2)\} \rightarrow \mathbf{x} \in \begin{cases} \Omega_1 \\ \Omega_2 \end{cases} \quad (5.2.6)$$

Уравнение (5.2.6) показывает, что решающая граница является квадратичной формой относительно вектора \mathbf{x} . В случае равных ковариационных

матриц $\Sigma_1 = \Sigma_2 = \Sigma$ граница становится линейной функцией относительно вектора \mathbf{x} :

$$h(\mathbf{x}) = (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} \mathbf{x} + \frac{1}{2} (\mathbf{M}_1^T \Sigma^{-1} \mathbf{M}_1 - \mathbf{M}_2^T \Sigma^{-1} \mathbf{M}_2) \quad (5.2.7)$$

$$\begin{matrix} < \\ -\ln\{p(\Omega_1)/p(\Omega_2)\} \\ > \end{matrix} \rightarrow \mathbf{x} \in \begin{cases} \Omega_1 \\ \Omega_2 \end{cases}$$

Поскольку решающее правило (5.2.7) представляет собой линейно преобразование n -мерного пространства в одномерное, то если \mathbf{x} является нормально распределенным случайным вектором, решающее правило $h(\mathbf{x})$ также будет нормальной случайной величиной. Поскольку $E\{\mathbf{x}|\Omega_i\} = \mathbf{M}_i$, то математическое ожидание и дисперсию $h(\mathbf{x})$ можно вычислить следующим образом:

$$\eta_i = E\{h(\mathbf{x})|\Omega_i\} = (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} E\{\mathbf{x}|\Omega_i\} + \frac{1}{2} (\mathbf{M}_1^T \Sigma^{-1} \mathbf{M}_1 - \mathbf{M}_2^T \Sigma^{-1} \mathbf{M}_2) \quad (5.2.8)$$

$$\eta_1 = -\frac{1}{2} (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} (\mathbf{M}_2 - \mathbf{M}_1) = -\eta \quad (5.2.9)$$

$$\eta_2 = +\frac{1}{2} (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} (\mathbf{M}_2 - \mathbf{M}_1) = +\eta \quad (5.2.10)$$

$$\begin{aligned} \sigma_i^2 &= E\{[h(\mathbf{x}) - \eta_i]^2|\Omega_i\} = E\{[(\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} (\mathbf{x} - \mathbf{M}_i)]^2|\Omega_i\} = \\ &= (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} E\{(\mathbf{x} - \mathbf{M}_i)(\mathbf{x} - \mathbf{M}_i)^T|\Omega_i\} \Sigma^{-1} (\mathbf{M}_2 - \mathbf{M}_1) = \\ &= (\mathbf{M}_2 - \mathbf{M}_1)^T \Sigma^{-1} (\mathbf{M}_2 - \mathbf{M}_1) = 2\eta \end{aligned} \quad (5.2.11)$$

На рис. 5.2.1. изображены плотности вероятности решающего правила $h(\mathbf{x})$, причем заштрихованные площади соответствуют вероятностям ошибки, обусловленным байесовским критерием, который минимизирует ошибку решения. Эти вероятности ошибок соответственно равны

$$\varepsilon_1 = \int_t^\infty p(h|\Omega_1) dh = \int_{(\eta+t)/\sigma}^\infty (2\pi)^{-\frac{1}{2}} \exp\left(-\frac{\xi^2}{2}\right) d\xi = \frac{1}{2} - \Phi\left(\frac{\eta+t}{\sigma}\right) \quad (5.2.12)$$

$$\varepsilon_2 = \int_{-\infty}^t p(h|\Omega_2) dh = \int_{(\eta-t)/\sigma}^\infty (2\pi)^{-\frac{1}{2}} \exp\left(-\frac{\xi^2}{2}\right) d\xi = \frac{1}{2} - \Phi\left(\frac{\eta-t}{\sigma}\right) \quad (5.2.13)$$

где

$$t = \ln\{p(\Omega_2)/p(\Omega_1)\},$$

$$\sigma^2 = \sigma_1^2 = \sigma_2^2 = 2\eta.$$

Общая ошибка считается как сумма ошибок по каждому классу

$$\varepsilon = \varepsilon_1 + \varepsilon_2 \quad (5.2.14)$$

Таким образом, если плотность вероятности отношения правдоподобия является нормальной, то вероятности ошибки можно вычислить пользуясь таблицей функции Лапласа $\Phi(\cdot)$, так как отношение правдоподобия – одномерная нормальная случайная величина. Например для используемых здесь данных наименьшая теоретически возможная вероятность ошибки составляет $P_{\min} = 2\%$

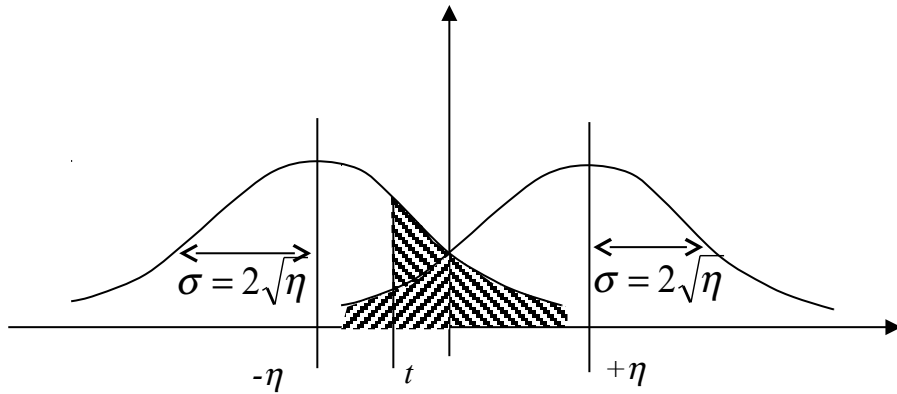


Рисунок 5.2.1. Плотность вероятности решающего правила $h(x)$.

Процедура проверки по генеральной совокупности

Генерируются две выборки: обучающая с малым числом объектов и контрольная с большим числом объектов каждого класса. Затем при некотором α по малонаполненной выборке проводится обучение, а полученное решающее правило проверялось на контрольной выборке с большим числом объектов. На обеих выборках по отношению числа неверно классифицированных объектов к их общему числу определяется вероятность ошибки распознавания в процентах. Далее величина штрафа α увеличивается, и эксперимент повторяется. Согласно предлагаемой гипотезе, при увеличении штрафа на негладкость вероятность ошибки распознавания на обучающей выборке должна увеличиваться из-за сужения допустимой области выбора решающего правила, а на контрольной выборке, наоборот, должна уменьшаться, из-за того, что регуляризованное решающее правило более реально отражает особенности генеральной совокупности.

Процедура «скользящий контроль»

Для эмпирической оценки качества распознавания М.Н. Вайнцвайгом была предложена процедура, впоследствии получившая название "скользящий контроль". Суть ее заключается в следующем. Из выборки удаляется один элемент и по оставшимся объектам проводится обучение. Затем на основании

полученного решающего правила удаленный элемент классифицируется. Если результат классификации совпадает с истинным классом объекта, то считается, что алгоритм дал верный результат, иначе - ошибся. Затем выбранный объект возвращается в выборку, из нее удаляется другой объект, и эксперимент повторяется. Такая процедура проводится над всеми точками выборки. Отношение числа ошибочно классифицированных векторов к размеру выборки и оценивает качество решающего правила.

В [12, 26] показано, что оценки скользящего контроля являются несмещенными, т.е. математическое ожидание результата контроля равно истинной величине качества. Полагают, хотя строгого доказательства этого утверждения не существует, что для большинства практически важных случаев дисперсия оценки "скользящий контроль" стремится к нулю с увеличением размера выборки примерно так же быстро, как дисперсия "экзамена". Отыскание дисперсии оценки метода "скользящего контроля" является одной из актуальных задач не только теории обучения распознаванию образов, но и теоретической статистики.

5.3 Результаты исследования эффективности штрафа на негладкость

Во избежание влияния на результат специфических особенностей малонаполненных выборок эксперимент проводится на пяти обучающих выборках с одинаковым числом элементов, а значение вероятности ошибки распознавания усредняется по ним. Для оценки того, как размер выборки влияет на эффективность штрафа на негладкость, опыты повторяются для различного числа элементов N в выборке, которое было как больше размерности признакового пространства, так и меньше его..

Результаты эксперимента представлены на рис. 5.3.1, где показана зависимость вероятности ошибки распознавания от величины штрафа на негладкость.

Анализируя эти зависимости, можно сделать следующие выводы. Во-первых, наибольший эффект штраф на негладкость дает для случая малонаполненных выборок, и эффективность его падает при увеличении числа объектов в выборке. Во-вторых, при увеличении α вероятность ошибки распознавания сначала уменьшается до некоторого значения, а затем начинает увеличиваться. Это связано с тем, что на область допустимых значений накладывается слишком сильное ограничение, и решающее правило становится очень "грубым".

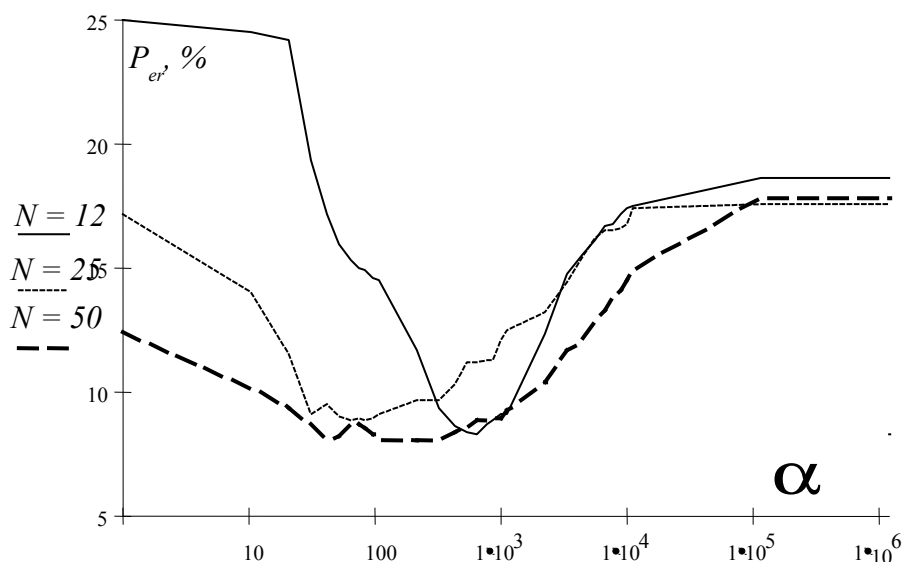


Рис. 5.3.1 Зависимость вероятности ошибки P_{er} от величины штрафа α на негладкость решающего правила.

5.4 Результаты исследования эффективности обучения распознаванию образов в метрическом пространстве.

Были сгенерированы обучающие выборки для любых двух строчных букв английского алфавита. Каждая выборка состояла из 30 символов, по пятнадцать объектов в каждом классе. Для всех выборок в метрическом пространстве было построено два решающих правила распознавания: без учета регуляризации и с учетом регуляризации. Для обоих решающих правил с помощью процедуры «скользящий контроль» была подсчитана ошибка распознавания. Результаты экспериментов представлены на рис. 5.4.1.

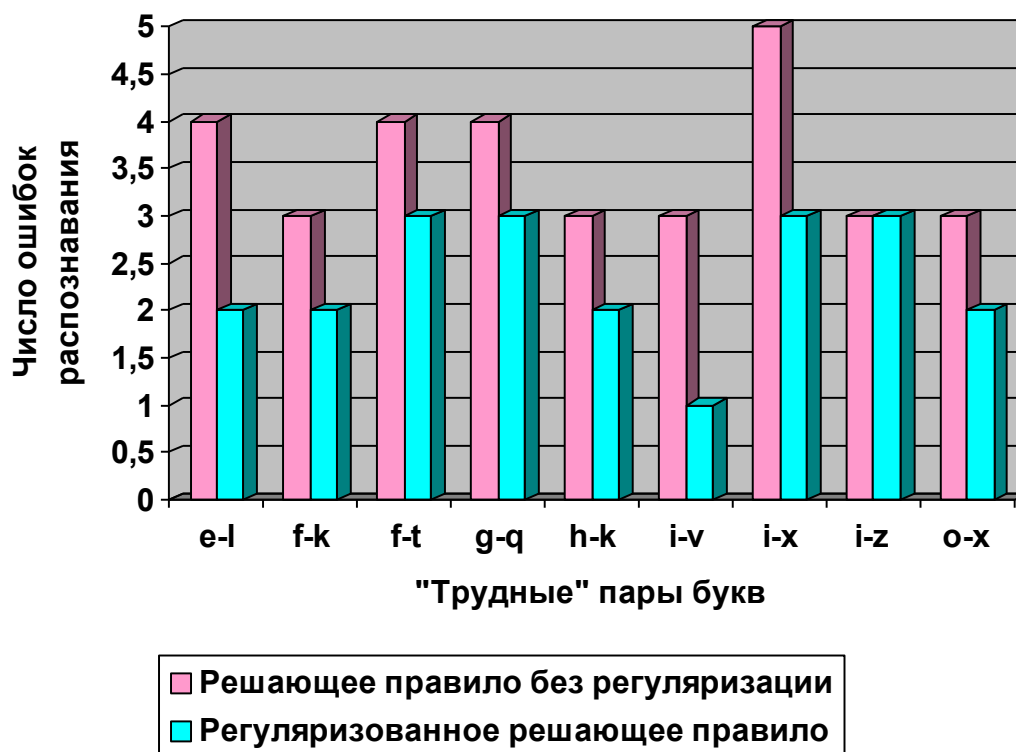


Рисунок 5.4.1 Иллюстрация необходимости регуляризации решающего правила распознавания в метрическом пространстве.

На диаграмме даны результаты процедуры «скользящих контроль» для девяти пар букв, распознавание которых в метрическом пространстве представляет собой особую сложность. Высота розовых колонок соответствует числу ошибочно классифицированных объектов в случае решающего правила без регуляризации, в то время как высота голубых колонок показывает, как ошибка распознавания может быть снижена за счет регуляризации.

Основные выводы

1. В работе дан анализ современных методов обучения распознаванию образов в пространствах числовых признаков.
2. Показано, что важной проблемой распознавания образов является обеспечение устойчивости получаемого решающего правила в случае короткой обучающей выборки по сравнению с размерностью признакового пространства.
3. Кроме того рассмотрена другая важная проблема широкого класса задач, в которых затруднительно, а порой и невозможно явно указать фиксированный набор легко измеряемых признаков объектов, в линейном пространстве которых задачу обучения распознаванию образов можно было бы решать как задачу построения разделяющей гиперплоскости.
4. Показана целесообразность необходимости учета априорной информации о классе решающих правил.
5. Предложен новый подход к проблеме регуляризации решающего правила, полученного при обучении распознаванию на многомерных данных, упорядоченных вдоль оси некоторого аргумента. Типичным примером объектов распознавания такого вида являются сигналы.
6. Для прикладных задач, в которых затруднительно, а порой и невозможно явно указать фиксированный набор легко измеряемых признаков объектов вместо линейного векторного признакового пространства предлагается рассматривать множество всех потенциальных объектов непосредственно как метрическое пространство.
7. Разработан алгоритм построения оптимальной разделяющей гиперплоскости по обучающей выборке объектов двух классов.
8. Разработан алгоритм построения системы оптимальных решающих правил на основе оптимальной разделяющей гиперплоскости для случая многих классов.
9. Разработаны приемы регуляризации алгоритмов обучения распознаванию сигналов с учетом требования гладкости линейного решающего правила как средство повышения устойчивости процесса обучения.
10. Разработан алгоритм построения решающего правила в метрическом пространстве для обучающей выборки двух классов
11. Разработан метод повышения качества решающего правила в метрическом пространстве
12. Разработан программно-алгоритмический комплекс, реализующий предложенные алгоритмы обучения распознаванию образов и обеспечивающий наглядное представление процесса и результата обучения.

13. Исследована работоспособность предложенных алгоритмов на реальных и модельных данных.

Министерство образования и науки Российской Федерации

Федеральное государственное образовательное учреждение
Высшего профессионального образования

Тульский государственный университет

Кафедра автоматике и телемеханики
(наименование выпускающей кафедры)

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
Выпускная квалификационная работа магистра

направление 230100
(код)

Информатика и вычислительная техника
(наименование)

программа подготовки 23010019
(код)

Компьютерный анализ и интерпретация данных
(наименование)

Исследование модели логистической регрессии в задаче
распознавания образов для нестационарной генеральной
совокупности
(тема)

Студент группы 240661/19 _____ Турков П.А.
(подпись, дата) (фамилия, инициалы)

Научный руководитель _____ Красоткина О.В.
(подпись, дата) (фамилия, инициалы)

Заведующий кафедрой _____ Фомичев А.А.
(подпись, дата) (фамилия, инициалы)

Руководитель
магистерской программы _____ Копылов А.В.
(подпись, дата) (фамилия, инициалы)

Тула 2012

Аннотация

Существуют задачи обучения распознаванию образов, в которых влияние некоторого скрытого фактора приводит к изменению свойств генеральной совокупности. Данная работа предлагает математическое и алгоритмическое описание для задач распознавания такого вида.

Представленное вероятностное обоснование основано на байесовском подходе к методу логистической регрессии для нахождения параметров решающего правила. Полученная процедура распознавания образов построена на общем принципе динамического программирования и обладает линейной вычислительной сложностью в противоположность полиномиальной для общего случая процедуры распознавания.

Пояснительная записка содержит 53 страницы, 4 рисунка, 4 таблицы.

Содержание

| | |
|---|----|
| Введение | 6 |
| 1. Проблема обучения распознаванию образов в условиях нестационарной генеральной совокупности | 8 |
| 2. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности | 9 |
| 2.1. Классическая задача обучения распознаванию образов | 9 |
| 2.2. Задача обучения распознаванию образов в нестационарной генеральной совокупности | 10 |
| 2.3. Вероятностное описание нестационарности параметров решающего правила | 14 |
| 3. Существующие методы обучения распознаванию образов в условиях нестационарной генеральной совокупности | 16 |
| 4. Обобщение метода опорных векторов для нестационарной генеральной совокупности | 20 |
| 4.1. Метод опорных векторов в классической задаче обучения распознаванию образов | 20 |
| 4.2. Вероятностное обоснование метода опорных векторов..... | 22 |
| 4.3. Распространение метода опорных векторов на случай нестационарной генеральной совокупности | 25 |
| 4.4. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода опорных векторов..... | 27 |
| 4.5. Численная реализация процедуры динамического программирования для оценивания параметров оптимальной разделяющей гиперплоскости .. | 28 |
| 5. Обобщение модели логистической регрессии на случай нестационарной генеральной совокупности | 36 |
| 5.1. Модель логистической регрессии в задаче обучения распознаванию образов..... | 36 |

| | |
|--|----|
| 5.2. Вероятностное обоснование модели логистической регрессии | 37 |
| 5.3. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода логистической регрессии..... | 39 |
| 5.4. Численная реализация процедуры динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью модели логистической регрессии | 41 |
| 6. Экспериментальное исследование | 47 |
| 6.1. Экспериментальное исследование на модельных данных | 47 |
| 6.2. Экспериментальное исследование на реальных данных..... | 48 |
| Заключение | 52 |
| Список использованных источников | 53 |

Введение

В последние годы распознавание образов находит все большее применение в повседневной жизни. Распознавание речи и рукописного текста значительно упрощает взаимодействие человека с компьютером, распознавание печатного текста используется для перевода документов в электронную форму.

В классической постановке задачи распознавания универсальное множество, называемое генеральной совокупностью, разбивается на части-образы, также называемые классами. Образ какого-либо объекта задается набором его частных проявлений. Методика отнесения элемента к какому-либо образу называется решающим правилом. В данной работе мы будем рассматривать задачу обучения с учителем, т.е. для построения решающего правила будет использоваться некоторое множество объектов, на которых известна их скрытая характеристика –образ (класс), к которому данный объект относится. Наиболее известной и изученной является следующая ситуация: распознавание производится на множестве выбранных из генеральной совокупности объектов, свойства которых не изменяются со временем. Очевидно, что в реальных условиях это не так, но часто изменениями характеристик объектов можно пренебречь, поэтому данный подход вполне применим в большей части практических случаев. Однако, при прогнозировании сложных явлений, например, социально-экономических или биологических процессов, предположение о стационарности ведет к недопустимо большой ошибке при классификации.

Целью данной работы является исследование модели логистической регрессии с целью последующего создания алгоритма обучения распознаванию образов для случая нестационарной генеральной совокупности.

Текст работы состоит из семи основных разделов. Первый раздел представленной работы описывает задачу обучения распознаванию образов в нестационарной генеральной совокупности, второй содержит краткую

характеристику существующих методов для задач распознавания указанного типа. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности представлена в третьей части. Четвертая часть посвящена методу распознавания образов в нестационарной генеральной совокупности, построенному с использованием модели опорных векторов. Указанный метод был исследован в рамках бакалаврской работы[22]. В последующем разделе представлено описание классического метода логистической регрессии. Построение метода обучения распознаванию в нестационарной генеральной совокупности на основе модели логистической регрессии содержится в шестой части. В последнем седьмом разделе описаны результаты экспериментального исследования построенного алгоритма распознавания образов в нестационарной генеральной совокупности.

1. Проблема обучения распознаванию образов в условиях нестационарной генеральной совокупности

Обычно в задачах распознавания образов предполагается, что свойства генеральной совокупности неизменны на протяжении всего процесса обучения. Однако мы можем столкнуться с задачами иного рода, в которых влияние каких-то скрытых факторов может привести к большим или меньшим изменениям в генеральной совокупности и, как следствие, в решающем правиле. Для такой ситуации обычно используются термины «нестационарная генеральная совокупность» («non-stationary environment») или «смещение концепта» («concept drift»).

Под смещением здесь понимается изменение свойств анализируемого явления, вследствие чего происходит «дрейф» решающего правила в признаковом пространстве. Поскольку единственным источником информации об исследуемом явлении являются объекты обучения, необходимо постоянное пополнение обучающего множества объектами, содержащими наиболее адекватные на данный момент данные о состоянии генеральной совокупности, что означает практически постоянный рост размера обучающей выборки. Возникает задача инкрементного обучения, когда после завершения построения решающего правила по заданному обучающему множеству в распоряжение разработчика поступают дополнительные объекты с известной для них скрытой характеристикой, которые было бы желательно использовать для дополнительного обучения, т.е. коррекции уже созданного классификатора. Однако, в случае нестационарной генеральной совокупности построенный в следующий момент времени классификатор может кардинально отличаться от предыдущего.

Таким образом, требуется по непрерывно поступающим объектам исследования осуществлять адаптацию решающего правила к происходящим изменениям в генеральной совокупности.

2. Постановка задачи обучения распознаванию образов в нестационарной генеральной совокупности

2.1. Классическая задача обучения распознаванию образов

В современной информатике интенсивно и успешно развивается методология анализа данных, направленная на алгоритмизацию поиска эмпирических закономерностей $y(\omega): \Omega \rightarrow Y$ во множествах объектов $\omega \in \Omega$, вообще говоря, произвольной природы. Требуется, анализируя предъявленный массив данных, представляющий собой совокупность значений некоторых характеристик объектов $(x(\omega_j), y(\omega_j))$ в пределах их доступного подмножества (обучающей выборки) $\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega$, продолжить наблюдаемую связь между этими характеристиками на все гипотетическое множество объектов $\omega \in \Omega$ (генеральную совокупность), чтобы можно было в дальнейшем для новых объектов, не участвовавших в обучении $\omega \in \Omega \setminus \Omega^*$, оценивать значения одних (целевых) характеристик $y(\omega)$ через значения других характеристик $x(\omega)$, более доступных для непосредственного измерения: $\hat{y}(x(\omega))$ [1,2,3,4].

В частности, если целевая характеристика принимает значения из конечного неупорядоченного множества $Y = \{y^{(1)}, \dots, y^{(m)}\}$, то такую задачу принято называть задачей обучения распознаванию образов. В простейшем случае, если рассматриваются лишь два класса, то в качестве двухэлементного множества их индексов обычно принимают множество $Y = \{-1, 1\}$.

Ограничимся рассмотрением задачи обучения распознаванию двух классов объектов, наблюдаемых через значения конечного числа их действительных признаков $x_i(\omega)$, $i = 1, \dots, n$. В этом случае каждый объект генеральной совокупности $\omega \in \Omega$ представлен точкой в конечномерном линейном пространстве признаков $\mathbf{x}(\omega) = (x_1(\omega), \dots, x_n(\omega)) \in \mathbb{R}^n$, а его скрытая фактическая принадлежность к одному из двух классов определяется

значением индекса класса $y(\omega) \in \{1, -1\}$. Тогда обучающая выборка Ω^* примет вид $\{\mathbf{X}, \mathbf{Y}\}$, где $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ и $\mathbf{Y} = \{y_i\}_{i=1}^N$. Будем строить линейный классификатор:

$$a(\mathbf{x}) = \text{sign}(\mathbf{a}^T \mathbf{x} + b) \quad (1)$$

Классический подход к обучению распознаванию двух классов объектов, развитый В.Н. Вапником [4], основан на понимании модели генеральной совокупности в виде разделяющей гиперплоскости $\mathbf{a}^T \mathbf{x} + b = 0$ в пространстве признаков \mathbb{R}^n , определяемой направляющим вектором $\mathbf{a} \in \mathbb{R}^n$ и параметром положения $b \in \mathbb{R}$, априори неизвестными наблюдателю. Предполагается, что соответствующая линейная дискриминантная функция $f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + b$ принимает в основном положительные значения для объектов первого класса и отрицательные – для второго:

$$f(\mathbf{x}(\omega)) = \mathbf{a}^T \mathbf{x}(\omega) + b \begin{cases} \text{преимущественно} > 0, & \text{если } y(\omega) = 1, \\ \text{преимущественно} < 0, & \text{если } y(\omega) = -1. \end{cases} \quad (2)$$

Обучающая совокупность $\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega$, содержащая примеры объектов обоих классов, определяет два конечных множества точек в \mathbb{R}^n :

$$\Omega^* = \{\omega_j, j = 1, \dots, N\} \subset \Omega, \quad \mathbf{x}_j = \mathbf{x}(\omega_j) \in \mathbb{R}^n, \quad y_j = y(\omega_j) \in \{1, -1\}. \quad (3)$$

Требуется по обучающему множеству определить значения параметров \mathbf{a} и b для решающего правила (1).

2.2. Задача обучения распознаванию образов в нестационарной генеральной совокупности

Мы остаемся в рамках линейного подхода к распознаванию, и предполагаем, что основное свойство нестационарной генеральной совокупности выражается изменяющейся во времени разделяющей гиперплоскостью, характеризующей преимущественное различие векторов признаков объектов двух классов.

Изменяющаяся во времени разделяющая гиперплоскость полностью определяется своим направляющим вектором и параметром положения,

которые должны рассматриваться как функции времени $\mathbf{a}_t: T \rightarrow \mathbb{R}^n$ и $b_t: T \rightarrow \mathbb{R}$, где T - упорядоченное множество моментов времени. Соответственно, всякий объект $\omega \in \Omega$ рассматривается всегда только вместе с указанием момента времени $t \in T$, в который он предъявлен (ω, t) :

$$f_t(\mathbf{x}(\omega)) = \mathbf{a}_t^T \mathbf{x}(\omega) + b_t \begin{cases} > 0, & y(\omega_t) = 1, \\ < 0, & y(\omega_t) = -1. \end{cases}$$

В классической постановке задачи обучения распознаванию образов каждый объект обучающей совокупности $\{\omega_j, j = 1, \dots, N\}$ представлен вектором его признаков и индексом принадлежности определенному классу, так что выборка в целом является множеством пар $\{(\mathbf{x}_j \in \mathbb{R}^n, y_j = \pm 1), j = 1, \dots, N\}$

В нестационарной интерпретации задачи обучения распознаванию образов каждый объект дополнительно характеризуется также моментом времени, в который был измерен его вектор признаков. В результате обучающая совокупность приобретает структуру множества троек $\{(\mathbf{x}_t \in \mathbb{R}^n, y_t = \pm 1, D_t), t = 1, \dots, T\}$, а учитывая возможность поступления в момент времени D_t нескольких объектов:

$\{(\mathbf{x}_j \in \mathbb{R}^n, y_j = \pm 1, D_t), j = (N_{t-1} + 1), K, N_t, t = 1, \dots, T, N_0 = 0\}$, где N_t - индекс объекта, поступившего первым в соответствующий момент времени.

Согласно принятой концепции нестационарной генеральной совокупности в разные моменты времени D_t из числа моментов, определяемых обучающей последовательностью, скрытая от наблюдателя разделяющая гиперплоскость характеризуется разными неизвестными значениями направляющего вектора $\mathbf{a}_t = \mathbf{a}_{D_t} \in \mathbb{R}^n$ и параметра положения $b_t = b_{D_t} \in \mathbb{R}$. Таким образом, объективно существует двухкомпонентный временной ряд со скрытой и наблюдаемой компонентами:

| | | | | | | | | |
|------------------------------------|--------------------------|------------------------------|-----|----------------------------------|----------------------------------|-----|----------------------------------|----------------------------------|
| время | D_1 | D_2 | ... | D_t | D_{t+1} | ... | D_{T-1} | D_T |
| параметры гиперплоскости | (a_1, b_1) | (a_2, b_2) | ... | (a_t, b_t) | (a_{t+1}, b_{t+1}) | ... | (a_{T-1}, b_{T-1}) | (a_T, b_T) |
| вектора признаков и индексы класса | $(x_j, y_j)_{j=1}^{N_1}$ | $(x_j, y_j)_{j=N_1+1}^{N_2}$ | ... | $(x_j, y_j)_{j=N_{t-1}+1}^{N_t}$ | $(x_j, y_j)_{j=N_t+1}^{N_{t+1}}$ | ... | $(x_j, y_j)_{j=N_{T-1}+1}^{N_T}$ | $(x_j, y_j)_{j=N_T+1}^{N_{T+1}}$ |

В классической постановке задачи распознавания образов нет свободы для интерпретации задачи обучения – требуется, анализируя предъявленную обучающую совокупность $\{(x_j, y_j), j=1, \dots, N\}$, дать оценку параметров разделяющей гиперплоскости (a, b) .

В рассматриваемой нами динамической постановке, задача обучения превращается в задачу анализа многокомпонентного временного ряда, в котором требуется, анализируя наблюдаемую компоненту, дать оценку скрытой компоненты. Это стандартная задача анализа временных рядов, специфика которой заключается лишь в предполагаемой модели связи между скрытой и наблюдаемой компонентами сигнала.

Согласно классификации задач оценивания скрытой компоненты сигнала, введенной Норбертом Винером [5], мы будем различать три основных вида задач обучения [6].

Задача фильтрации обучающей последовательности. Пусть D_t – момент поступления очередного объекта, к которому уже зарегистрированы векторы признаков и индексы классов объектов

$$\left\{ \dots, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \left\{ (x_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right\},$$

поступивших в предыдущие моменты времени до текущего момента включительно $(\dots, D_{t-2}, D_{t-1}, D_t)$. Требуется, анализируя уже поступившую часть обучающей последовательности, дать оценку параметров разделяющей гиперплоскости в текущий момент времени:

$$(\hat{\mathbf{a}}_{N_t|N_t}, \hat{\mathbf{b}}_{N_t|N_t}) = (\hat{\mathbf{a}}_{N_{D_t}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_t}|N_{D_t}}) =$$

$$= F_t \left(\dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right)$$

Здесь двойная индексация оценки $N_t | N_t$ указывает на то, что ищется оценка параметров разделяющей гиперплоскости в момент поступления N_t -го объекта после получения ровно N_t элементов обучающей последовательности.

Задача интерполяции обучающей последовательности. Пусть, по-прежнему, обучающая последовательность зарегистрирована вплоть до некоторого момента времени D_t , так что доступна последовательность

$$\left\{ \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right\}.$$

Требуется оценить параметры разделяющей гиперплоскости в некоторый предыдущий момент времени D_k , $k < t$:

$$(\hat{\mathbf{a}}_{N_k|N_t}, \hat{\mathbf{b}}_{N_k|N_t}) = (\hat{\mathbf{a}}_{N_{D_k}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_k}|N_{D_t}}) =$$

$$= F_t \left(\dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{k-1} + 1), K, N_k \right\}, \dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right)$$

Двойная индексация $N_k | N_t$ подчеркивает, что речь идет об оценке параметров разделяющей гиперплоскости в момент поступления N_k -го объекта после того, как зарегистрирована вся обучающая последовательность длины N_t .

Задача экстраполяции обучающей последовательности (задача прогноза). Анализируя часть обучающей последовательности, поступившую до некоторого момента времени D_t включительно, требуется оценить значения параметров разделяющей гиперплоскости в некоторый будущий момент времени $\tau > D_t$:

$$(\hat{\mathbf{a}}_{N_{\tau}|N_t}, \hat{\mathbf{b}}_{N_{\tau}|N_t}) = (\hat{\mathbf{a}}_{N_{D_t}|N_{D_t}}, \hat{\mathbf{b}}_{N_{D_t}|N_{D_t}}) =$$

$$= F_t \left(\dots, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-3} + 1), K, N_{t-2} \right\}, \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-2} + 1), K, N_{t-1} \right\}, \right.$$

$$\left. \left\{ (\mathbf{x}_j \in \mathbb{R}^n, y_j), j = (N_{t-1} + 1), K, N_t \right\} \right).$$

2.3. Вероятностное описание нестационарности параметров решающего правила

Будем полагать, что в нулевой момент времени априорная плотность распределения параметров разделяющей гиперплоскости является равномерной и равно единице на всей числовой оси. Поскольку ее интеграл не равен единице, плотностью вероятности в классическом понимании она не является. Однако в [7] представлена возможность интерпретации такой функции как распределений вероятности. Подобные распределения называются несобственными.

В [8] предлагается для учета динамики решающего правила направляющий вектор $\mathbf{a}_j \in \mathbb{R}^n$ рассматривать как случайный стационарный процесс:

$$\mathbf{a}_t = q\mathbf{a}_{t-1} + \xi_t, M(\xi_t) = \mathbf{0}, M(\xi_t \xi_t^T) = d\mathbf{I}, 0 \leq q < 1 \quad (4)$$

В силу стационарности процесса \mathbf{a}_j имеем $M(\mathbf{a}_j) = \mathbf{0}$,
 $M(\mathbf{a}_j \mathbf{a}_j^T) = M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) = M(\mathbf{a}_0 \mathbf{a}_0^T) = \mathbf{I}$

Справедлива следующая

Теорема 1 [9]. Для случайного стационарного процесса (4) справедливо соотношение $q = \sqrt{1-d}$.

Доказательство

Т.к. случайные величины \mathbf{a}_j , ξ_j независимы, то справедливо следующее соотношение:

$$M(\mathbf{a}_j \mathbf{a}_j^T) = q^2 M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) + M(\xi_j \xi_j^T)$$

Подставляем в это выражение следующие соотношения: $M(\xi_j \xi_j^T) = d\mathbf{I}$,

$M(\mathbf{a}_j \mathbf{a}_j^T) = M(\mathbf{a}_{j-1} \mathbf{a}_{j-1}^T) = M(\mathbf{a}_0 \mathbf{a}_0^T) = \mathbf{I}$, тогда получим

$$\begin{aligned}
(1-q^2)\mathbf{I} &= d\mathbf{I} \\
q^2 &= 1-d \\
q &= \sqrt{1-d}
\end{aligned}$$

Что и требовалось доказать.

Таким образом, априорное распределение параметра \mathbf{a}_t :

$$\psi_a(\mathbf{a}_t | \mathbf{a}_{t-1}) = N(\mathbf{a}_t | \sqrt{1-d}\mathbf{a}_{t-1}, d\mathbf{I})$$

Относительно параметра b_t аналогично предполагаем, что он является случайным стационарным процессом:

$$\begin{aligned}
b_t &= b_{t-1} + \eta_t, M(\eta_t) = 0, M(\eta_t^2) = d' \\
\psi_b(b_t) &= N(b_t | b_{t-1}, d')
\end{aligned}$$

Таким образом, совместная априорная плотность распределения параметров \mathbf{a}_t и b_t имеет вид:

$$\begin{aligned}
\psi(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1}) &= N(\mathbf{a}_t | \sqrt{1-d}\mathbf{a}_{t-1}, d\mathbf{I})N(b_t | b_{t-1}, d') = \\
&= \frac{1}{d^{n/2}(2\pi)^{n/2}} \exp\left(-\frac{1}{2d}(\mathbf{a}_t - \sqrt{1-d}\mathbf{a}_{t-1})^T(\mathbf{a}_{t-1} - \sqrt{1-d}\mathbf{a}_{t-1})\right) \frac{1}{2d'} \exp\left(-\frac{1}{2d'}(b_t - b_{t-1})^2\right) \quad (5)
\end{aligned}$$

3. Существующие методы обучения распознаванию образов в условиях нестационарной генеральной совокупности

Также как и при решении задач в условиях стационарной генеральной совокупности существующие методы распознавания для нестационарного случая можно разделить на те, которые используют одиночный классификатор, и те, что построены как ансамбль (композиция) классификаторов.

Большинство алгоритмов первой группы тем или иным образом эксплуатируют технологию временного окна, которая состоит в отборе из всех поступивших обучающих данных некоторого числа объектов, полученных непосредственно перед моментом контроля, предполагая, что именно в них содержится наиболее релевантная информация. Количество выбираемых объектов составляет длину окна, которая может быть постоянной или переменной (Рисунок 1).

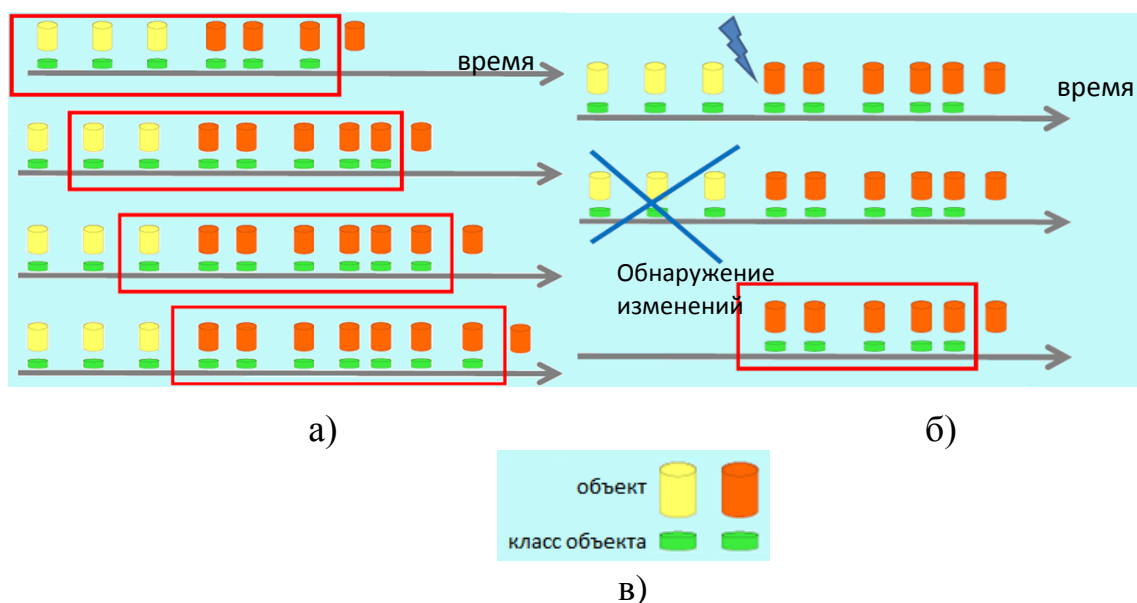


Рисунок 1 – Технология временного окна: (а) - постоянной длины; (б) – переменной длины, где объекты обозначены как (в)

Постоянная длина окна задается пользователем на основе каких-то априорных соображений о скорости изменений в среде, в случае же переменной длины для ее определения используется обнаружение

происходящих в генеральной совокупности изменений. Обнаружение может быть построено с помощью статистических методов [10]или теории информации [11].Наиболее известными методами этой группы являются алгоритмы семейства FLORA [12], ADWIN [13], а также TMF [14].

Отдельно стоит отметить алгоритмы обучения с одиночным классификатором, в которых временное окно не используется. В таких методах в процессе получения объектов обучающей выборки для обучения классификатора отбираются экстремальные объекты, содержащие наиболее важную и свежую информацию [15] или, наоборот, обнаруживаются и удаляются те объекты, которые содержат более не актуальную информацию, а обучение нового классификатора производится на оставшихся данных [16].

Более сложными и, как правило, более точными являются методы, основанные на композиции алгоритмов. В этом случае на множестве всех объектов обучающей выборки строится семейство решающих правил, которые объединяются затем на основе голосования или взвешенного голосования составляющих. Здесь можно выделить два подхода:

1) При поступлении новых данных выбросить из ансамбля худший по качеству классификатор, дополнив композицию новым, обученным на поступивших данных. Одним из первых алгоритмов этой группы был SEA [17]

2) При поступлении новых данных переобучить на них все классификаторы композиции.

Представителем этого подхода является AccuracyWeightedEnsemble (AWE) [18]

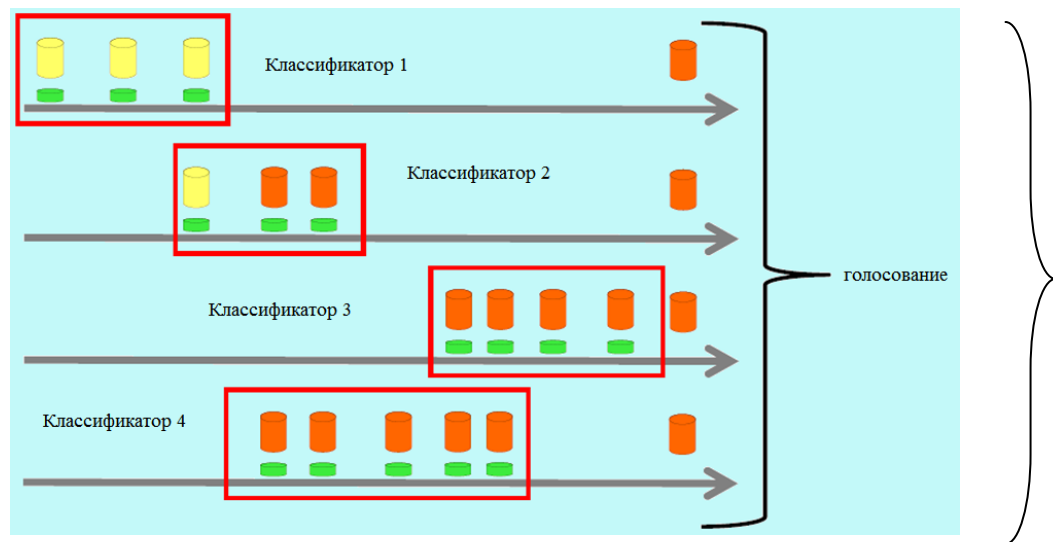


Рисунок 2 – Композиция классификаторов

В качестве составляющих композиции могут выбираться методы из первой группы, так построен, например ADWIN Bagging [19]. Как следует из названия, он использует алгоритм адаптивного окна ADWIN. Также компонентами ансамбля могут быть решающие деревья (в частности, большой популярностью пользуется их специальный вид - деревья Хефдинга), примером может служить EnsembleCombiningRestrictedHoeffdingTrees [20].

По сравнению со стационарным случаем, ситуация нестационарной генеральной совокупности изучена достаточно слабо, что объясняется относительной новизной рассматриваемой темы. В данной области отсутствует четкая система определений, имеют место разночтения в значениях терминов у различных авторов. Существующие методы являются в большей степени эвристическими и предназначены для решения одной конкретной задачи или узкой группы задач.

На первый взгляд это объясняется отсутствием острой необходимости в таких методах из-за небольшого количества задач именно для нестационарной генеральной совокупности, а также тем, что имеются методы распознавания для стационарного решающего правила, которые позволяют получить приемлемые результаты даже в случае небольшого его смещения. Однако, необходимо заметить, что ни один объект реального мира не

является неизменным во времени, отсюда следует, что учитывать поступающие объекты необходимо в неразрывной связи с тем моментом времени, в который они были получены. Таким образом, любая задача распознавания является задачей в нестационарной генеральной совокупности и должна рассматриваться именно так.

Тогда очевидно, что использование для их решения алгоритмов, учитывающих смещение решающего правила, будет гораздо более эффективно, нежели алгоритмов, этого не учитывающих. Особенно четко это видно в тех ситуациях, в которых пренебрежение смещением решающего правила может привести к недопустимо большой ошибке при распознавании.

4. Обобщение метода опорных векторов для нестационарной генеральной совокупности

4.1. Метод опорных векторов в классической задаче обучения распознаванию образов

Естественно в качестве оценки модели генеральной совокупности попытаться выбрать параметры линейной дискриминантной функции так, чтобы она принимала положительные значения для объектов первого класса и отрицательные для другого: $y_j(\mathbf{a}^T \mathbf{x}_j + b) > 0$. Если такие значения параметров найдутся (обучающая совокупность линейно разделима, т.е. выпуклые оболочки множеств точек двух классов не пересекаются), то существует множество значений, удовлетворяющих этим неравенствам. Выдвинутая В.Н. Вапником концепция оптимальной разделяющей гиперплоскости [4] заключается в выборе таких значений параметров, при которых «зазор» между минимальным положительным и максимальным отрицательным значениями дискриминантной функции был бы максимальным: $\varepsilon \rightarrow \max$ при условиях $\mathbf{a}^T \mathbf{x}_j + b \geq \varepsilon > 0$ для объектов первого класса $y_j = 1$ и $\mathbf{a}^T \mathbf{x}_j + b \leq -\varepsilon < 0$ для объектов второго класса $y_j = -1$, или, что эквивалентно, $y_j(\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0$ для всех объектов обучающей совокупности $j = 1, \dots, N$.

Здесь евклидова норма направляющего вектора $\|\mathbf{a}\| = (\mathbf{a}^T \mathbf{a})^{1/2}$ условна, ее всегда можно принять равной единице, изменив величину параметра положения b . Буквальное понимание концепции оптимальной разделяющей гиперплоскости приводит к следующему критерию обучения:

$$\begin{cases} \varepsilon \rightarrow \max, \\ y_j(\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0, \quad j = 1, \dots, N, \\ \mathbf{a}^T \mathbf{a} = 1. \end{cases} \quad (6)$$

Обычно этот критерий записывают в более удобной форме, которая получается из (6) делением обеих частей всех неравенств на ε

$$y_j \left[\left((1/\varepsilon) \mathbf{a} \right)^T \mathbf{x}_j + (1/\varepsilon) b \right] \geq 1$$

и введением обозначений $(1/\varepsilon) \mathbf{a} = \mathbf{a}^*$ и $(1/\varepsilon) b = b^*$. В этих обозначениях $\mathbf{a}^{*T} \mathbf{a}^* = (1/\varepsilon^2) \mathbf{a}^T \mathbf{a}$, т.е. $\mathbf{a}^{*T} \mathbf{a}^* = (1/\varepsilon^2)$ при выполнении ограничения $\mathbf{a}^T \mathbf{a} = 1$, требование максимизации «зазора» $\varepsilon \rightarrow \max$ эквивалентно требованию минимизации квадрата нормы направляющего вектора $\mathbf{a}^{*T} \mathbf{a}^* \rightarrow \min$. Сохраняя прежние обозначения \mathbf{a} и b для масштабированных параметров, мы получим эквивалентную запись критерия обучения (6):

$$\begin{cases} \mathbf{a}^T \mathbf{a} \rightarrow \min, \\ y_j (\mathbf{a}^T \mathbf{x}_j + b) \geq \varepsilon > 0, \quad j = 1, \dots, N. \end{cases}$$

Если выпуклые оболочки двух классов в обучающей совокупности пересекаются, В.Н. Вапник предложил использовать следующий критерий, вообще говоря, эвристический, допускающий насильственное смещение точек, «мешающих» линейному разделению:

$$\begin{cases} J(\mathbf{a}, b, \delta_j, j = 1, \dots, N) = \mathbf{a}^T \mathbf{a} + C \sum_{j=1}^N \delta_j \rightarrow \min, \\ y_j (\mathbf{a}^T \mathbf{x}_j + b) \geq 1 - \delta_j, \quad \delta_j \geq 0, \quad j = 1, \dots, N. \end{cases} \quad (7)$$

Очевидно, что это задача квадратичного программирования, т.е. задача минимизации квадратичной функции при ограничениях и виде линейных неравенств. Заметим, что каждое ограничение соответствует одному объекту обучающей совокупности, требуя, чтобы этот объект, по-возможности, правильно классифицировался дискриминантной функцией.

Именно этот критерий обучения распознаванию двух классов объектов получил огромную популярность в литературе под названием метода опорных векторов. Такое название определяется тем обстоятельством, что в точке минимума целевой функции активными оказываются лишь небольшое число ограничений-неравенств, соответствующих векторам признаков лишь некоторых объектов, которые полностью определяют оптимальные значения параметров разделяющей гиперплоскости. Эти векторы называются

опорными векторами в данной обучающей совокупности, давая название методу в целом.

Задачу квадратичного программирования (7) обычно записывают и решают в двойственной форме, в которой роль опорных векторов становится очевидной.

4.2. Вероятностное обоснование метода опорных векторов

Пусть в пространстве признаков R^n объективно определена разделяющая гиперплоскость $\mathbf{a}^T \mathbf{x} + b = 0$, неизвестная наблюдателю. В качестве модели генеральной совокупности будем рассматривать два параметрических семейства плотностей распределения вероятностей, $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$ и $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$, связанных с двумя классами объектов $y(\omega) = 1$ и $y(\omega) = -1$, и сконцентрированные преимущественно по разные стороны гиперплоскости. Совместную плотность распределения конечного множества векторов признаков объектов известных классов в составе обучающей совокупности (3) будем понимать как плотность распределения выборки независимых реализаций этих двух распределений:

$$\Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b) = \prod_{j=1}^N \varphi_{y_j}(\mathbf{x}_j | \mathbf{a}, b) = \left(\prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b) \right) \left(\prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right).$$

Пусть, далее, выбрана априорная плотность совместного распределения вероятностей $\psi(\mathbf{a}, b)$ для параметров распределений $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$ и $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$. Тогда апостериорная плотность распределения параметров \mathbf{a} и b относительно обучающей совокупности определяется формулой Байеса:

$$p(\mathbf{a}, b | y_j, j = 1, \dots, N) = \frac{\psi(\mathbf{a}, b) \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b)}{\int \psi(\mathbf{a}', b') \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}', b') d\mathbf{a}' db'}. \quad (8)$$

Поскольку знаменатель не зависит от целевых переменных, то достаточно рассматривать только числитель:

$$p(\mathbf{a}, b | y_j, j = 1, \dots, N) \propto \psi(\mathbf{a}, b) \Phi(\mathbf{x}_j, j = 1, \dots, N | y_j, j = 1, \dots, N, \mathbf{a}, b) =$$

$$\psi(\mathbf{a}, b) \left(\prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b) \right) \left(\prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right).$$

Принцип максимизации плотности апостериорного распределения в пространстве параметров модели генеральной совокупности приводит к байесовскому правилу обучения:

$$\left(\hat{\mathbf{a}}, \hat{b} | (\mathbf{x}_j, y_j), j = 1, \dots, N \right) = \arg \max_{\mathbf{a}, b} p(\mathbf{a}, b | y_j, j = 1, \dots, N) =$$

$$\arg \max_{\mathbf{a}, b} \left[\log \psi(\mathbf{a}, b) + \sum_{j: y_j=1} \log \varphi_1(\mathbf{x}_j | \mathbf{a}, b) + \sum_{j: y_j=-1} \log \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b) \right]. \quad (9)$$

В работе [21] предложена следующая вероятностная модель генеральной совокупности. Рассмотрим несобственные плотности распределения $\varphi_1(\mathbf{x}(\omega) | \mathbf{a}, b)$ и $\varphi_{-1}(\mathbf{x}(\omega) | \mathbf{a}, b)$, определяемые выражениями

$$\varphi_1(\mathbf{x} | \mathbf{a}, b) = \begin{cases} 1, & \mathbf{a}^T \mathbf{x} + b > 1, \\ \exp \left[-c \left(1 - (\mathbf{a}^T \mathbf{x} + b) \right) \right], & \mathbf{a}^T \mathbf{x} + b < 1, \end{cases}$$

$$\varphi_{-1}(\mathbf{x} | \mathbf{a}, b) = \begin{cases} 1, & \mathbf{a}^T \mathbf{x} + b < -1, \\ \exp \left[-c \left(1 + (\mathbf{a}^T \mathbf{x} + b) \right) \right], & \mathbf{a}^T \mathbf{x} + b > -1. \end{cases} \quad (10)$$

Наглядное представление этой пары несобственных плотностей распределения приведено на Рисунке 3. Очевидно, что распределения полностью соответствуют качественной модели генеральной совокупности(2), являясь равномерными при достаточном удалении от разделяющей гиперплоскости $\mathbf{a}^T \mathbf{x} + b > 1$ либо $\mathbf{a}^T \mathbf{x} + b < -1$, а также вдоль любой прямой, параллельной ей. Конкретизация качественной модели (2) заключается в том, что несобственные плотности приняты экспоненциально уменьшающимися при $\mathbf{a}^T \mathbf{x} + b < 1$ и, соответственно, $\mathbf{a}^T \mathbf{x} + b > -1$.

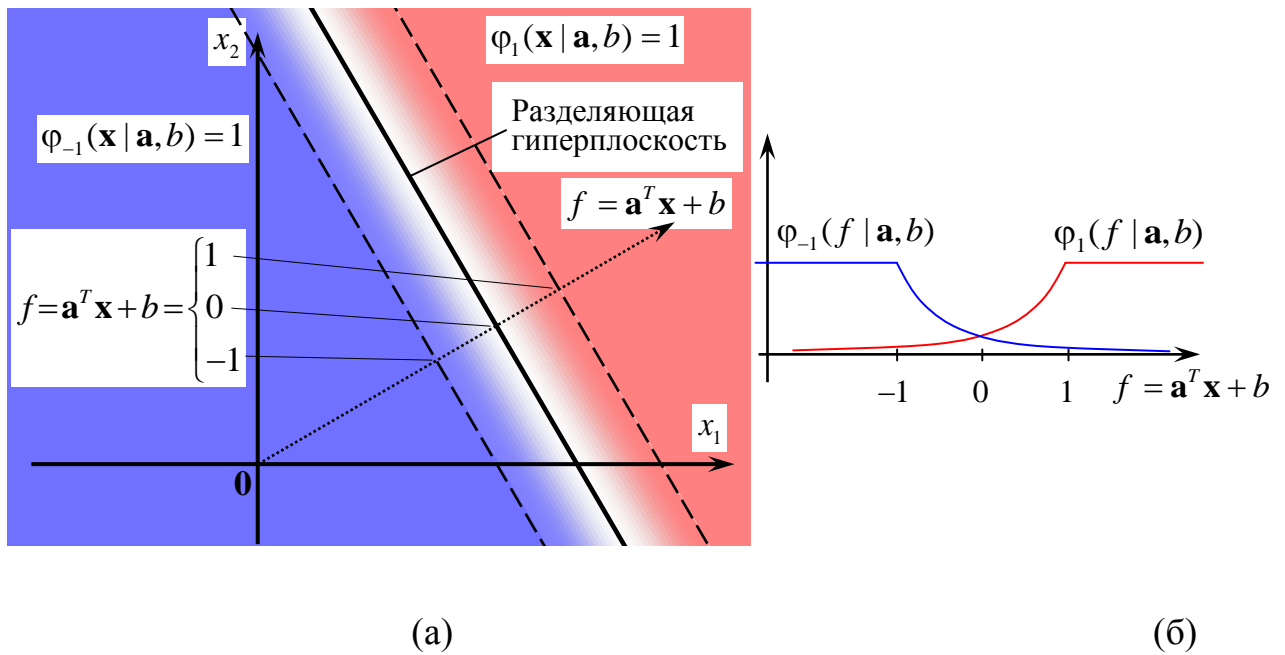


Рисунок 3 - Яркое представление несобственных плотностей распределения двух классов в двумерном пространстве признаков (а) и их значения вдоль направляющего вектора разделяющей гиперплоскости (б).

Направляющий вектор разделяющей гиперплоскости \mathbf{a} будем считать априори нормально распределенным с независимыми компонентами, имеющими одинаковые априорные дисперсии σ^2 . Что же касается параметра положения разделяющей гиперплоскости b , то будем считать, что отсутствуют какие-либо априорные предположения о его значении, что выражается равномерным несобственным распределением, равным единице на всей числовой оси. Тогда совместное априорное распределение параметров разделяющей гиперплоскости будет выражаться плотностью

$$\psi(\mathbf{a}, b | \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2} \mathbf{a}^T \mathbf{a}\right).$$

Очевидно, что это несобственная плотность, поскольку ее интеграл по переменной b не существует.

Для несобственных плотностей распределения векторов признаков объектов двух классов и априорного несобственного распределения

параметров разделяющей гиперплоскости байесовский критерий обучения(9) примет вид[21]

$$\begin{aligned} & \left(\hat{\mathbf{a}}, \hat{b} \mid (\mathbf{x}_j, y_j), j = 1, \dots, N \right) = \\ & \arg \min_{\mathbf{a}, b} \left[\frac{1}{2\sigma^2} \mathbf{a}^T \mathbf{a} + c \sum_{\substack{j: y_j=1 \\ \mathbf{a}^T \mathbf{x}_j + b < 1}} (1 - (\mathbf{a}^T \mathbf{x}_j + b)) + c \sum_{\substack{j: y_j=-1 \\ \mathbf{a}^T \mathbf{x}_j + b > -1}} (1 + (\mathbf{a}^T \mathbf{x}_j + b)) \right]. \end{aligned} \quad (11)$$

При выборе коэффициента штрафа на сумму смещений точек обучающей совокупности $C = 2\sigma^2 c$ этот критерий полностью идентичен классическому критерию обучения методом опорных векторов.

4.3. Распространение метода опорных векторов на случай нестационарной генеральной совокупности

Вспомним описанное в пункте 2.2 предположение, заключающееся в описании свойства нестационарности генеральной совокупности при помощи разделяющей гиперплоскости, со временем изменяющей свое положение. Следовательно, ее направляющий вектор и параметр положения рассматриваются как функции времени \mathbf{a}_t и b_t : $f_t(\mathbf{x}(\omega)) = \mathbf{a}_t^T \mathbf{x} + b_t$ и модельное представление генеральной совокупности в виде несобственных распределений двух классов в пространстве признаков(10) будет иметь вид:

$$\varphi(\mathbf{x} \mid \mathbf{a}_t, b_t, y; c) = \begin{cases} const, & yz(\mathbf{a}_t, \mathbf{x}) \geq 1, \\ e^{-c(1-yz(\mathbf{a}_t, \mathbf{x}))}, & yz(\mathbf{a}_t, \mathbf{x}) < 1, \end{cases}$$

где $z(\mathbf{x}, \mathbf{a}_t) = \mathbf{a}_t^T \mathbf{x} + b = 0$.

Тогда совместная условная априорная вероятность принадлежности объектов в составе обучающей совокупности выражается как произведение:

$$\Phi(\mathbf{Y} \mid \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2) = \prod_{j=1}^{N_T} \varphi(\mathbf{x} \mid \mathbf{a}_t, b_t, y; c)$$

Пусть совместная априорная плотность распределения параметров за весь период времени определяется как произведение плотностей для отдельных отсчетов:

$$\Psi(\mathbf{a}_t, b_t, t = 1, \dots, T) = \prod_{t=1}^T \psi_t(\mathbf{a}_t, b_t \mid \mathbf{a}_{t-1}, b_{t-1}),$$

где условная априорная плотность $\psi(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1})$ определяется согласно (5). Будем предполагать, что в нулевой момент времени априорное распределение параметров разделяющей гиперплоскости является несобственным и имеет вид:

$$\psi_0(\mathbf{a}_0, b_0) \propto \psi_0(\mathbf{a}_0) = N(\mathbf{a}_0 | \mathbf{0}, \mathbf{I}).$$

Применение принципа максимизации апостериорной вероятности для оценивания последовательности параметров $(\mathbf{a}_t, b_t)_{t=1}^T$ при принятых предположениях об априорных плотностях параметров направляющего вектора и плотностях классов приводит к следующему оптимизационному критерию

$$J(\mathbf{a}_t, b_t, \delta_t, t=0, \dots, T) = \mathbf{a}_0^T \mathbf{a}_0 + \frac{1}{d} \sum_{t=1}^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1})^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1}) + \frac{1}{d'} \sum_{t=1}^T (b_t - b_{t-1})^2 + \sum_{t=1}^T \sum_{j=1}^{N_t} \delta_{j,t} \rightarrow \min_{[\mathbf{a}_t, b_t]_{j=1}^T} \quad (12)$$

$$y_{j,t}(\mathbf{a}_t^T \mathbf{x}_{j,t} + b_t) \geq 1 - \delta_{j,t}, \delta_{j,t} \geq 0, \\ j = 1, \dots, N_t, t = 1, \dots, T$$

который мы перепишем в общем виде:

$$J(\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=0}^T [(\mathbf{z}'_t - \mathbf{z}^0_t)^T \mathbf{Q}_t (\mathbf{z}'_t - \mathbf{z}^0_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t] + \sum_{t=1}^T (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \rightarrow \min_{\mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_T}, \\ \mathbf{g}_j^T \cdot \mathbf{z}'_t + z''_j - 1 \geq 0, j = (N_{t-1} + 1), \dots, N_t, t = 1, \dots, T, \\ z''_j \geq 0, j = (N_{t-1} + 1), \dots, N_t, t = 1, \dots, T. \quad (13)$$

где

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{z}'_t \\ \mathbf{z}''_t \end{bmatrix}, \mathbf{z}'_t = \begin{bmatrix} \mathbf{a}_t \\ b_t \end{bmatrix}, \mathbf{z}''_t = [\delta_j, j = (N_{t-1} + 1), \dots, N_t], \mathbf{g}_j = \begin{bmatrix} y_j \mathbf{x}_j \\ y_j \end{bmatrix}, j = (N_{t-1} + 1), \dots, N_t, \mathbf{e}_t = [1]_{N_{t-1}+1}^{N_t}, \\ \mathbf{Q}_t^0 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}, \mathbf{U}_t = \begin{bmatrix} \frac{1}{d} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{d} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{d'} \end{bmatrix}, \mathbf{A}_t = \begin{bmatrix} \sqrt{1-d} & 0 & \dots & 0 & 0 \\ 0 & \sqrt{1-d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \sqrt{1-d} & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \\ t = 1, \dots, T$$

4.4. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода опорных векторов

В работе [22] был предложен алгоритм, основанный на идее использования для оптимизации получившегося парно-сепарабельного критерия (12) общего принципа динамического программирования.

Введем новые обозначения

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{z}'_t & z''_t \end{bmatrix}^T, \quad \mathbf{z}'_t = \begin{bmatrix} \mathbf{a}_t^T & b_t \end{bmatrix}^T, \quad \mathbf{z}''_t = \begin{bmatrix} \delta_j \end{bmatrix}_{j=1}^{N_t}, \quad \zeta_t(\mathbf{z}'_t) = (\mathbf{z}'_t - \mathbf{z}_t^0)^T \mathbf{Q}_t^0 (\mathbf{z}'_t - \mathbf{z}_t^0),$$

$$\chi_t(\mathbf{z}''_t) = \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t, \quad \mathbf{e}_t = [1]_1^{N_t}, \quad t = 1, \dots, T, \quad \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) = (\mathbf{z}'_t - \mathbf{A}_j \mathbf{z}'_{t-1})^T \mathbf{U}_j (\mathbf{z}'_t - \mathbf{A}_j \mathbf{z}'_{t-1})$$

и перепишем критерий (12) в более удобном виде

$$J(\mathbf{z}_0, \dots, \mathbf{z}_T) = \sum_{t=0}^T \zeta_t(\mathbf{z}'_t) + \sum_{t=0}^T \chi(\mathbf{z}''_t) + \sum_{t=1}^T \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) \rightarrow \min, \quad \mathbf{z}_t \in Z_t$$

где области допустимых значений переменных определяются условиями

$$\{\mathbf{z} \in \square^{n+2} : \mathbf{g}_j^T \cdot \mathbf{z}'_t + z''_j - 1 \geq 0, \quad j = (N_{t-1} + 1), \dots, N_t, \quad t = 0, \dots, T, \quad \mathbf{z}''_t \geq 0\}.$$

Метод динамического программирования основан на понятии последовательности функций Беллмана

$$\tilde{J}_t(\mathbf{z}_t) = \min_{\mathbf{z}_0, \dots, \mathbf{z}_{t-1}} J_t([\mathbf{z}_s]_{s=1}^t), \quad [\mathbf{z}_s \in Z_s]_{s=0}^{t-1},$$

связанных с частичными критериями

$$J_t(\mathbf{z}_0, \dots, \mathbf{z}_t) = \sum_{s=0}^t \zeta_s(\mathbf{z}'_s) + \sum_{s=0}^t \chi(\mathbf{z}_s) + \sum_{s=1}^t \gamma_s(\mathbf{z}'_{s-1}, \mathbf{z}'_s),$$

имеющими такую же структуру, как и полная целевая функция, но определенными на множестве переменных $Z_t = (\mathbf{z}_s, s = 0, \dots, t)$. Более подробно о методе динамического программирования и функциях Беллмана будет сказано в следующем разделе. Для получения фильтрационных оценок параметров разделяющей гиперплоскости мы будем использовать фундаментальное свойство функции Беллмана

$$\tilde{J}_t(\mathbf{z}_t) = \zeta_t(\mathbf{z}'_t) + \chi(\mathbf{z}''_t) + \min_{\mathbf{z}'_{t-1}, \mathbf{z}''_{t-1}} \left[\gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1}) \right] \quad (14)$$

которое называется прямым рекуррентным соотношением. Процедура начинается со значения первой функции Беллмана $\tilde{J}_0(\mathbf{z}_0) = \zeta_0(\mathbf{z}_0') + \chi(\mathbf{z}_0'')$. Затем, функции Беллмана рекуррентно пересчитываются для последующих отсчетов, в соответствии с прямым рекуррентным соотношением. При этом минимум функции Беллмана на каждом шаге определяет фильтрационное значение параметров оптимальной разделяющей гиперплоскости

$$\hat{\mathbf{z}}_t = \arg \min_{\mathbf{z}_t} \hat{J}_t(\mathbf{z}_t), \mathbf{z}_t \in Z_t \quad (15)$$

4.5. Численная реализация процедуры динамического программирования для оценивания параметров оптимальной разделяющей гиперплоскости

Предполагается, что существует подходящая компактная форма представления функций Беллмана, позволяющая хранить эти функции в памяти. В нашем случае предыдущая функция Беллмана в (14) является кусочно-квадратичной. При этом дробность кусочной квадратичности функций Беллмана будет нарастать на каждом шаге процедуры динамического программирования, что означает невозможность подобрать для них адекватное конечно-параметрическое семейство. Это делает невозможным численную реализацию процедуры динамического программирования.

Предложенная в [23] идея приближенной реализации процедуры динамического программирования заключается в замене функции

$$F_t(\mathbf{z}_t') = \min_{\mathbf{z}_{t-1} \in Z_{t-1}} [\gamma_t(\mathbf{z}_{t-1}', \mathbf{z}_t') + \tilde{J}_{t-1}(\mathbf{z}_{t-1})]$$

подходящей квадратичной функцией $\hat{F}_t(\mathbf{z}_t') = \hat{c}_t + (\mathbf{z}_t' - \hat{\mathbf{z}}_t)^T \hat{\mathbf{Q}}_t (\mathbf{z}_t' - \hat{\mathbf{z}}_t)$

Тогда квадратичными будут и следующие аппроксимации функций Беллмана и станет возможной численная реализация процедуры динамического программирования.

Таким образом, квадратичная аппроксимация очередной функций Беллмана сводится к подбору подходящих значений параметров $(\hat{c}_t, \hat{\mathbf{z}}_t, \hat{\mathbf{Q}}_t)$

квадратичной функции $\hat{F}_t(\mathbf{z}'_t)$, которые обеспечивали бы сохранение основных особенностей, вообще говоря, неквадратичной функции и, следовательно, исходной функции Беллмана. Такими особенностями являются, положение точек минимума функции $\hat{\mathbf{z}}_t = \arg \min F_t(\mathbf{z}'_t)$, значения в точках минимума $\hat{c}_t = \min F_t(\mathbf{z}'_t)$, а также матрица вторых производных в точке минимума $\hat{\mathbf{Q}}_t = \nabla^2 F_t(\mathbf{z}'_t) \Big|_{\arg \min F_t(\mathbf{z}'_t)}$. Представляется предпочтительным передать эти параметры в точности, выбрав

$$\hat{c}_t = \min F_t(\mathbf{z}'_t), \hat{\mathbf{z}}_t = \arg \min F_t(\mathbf{z}'_t), \hat{\mathbf{Q}}_t = \nabla^2 F_t(\mathbf{z}'_t) \Big|_{\arg \min F_t(\mathbf{z}'_t)} \quad (16)$$

Теорема 2[22]. Пусть предыдущая функция Беллмана $\bar{J}_{t-1}(\mathbf{z}_{t-1})$ является квадратичной по \mathbf{z}'_{t-1} и линейной по \mathbf{z}''_{t-1} , функция связи $\gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t)$ является квадратичной.

$$\begin{aligned} \bar{J}_{t-1}(\mathbf{z}_{t-1}) &= \tilde{c}_{t-1} + (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} \\ \gamma_t(\mathbf{z}'_{t-1}, \mathbf{z}'_t) &= (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \end{aligned}$$

Тогда параметры $(\hat{c}_t, \hat{\mathbf{z}}_t, \hat{\mathbf{Q}}_t)$ квадратичной аппроксимации $\hat{F}_t(\mathbf{z}'_t)$ функции $F_t(\mathbf{z}'_t)$, удовлетворяющие условиям (16), имеют вид

$$\begin{aligned} \hat{\mathbf{z}}_t &= \dot{\mathbf{z}}'_{t-1} \\ \dot{\mathbf{z}}_{t-1} &= \begin{bmatrix} \dot{\mathbf{z}}'_{t-1} \\ \dot{\mathbf{z}}''_{t-1} \end{bmatrix} \end{aligned} \quad (17)$$

$$\hat{c}_t = \tilde{c}_{t-1} + \mathbf{C} \mathbf{e}_{t-1}^T \dot{\mathbf{z}}''_{t-1} + (\dot{\mathbf{z}}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\dot{\mathbf{z}}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) \quad (18)$$

$$\hat{\mathbf{Q}}_t = \mathbf{U}_t^T \mathbf{A}_t (\mathbf{H}_{t-1}^{11})^T \bar{\mathbf{Q}}_{t-1} \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t + (\mathbf{A}_t \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t - \mathbf{I})^T \mathbf{U}_t (\mathbf{A}_t \mathbf{H}_{t-1}^{11} \mathbf{A}_t^T \mathbf{U}_t - \mathbf{I}) \quad (19)$$

где

$$(\dot{\mathbf{z}}_{t-1}) = \arg \min_{\substack{\mathbf{z}_{t-1}, \\ \mathbf{g}_j^T \mathbf{z}_{t-1} + z_j'' - 1 \geq 0, \\ j=(N_{t-2}+1), \dots, N_{t-1}, \\ \mathbf{z}_{t-1} \geq 0}} \left[(\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} \right] \quad (20)$$

Матрица \mathbf{H}_{t-1}^{11} является левым верхним блоком $(n \times n)$ матрицы

$$\mathbf{H}_{t-1} = \begin{pmatrix} \mathbf{H}_{t-1}^{11} & \mathbf{H}_{t-1}^{12} & \mathbf{H}_{t-1}^{13} & \mathbf{H}_{t-1}^{14} & \mathbf{H}_{t-1}^{15} \\ \mathbf{H}_{t-1}^{21} & \mathbf{H}_{t-1}^{22} & \mathbf{H}_{t-1}^{23} & \mathbf{H}_{t-1}^{24} & \mathbf{H}_{t-1}^{25} \\ \mathbf{H}_{t-1}^{31} & \mathbf{H}_{t-1}^{32} & \mathbf{H}_{t-1}^{33} & \mathbf{H}_{t-1}^{34} & \mathbf{H}_{t-1}^{35} \\ \mathbf{H}_{t-1}^{41} & \mathbf{H}_{t-1}^{42} & \mathbf{H}_{t-1}^{43} & \mathbf{H}_{t-1}^{44} & \mathbf{H}_{t-1}^{45} \\ \mathbf{H}_{t-1}^{51} & \mathbf{H}_{t-1}^{52} & \mathbf{H}_{t-1}^{53} & \mathbf{H}_{t-1}^{54} & \mathbf{H}_{t-1}^{55} \end{pmatrix} = \begin{pmatrix} \bar{\mathbf{Q}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t & -\mathbf{G}'^T & \mathbf{0} & \mathbf{0} & -\mathbf{G}''^T \\ \mathbf{G}' & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{V}'^T & \mathbf{V}''^T \\ \mathbf{0} & \mathbf{0} & \mathbf{V}' & \mathbf{0} & \mathbf{0} \\ \mathbf{G}'' & \mathbf{0} & \mathbf{V}'' & \mathbf{0} & \mathbf{0} \end{pmatrix}^{-1},$$

где $\mathbf{G}_{t-1} = [\mathbf{g}_j^T]_{j=N_{t-2}+1}^{N_{t-1}}$.

Доказательство.

Введем обозначения

$$(\dot{\mathbf{z}}_{t-1}, \hat{\mathbf{z}}_t) = \arg \min_{\substack{\mathbf{z}_{t-1}, \mathbf{z}_t \\ \mathbf{g}_j^T \mathbf{z}'_{t-1} + \mathbf{z}_j'' - 1 \geq 0, j=(N_{t-2}+1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[\tilde{c}_{t-1} + (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} + \right. \\ \left. + (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right]$$

Очевидно, что компонента $\hat{\mathbf{z}}_t$ этого составного вектора является точкой минимума функции $F_t(\mathbf{z}'_t)$. Эта задача оптимизации распадается на две независимых задачи оптимизации по каждой из переменных $\dot{\mathbf{z}}_{t-1}$ и $\hat{\mathbf{z}}_t$ в отдельности,

$$\begin{aligned} \dot{\mathbf{z}}_{t-1} &= \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{g}_j^T \mathbf{z}'_{t-1} + \mathbf{z}_j'' - 1 \geq 0, j=(N_{t-2}+1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[(\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \mathbf{C} \mathbf{e}_{t-1}^T \mathbf{z}''_{t-1} \right] \\ \dot{\mathbf{z}}_{t-1} &= \begin{bmatrix} \dot{\mathbf{z}}'_{t-1} \\ \mathbf{z}''_{t-1} \end{bmatrix} \\ \hat{\mathbf{z}}_t &= \arg \min_{\mathbf{z}_t} \left[(\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right] \end{aligned} \quad (1)$$

Задача определения переменной $\dot{\mathbf{z}}_{t-1}$ в (1) является задачей общего квадратичного программирования небольшой размерности $(n+1+(N_{t-1}-N_{t-2})) \times (n+1+(N_{t-1}-N_{t-2}))$ с ограничениями, решить которые можно довольно легко из-за небольшой размерности с помощью метода внутренней точки или симплекс метода. После нахождения переменной $\dot{\mathbf{z}}_{t-1}$ в (1) оставшаяся переменная $\hat{\mathbf{z}}_t$ определяется как минимум квадратичной формы $(\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})$, т.е.

$$\hat{\mathbf{z}}_t = \arg \min_{\mathbf{z}_t} \left[(\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) \right] = \mathbf{A}_t \mathbf{z}'_{t-1}$$

Параметр \hat{c}_t есть не что иное, как значение минимума

$$\hat{c}_t = \tilde{c}_{t-1} + C\mathbf{e}_{t-1}^T \dot{\mathbf{z}}_{t-1}'' + (\dot{\mathbf{z}}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\dot{\mathbf{z}}_{t-1}' - \tilde{\mathbf{z}}_{t-1})$$

Остается выбрать матрицу $\hat{\mathbf{Q}}_t$, определяющую скорость квадратичного возрастания значения функции $F_t(\mathbf{z}_t')$ при отклонения от точки минимума $\hat{\mathbf{z}}_t$ в разных направлениях в пространстве векторного аргумента $\mathbf{z}_t' \in \mathbb{R}^{n+1}$.

$$F_t(\mathbf{z}_t') = \min_{\substack{\mathbf{z}_{t-1}', \mathbf{z}_{t-1}'' \\ \mathbf{g}_j^T \mathbf{z}_{t-1}' + z_j'' - 1 \geq 0, j = (N_{t-2} + 1), \dots, N_{t-1} \\ \mathbf{z}_{t-1}'' \geq 0}} \left[\tilde{c}_{t-1} + (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1}) + C\mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' + \right. \\ \left. + (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}')^T \mathbf{U}_t (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}') \right] \quad (2)$$

Варьирование \mathbf{z}_t' будет, вообще говоря, приводить к изменению множеств активных ограничений, однако при малых отклонениях состав этих ограничений будет, скорее всего, оставаться тем же, что и в точках минимума. Тогда при малых отклонениях функция $\hat{F}_t(\mathbf{z}_t')$ будет определяться как результат минимизации при тех ограничениях типа равенств, которые были активны в точке минимума как ограничения типа неравенств.

$$F_t(\mathbf{z}_t') = \min_{\mathbf{z}_{t-1}', \mathbf{z}_{t-1}''} \left[\tilde{c}_{t-1} + (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1}' - \tilde{\mathbf{z}}_{t-1}) + C\mathbf{e}_{t-1}^T \mathbf{z}_{t-1}'' + (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}')^T \mathbf{U}_t (\mathbf{z}_t' - \mathbf{A}_t \mathbf{z}_{t-1}') \right] \\ \left\{ \begin{array}{l} \mathbf{g}_j^T \mathbf{z}_{t-1}' - 1 = 0, j = (N_{t-2} + 1), \dots, k_1 \\ z_j'' = 0, j = (N_{t-2} + 1), \dots, k_2 \\ \mathbf{g}_j^T \mathbf{z}_{t-1}' + z_j'' - 1 = 0, j = (k_2 + 1), \dots, N_{t-1} \end{array} \right.$$

$$\left\{ \begin{array}{l} \mathbf{G}'\mathbf{z}'_{t-1} - \mathbf{e}' = \mathbf{0}, \mathbf{G}' = [\mathbf{g}'_j^T]_{j=N_{t-2}+1}^{k_1}, \mathbf{e}' = [1]_{j=N_{t-2}+1}^{k_1} \\ \\ \mathbf{V}'\mathbf{z}''_{t-1} = \mathbf{0}, \mathbf{V}'((k_2 - N_{t-2}) \times (N_{t-1} - N_{t-2})) = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 & \dots & 0 \end{bmatrix} \\ \\ \mathbf{G}''\mathbf{z}'_{t-1} + \mathbf{V}''\mathbf{z}''_{t-1} - \mathbf{e}'' = \mathbf{0}, \mathbf{G}'' = [\mathbf{g}''_j^T]_{j=k_2+1}^{N_{t-1}}, \mathbf{V}''((N_{t-1} - k_2) \times (N_{t-1} - N_{t-2})) = \begin{bmatrix} 0 & \dots & 0 & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & 0 & 1 & \dots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & 0 & \dots & 1 \end{bmatrix}, \\ \\ \mathbf{e}'' = [1]_{j=k_2+1}^{N_{t-1}} \end{array} \right.$$

$$L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1}) = \frac{1}{2} \tilde{c}_{t-1} + \frac{1}{2} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \tilde{\mathbf{Q}}_{t-1} (\mathbf{z}'_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \frac{1}{2} C \mathbf{e}''^T \mathbf{V}'' \mathbf{z}''_{t-1} + \\ + \frac{1}{2} (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1})^T \mathbf{U}_t (\mathbf{z}'_t - \mathbf{A}_t \mathbf{z}'_{t-1}) - \boldsymbol{\mu}_{t-1}^T (\mathbf{G}' \mathbf{z}'_{t-1} - \mathbf{e}') - \mathbf{v}_{t-1}^T (\mathbf{G}'' \mathbf{z}'_{t-1} + \mathbf{V}'' \mathbf{z}''_{t-1} - \mathbf{e}'') - \boldsymbol{\eta}_{t-1}^T \mathbf{V}' \mathbf{z}''_{t-1},$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{z}'_{t-1}} = \bar{\mathbf{Q}}_{t-1} \mathbf{z}'_{t-1} - \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} - \mathbf{A}_t^T \mathbf{U}_t \mathbf{z}'_t + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t \mathbf{z}'_{t-1} - \mathbf{G}'^T \boldsymbol{\mu}_{t-1} - \mathbf{G}''^T \mathbf{v}_{t-1} = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{z}''_{t-1}} = \frac{1}{2} C \mathbf{V}''^T \mathbf{e}'' - \mathbf{V}''^T \mathbf{v}_{t-1} - \mathbf{V}'^T \boldsymbol{\eta}_{t-1} = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \boldsymbol{\mu}_{t-1}} = -\mathbf{G}' \mathbf{z}'_{t-1} + \mathbf{e}' = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \mathbf{v}_{t-1}} = -\mathbf{G}'' \mathbf{z}'_{t-1} - \mathbf{V}'' \mathbf{z}''_{t-1} + \mathbf{e}'' = 0$$

$$\frac{L(\mathbf{z}_{t-1}, \boldsymbol{\mu}_{t-1}, \mathbf{v}_{t-1}, \boldsymbol{\eta}_{t-1})}{\partial \boldsymbol{\eta}_{t-1}} = -\mathbf{V}' \mathbf{z}''_{t-1} = 0$$

Полученные условия приводят к системе линейных уравнений:

$$\left\{ \begin{array}{l} (\bar{\mathbf{Q}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{A}_t) \mathbf{z}'_{t-1} - \mathbf{G}'^T \boldsymbol{\mu}_{t-1} - \mathbf{G}''^T \mathbf{v}_{t-1} = \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{A}_t^T \mathbf{U}_t \mathbf{z}'_t \\ \mathbf{G}' \mathbf{z}'_{t-1} = \mathbf{e}' \\ \mathbf{V}''^T \mathbf{v}_{t-1} + \mathbf{V}'^T \boldsymbol{\eta}_{t-1} = \frac{1}{2} C \mathbf{V}''^T \mathbf{e}'' \\ \mathbf{V}' \mathbf{z}''_{t-1} = 0 \\ \mathbf{G}'' \mathbf{z}'_{t-1} + \mathbf{V}'' \mathbf{z}''_{t-1} = \mathbf{e}'' \end{array} \right.$$

Или в обобщенно матричной форме:

откуда следует

Введем обозначение

Размерность блока \mathbf{H}_{t-1}^{11} $(n+1) \times (n+1)$. Тогда

линейную функцию в

$$F(s) = \min_{s_0, s_1, \dots, s_{n-1}} \sum_{i=0}^{n-1} \lambda_i \left[s_{i+1} + (s_i^* - s_{i+1})^* \Phi_i(s_i^* - s_{i+1}) + (s_i^* - \lambda_i s_{i+1})^* \Psi_i(s_i^* - \lambda_i s_{i+1}) \right]$$

(2), получим:

Таким образом, матрицу при квадратичном члене в $\hat{F}_l(\mathbf{z}'_l)$ следует брать равной:

Алгоритм заключается в последовательном вычислении параметров

$$\bar{\mathbf{Q}}_t = \hat{\mathbf{Q}}_t + \mathbf{Q}_t^0, \tilde{\mathbf{z}}_t = (\bar{\mathbf{Q}}_t)^{-1} (\mathbf{Q}_t^0 \mathbf{z}_t^0 + \hat{\mathbf{Q}}_t \hat{\mathbf{z}}_t), \tilde{c}_t = c'_t$$

квадратичных аналогов функций Беллмана $\tilde{J}_j(\mathbf{x}_j)$

$$\tilde{J}_t(\mathbf{z}_t) \equiv \zeta_t(\mathbf{z}'_t) + \chi(\mathbf{z}''_t) + \hat{F}_t(\mathbf{z}'_t) = \tilde{c}_t + (\mathbf{z}'_t - \tilde{\mathbf{z}}_t)^T \bar{\mathbf{Q}}_t (\mathbf{z}'_t - \tilde{\mathbf{z}}_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t$$

в направлении $t = 0, \dots, T$, начиная с исходных присвоений $\tilde{\mathbf{Q}}_0 = \mathbf{Q}_0^0, \tilde{\mathbf{z}}_0 = \mathbf{z}_0^0, \tilde{c}_0 = 0$, с использованием формулы (17), связанной с решением на каждом шаге соответствующей задачи квадратичного программирования (20) размерности $(n+2)$, и далее формул (18), (19).

Оптимальное значение векторной переменной в последней вершине $t = T$ получается непосредственно с помощью минимизирования квадратичной функции Беллмана в последней вершине при соответствующих ограничениях

$$\bar{\mathbf{z}}_T = \arg \min_{\substack{\mathbf{z}_T \\ \mathbf{g}_j^T \mathbf{z}'_T + z_j'' - 1 \geq 0, j = (N_{T-1}+1), \dots, N_T \\ \mathbf{z}''_T \geq 0}} \tilde{J}_T(\mathbf{x}_T) = \arg \min_{\substack{\mathbf{z}_T \\ \mathbf{g}_j^T \mathbf{z}'_T + z_j'' - 1 \geq 0, j = (N_{T-1}+1), \dots, N_T \\ \mathbf{z}''_T \geq 0}} \left[\tilde{c}_T + (\mathbf{z}'_T - \tilde{\mathbf{z}}_T)^T \bar{\mathbf{Q}}_T (\mathbf{z}'_T - \tilde{\mathbf{z}}_T) + \mathbf{C} \mathbf{e}_T^T \mathbf{z}''_T \right]$$

Таким образом, мы получим решение задачи фильтрации $\hat{\mathbf{z}}_T$.

Для нахождения задачи интерполяции, в направлении $t = T, \dots, 0$ вычисляются оптимальные значения переменных по обратному рекуррентному соотношению

$$\bar{\mathbf{z}}_t = (\mathbf{z}_{t-1}(\bar{\mathbf{z}}_t)) = \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{z}_{t-1} \in \mathbf{Z}_{t-1}}} \left[\gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \bar{J}_{t-1}(\mathbf{z}_{t-1}) \right] = \arg \min_{\substack{\mathbf{z}_{t-1} \\ \mathbf{z}_{t-1} \in \mathbf{Z}_{t-1}}} \left[(\mathbf{z}'_t - \hat{\mathbf{z}}_t)^T \hat{\mathbf{Q}}_t (\mathbf{z}'_t - \hat{\mathbf{z}}_t) + \mathbf{C} \mathbf{e}_t^T \mathbf{z}''_t \right]$$

где параметры $\hat{\mathbf{Q}}_t$, $\hat{\mathbf{z}}_t$ определяются из соотношений (17)–(19).

Достоинством указанного алгоритма является то, что для построения решающего правила при поступлении новой порции объектов необходимо лишь знание параметров классификатора в данный момент и не требуется хранение обучающего множества. Данный алгоритм является достаточно точным, что было подтверждено экспериментами на реальных данных. Скорость его работы также вполне приемлема для задач распознавания, в которых данные поступают группами объемом до нескольких сотен.

Недостаток метода состоит в том, что возможность численной реализации процедуры динамического программирования основана на

предположении о существовании параметрического семейства, которому на каждом шаге процедуры принадлежат функции Беллмана. В данной задаче такого параметрического семейства не существует, и приходится аппроксимировать неквадратичные функции Беллмана их квадратичными аналогами, что неизбежно приводит к ухудшению качества распознавания. Однако, на больших массивах данных из-за необходимости решения для каждого отсчета оптимизационной задачи (20) время поиска может оказаться неудовлетворительным.

5. Обобщение модели логистической регрессии на случай нестационарной генеральной совокупности

5.1. Модель логистической регрессии в задаче обучения распознаванию образов

Одним из методов восстановления эмпирической зависимости $y(\omega): \Omega \rightarrow Y$ является логистическая регрессия, которая, несмотря на присутствие в названии слова «регрессия», предназначена для решения задач классификации.

Логистическая регрессия основана на следующей модели, описывающей апостериорные вероятности принадлежности объектов классам[24]:

$$\begin{aligned} P(y=1|\mathbf{x}) &= \sigma(z) \\ P(y=-1|\mathbf{x}) &= 1 - \sigma(z), \end{aligned} \quad (21)$$

где $\sigma(z) = \frac{1}{1+e^{-z}}$ – логистическая функция (Рисунок 4), $z = x_1\theta_1 + x_2\theta_2 + \dots + x_n\theta_n$ – линейная модель регрессии. Для вышеописанной задачи двуклассового распознавания с учетом(2) выражения(21) приобретают вид:

$$\begin{aligned} P(y=1|\mathbf{x}) &= \sigma(-(\mathbf{a}^T \mathbf{x} + b)) \\ P(y=-1|\mathbf{x}) &= \sigma(\mathbf{a}^T \mathbf{x} + b) \end{aligned} \quad (22)$$

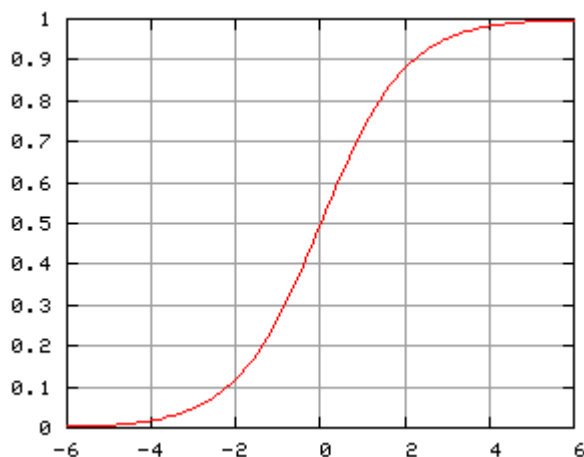


Рисунок 4 - Логистическая функция

Для оценки оптимальных значений параметров \mathbf{a} и b обычно используется метод максимального правдоподобия, в соответствии с

которым для их нахождения требуется оптимизировать следующий критерий:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \prod_{j=1}^N P(y = y_j | \mathbf{x} = \mathbf{x}_j),$$

что эквивалентно:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \sum_{j=1}^N \log P(y = y_j | \mathbf{x} = \mathbf{x}_j). \quad (23)$$

Решение задачи (23) возможно только итерационными методами, такими как градиентный спуск или метод Ньютона, что является недостатком логистической модели. Однако, после определения параметров решающего правила появляется возможность оценивать вероятности принадлежности объектов к каждому из двух классов.

Улучшить обобщающую способность логистической регрессии позволяет применение регуляризации. Идея ее в том, что направляющий вектор гиперплоскости \mathbf{a} и параметр положения b полагаются случайными с некоторой априорной плотностью распределения $p(\mathbf{a}, b)$, тогда применение метода максимизации апостериорной вероятности приводит к следующему оптимизационному критерию:

$$(\hat{\mathbf{a}}, \hat{b}) = \arg \max_{\mathbf{a}, b} \prod_{j=1}^N P(y = y_j | \mathbf{x} = \mathbf{x}_j, \mathbf{a}, b) p(\mathbf{a}, b)$$

5.2. Вероятностное обоснование модели логистической регрессии

Запишем логистические функции $P(y=1|\mathbf{x})$ и $P(y=-1|\mathbf{x})$ (22), описывающие апостериорные вероятности принадлежности объекта к каждому из двух классов, в следующем виде:

$$\begin{aligned} \phi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= P(y=1|\mathbf{x}) = \frac{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)} \\ \phi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= P(y=-1|\mathbf{x}) = \frac{\varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)}, \end{aligned} \quad (24)$$

где

$$\begin{aligned}\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &\propto \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right) \\ \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2) &\propto \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2\right).\end{aligned}\quad (25)$$

Нетрудно убедиться, что функции (24) полностью эквивалентны (22). В самом деле, например, для апостериорной вероятности распределения класса $y = 1$

$$\begin{aligned}\phi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) &= \frac{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2)}{\varphi_1(\mathbf{x} | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x} | \mathbf{a}, b, \sigma^2)} = \\ &= \frac{\exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)}{\exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2\right) + \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)} = \\ &= \frac{1}{1 + \exp\left(-\frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b + 1)^2 + \frac{1}{2\sigma^2}(\mathbf{a}^T \mathbf{x} + b - 1)^2\right)} = \\ &= \frac{1}{1 + \exp\left(-\frac{1}{\sigma^2}(\mathbf{a}^T \mathbf{x} + b)\right)} = \frac{1}{1 + \exp\left(-\frac{2}{\sigma^2}(\mathbf{a}^T \mathbf{x} + b)\right)}\end{aligned}$$

Аналогично для логистической функции противоположного класса $y = -1$.

Тогда совместная условная апостериорная вероятность для всей обучающей совокупности $\{\mathbf{X}, \mathbf{Y}\}$:

$$\begin{aligned}\Phi(Y | X, \mathbf{a}, b, \sigma^2) &= \prod_{j: y_j=1} \phi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) \prod_{j: y_j=-1} \phi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) = \\ &= \frac{\prod_{j: y_j=1} \varphi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) \prod_{j: y_j=-1} \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2)}{\prod_j (\varphi_1(\mathbf{x}_j | \mathbf{a}, b, \sigma^2) + \varphi_{-1}(\mathbf{x}_j | \mathbf{a}, b, \sigma^2))}\end{aligned}$$

Внесем значение индекса класса объекта в квазинормальные распределения (25) и перепишем функции **Ошибка! Закладка не определена.** в более краткой форме:

$$\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) = \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(1 | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(-1 | \mathbf{x}, \mathbf{a}, b, \sigma^2)}, \quad (26)$$

где

$$\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) \propto \exp\left[-\frac{1}{2\sigma^2}\left(1 - y(\mathbf{a}^T \mathbf{x} + b)\right)^2\right] \quad (27)$$

Критерий обучения, построенный по принципу максимизации апостериорной вероятности,

$$\Phi(Y | X, \mathbf{a}, b, \sigma^2) \rightarrow \max_{\mathbf{a}, b},$$

$$\Phi(Y | X, \mathbf{a}, b, \sigma^2) = \prod_{j=1}^N \phi(y_j | \mathbf{x}_j, \mathbf{a}, b, \sigma^2) = \prod_{j=1}^N \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}$$

Максимизация функции $\Phi(Y | X, \mathbf{a}, b, \sigma^2)$ эквивалентна максимизации ее логарифма

$$\log \Phi(Y | X, \mathbf{a}, b, \sigma^2) \rightarrow \max_{\mathbf{a}, b},$$

$$\begin{aligned} \log \Phi(Y | X, \mathbf{a}, b, \sigma^2) &= \log \prod_{j=1}^N \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)} = \\ &= \sum_{j=1}^N \log \frac{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)}{\varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2) + \varphi(y | \mathbf{x}, \mathbf{a}, b, \sigma^2)} \end{aligned}$$

С учетом Ошибка! Закладка не определена. Ошибка! Закладка не определена.

$$\begin{aligned} G(\mathbf{a}, b | \sigma, (y_i, \mathbf{x}_i)_{i=1}^N) &= \sum_{i=1}^N \left(\frac{1}{2\sigma^2} (1 - y_i(\mathbf{a}^T \mathbf{x}_i + b))^2 \right) + \\ &+ \sum_{i=1}^N \log \left[\exp \left(-\frac{1}{2\sigma^2} (1 + \mathbf{a}^T \mathbf{x}_i + b)^2 \right) + \exp \left(-\frac{1}{2\sigma^2} (1 - \mathbf{a}^T \mathbf{x}_i - b)^2 \right) \right] \rightarrow \min_{\mathbf{a}, b} \end{aligned} \quad (28)$$

Можно доказать, что для критерия (28) имеет место

Теорема 3. Если априорные плотности распределения классов имеют вид (26) – (27), то при $N \rightarrow \infty$ и равнонаполненности классов решение оптимизационной задачи (27) сводится к оптимизации критерия:

$$\hat{G}(\mathbf{a}, b | \sigma, (y_i, \mathbf{x}_i)_{i=1}^N) = \sum_{i=1}^N \left(\frac{1}{2\sigma^2} (1 - y_i(\mathbf{a}^T \mathbf{x}_i + b))^2 \right) \rightarrow \min_{\mathbf{a}, b}.$$

5.3. Процедура динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью метода логистической регрессии

В терминах логистической модели представление о нестационарной генеральной совокупности, изложенное в п. 2.2 выражается в виде

предположения, что параметры логистических функций (26), описывающих апостериорные вероятности принадлежности объекта к каждому из двух классов, изменяются во времени:

$$\phi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) = \frac{\varphi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2)}{\varphi(1 | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) + \varphi(-1 | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2)}, \quad (29)$$

где

$$\varphi(y | \mathbf{x}, \mathbf{a}_t, b_t, \sigma^2) \propto \exp \left[-\frac{1}{2\sigma^2} \left(1 - y(\mathbf{a}_t^T \mathbf{x} + b_t) \right)^2 \right]. \quad (30)$$

Совместная условная апостериорная вероятность принадлежности объектов в составе обучающей совокупности выражается как произведение:

$$\Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2) = \prod_{j=1}^{N_t} \phi(y_j | \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2)$$

где согласно(29)

$$\phi(y_j | \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) = \frac{\exp \left[-\frac{1}{2\sigma^2} \left(1 - y_j(\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right]}{\exp \left[-\frac{1}{2\sigma^2} \left(1 - (\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right] + \exp \left[-\frac{1}{2\sigma^2} \left(1 + (\mathbf{a}_t^T \mathbf{x}_j + b_t) \right)^2 \right]},$$

Совместная априорная плотность распределения параметров за весь период времени определяется как произведение плотностей для отдельных отсчетов (5):

$$\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) = \prod_{t=1}^T \psi_t(\mathbf{a}_t, b_t | \mathbf{a}_{t-1}, b_{t-1})$$

Апостериорное распределение параметров разделяющих гиперплоскостей после наблюдения обучающей совокупности определяется формулой Байеса:

$$\begin{aligned} P(\mathbf{a}_t, b_t, t = 0, \dots, T | \mathbf{X}, \mathbf{Y}) &= \frac{\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T)}{\int \dots \int \dots \int \Psi(\mathbf{a}'_t, b'_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}, \mathbf{a}'_t, b'_t, t = 1, \dots, T) db_0 \dots db_T d\mathbf{a}_0 \dots d\mathbf{a}_T} = \\ &= \frac{\Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T)}{F(\mathbf{Y} | \mathbf{X})} \propto \Psi(\mathbf{a}_t, b_t, t = 0, \dots, T) \Phi(\mathbf{Y} | \mathbf{X}; \mathbf{a}_t, b_t, \sigma^2, t = 1, \dots, T) \end{aligned}$$

А обучение будем понимать как вычисление байесовской оценки параметров разделяющих гиперплоскостей:

$$\begin{aligned}
& (\mathbf{a}_t, b_t, t=1, \dots, T \mid \mathbf{X}, \mathbf{Y}, \sigma^2, d, d') = \arg \max \Phi(\mathbf{Y} \mid \mathbf{X}, \mathbf{a}_t, b_t, \sigma^2, t=1, \dots, T) \Psi(\mathbf{a}_t, b_t, t=1, \dots, T) = \\
& = \arg \max \left[\ln \Phi(\mathbf{Y} \mid \mathbf{X}, \mathbf{a}_t, b_t, \sigma^2, t=1, \dots, T) + \ln \Psi(\mathbf{a}_t, b_t, t=1, \dots, T) \right] = \\
& = \arg \max \left[\sum_{t=1}^T \sum_{j=N_{t-1}+1}^{N_t} \ln \varphi(y_j \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) - \sum_{i=1}^N \ln \left[\varphi(1 \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) + \varphi(-1 \mid \mathbf{x}_j, \mathbf{a}_t, b_t, \sigma^2) \right] + \right. \\
& \quad \left. + \sum_{t=1}^T \ln \psi_t(\mathbf{a}_t, b_t \mid \mathbf{a}_{t-1}, b_{t-1}) \right]
\end{aligned}$$

Используя Теорема 3, получим следующую эквивалентную оптимизационную задачу:

$$\begin{aligned}
J(\mathbf{a}_t, b_t, t=1, \dots, T) = & \left[\frac{1}{2\sigma^2} \sum_{t=1}^T \sum_{j=N_{t-1}+1}^{N_t} (1 - y_j(\mathbf{a}_t^T \mathbf{x}_j + b_t))^2 + \right. \\
& \left. + \frac{1}{d} \sum_{t=2}^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1})^T (\mathbf{a}_t - \sqrt{1-d} \mathbf{a}_{t-1}) + \frac{1}{2d'} \sum_{t=2}^T (b_t - b_{t-1})^2 \right] \rightarrow \min_{(\mathbf{a}_t, b_t)_{t=1}^T}
\end{aligned}$$

Именно этот критерий мы будем в дальнейшем использовать, однако удобнее записать его в более общем виде:

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=1}^T (\mathbf{z}_t - \mathbf{z}_t^0)^T \mathbf{Q}_t (\mathbf{z}_t - \mathbf{z}_t^0) + \sum_{t=2}^T (\mathbf{z}_t - \mathbf{A} \mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A} \mathbf{z}_{t-1}) \rightarrow \min_{\mathbf{z}_1, \dots, \mathbf{z}_T}, \quad (31)$$

где введены следующие обозначения:

$$\begin{aligned}
\mathbf{z}_t = & \begin{bmatrix} \mathbf{a}_t \\ b_t \end{bmatrix}; \quad \mathbf{z}_t^0 = (\mathbf{Q}_t^T \mathbf{Q}_t)^{-1} \mathbf{Q}_t^T \sum_{j=N_{t-1}}^{N_t} \mathbf{g}_j; \quad \mathbf{Q}_t = C \sum_{j=N_{t-1}}^{N_t} \mathbf{g}_j \mathbf{g}_j^T; \quad \mathbf{g}_j = \begin{bmatrix} y_j \mathbf{x}_j \\ y_j \end{bmatrix}, \quad t=1, \dots, T \\
\mathbf{U} = & \begin{bmatrix} \frac{1}{d} & 0 & \dots & 0 & 0 \\ 0 & \frac{1}{d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \frac{1}{d} & 0 \\ 0 & 0 & \dots & 0 & \frac{1}{d'} \end{bmatrix}; \quad \mathbf{A} = \begin{bmatrix} \sqrt{1-d} & 0 & \dots & 0 & 0 \\ 0 & \sqrt{1-d} & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \sqrt{1-d} & 0 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}.
\end{aligned}$$

5.4. Численная реализация процедуры динамического программирования для оценивания параметров решающего правила при обучении распознаванию образов в нестационарной генеральной совокупности с помощью модели логистической регрессии

Парно-сепарабельный критерий (31) имеет вид

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = \sum_{t=1}^T \zeta_t(\mathbf{z}_t) + \sum_{t=2}^T \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) \rightarrow \min, \quad \mathbf{z}_t \in Z_t, \quad (32)$$

где области допустимых значений переменных определяются условиями

$$Z_t = \{ \mathbf{z}_t \in \mathbb{R}^{n+1}, t = 1, \dots, T \}$$

Функции $\zeta_t(\mathbf{z}_t), \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t)$ являются квадратичными

$$\begin{aligned}\zeta_t(\mathbf{z}_t) &= (\mathbf{z}_t - \mathbf{z}_t^0)^T \mathbf{Q}_t (\mathbf{z}_t - \mathbf{z}_t^0) \\ \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) &= (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})\end{aligned}$$

Рассмотрим способ решения задачи (32) на основе принципа динамического программирования [25], который хотя и адекватен парно-сепарабельным целевым функциям, но был создан для решения задач оптимизации с дискретными аргументами.

Применительно к задаче(32), центральная идея метода динамического программирования заключается в понятии последовательности функций Беллмана $J_t(\mathbf{z}_t) = \min_{\mathbf{z}_1, \dots, \mathbf{z}_{t-1}} J_t(\mathbf{z}_1, \dots, \mathbf{z}_t), \mathbf{z}_s \in Z_s, s = 1, \dots, t-1$, связанных с частичными критериями

$$J(\mathbf{z}_1, \dots, \mathbf{z}_t) = \sum_{s=1}^t \zeta_s(\mathbf{z}_s) + \sum_{s=2}^t \gamma_s(\mathbf{z}_{s-1}, \mathbf{z}_s),$$

имеющими такую же структуру, как и полная целевая функция(32), но определенными на множестве переменных $Z_t = (\mathbf{z}_s, s = 1, \dots, t)$. Нетрудно заметить, что

$$J_t(\mathbf{z}_1, \dots, \mathbf{z}_t) = \zeta_t(\mathbf{z}_t) + \gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + J_{t-1}(\mathbf{z}_1, \dots, \mathbf{z}_{t-1})$$

При $t=1$, очевидно:

$$J_1(\mathbf{z}_1) = \zeta_1(\mathbf{z}_1), \quad (33)$$

а в последний момент времени $t=T$ функция Беллмана определяется следующим выражением

$$J(\mathbf{z}_1, \dots, \mathbf{z}_T) = J_T(\mathbf{z}_1, \dots, \mathbf{z}_T) \quad (34)$$

Фундаментальное свойство функции Беллмана

$$\begin{aligned}\tilde{J}_t(\mathbf{z}_t) &= \zeta_t(\mathbf{z}_t) + \min_{\mathbf{z}_{t-1}} [\gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1})], \\ t &= 2, \dots, T, \mathbf{z}_{t-1} \in Z_{t-1},\end{aligned} \quad (35)$$

будем называть прямым рекуррентным соотношением[25], а функцию

$$(\mathbf{z}_{t-1}) = (\mathbf{z}_{t-1}(\mathbf{z}_t)) = \arg \min_{\mathbf{z}_{t-1}} [\gamma_t(\mathbf{z}_{t-1}, \mathbf{z}_t) + \tilde{J}_{t-1}(\mathbf{z}_{t-1})], \mathbf{z}_{t-1} \in Z_{t-1} \quad (36)$$

будем называть обратным рекуррентным соотношением.

В основе процедуры оптимизации лежит предположение, что существуют, во-первых, достаточно эффективный способ решения частных задач оптимизации, входящих в (35)и, во-вторых, подходящая компактная форма представления функций Беллмана или обратных рекуррентных соотношений (36)позволяющая хранить эти функции в памяти.

Процедура оптимизации пробегает дважды по всем отсчетам вперед от начала к концу сигнала $t = 1, \dots, T$ и затем назад от конца к началу $t = T, \dots, 1$.

Процедура начинается со значения первой функции Беллмана $t = 1$, $\tilde{J}_1(\mathbf{z}_1) = \zeta_1(\mathbf{z}_1)$ далее осуществляется пересчет функций Беллмана для последующих отсчетов $t = 1, \dots, T$ в соответствии с прямым рекуррентным соотношением(35). При этом функции Беллмана или обратные рекуррентные соотношения (36)должны быть запомнены для всех $t = 1, \dots, T$. Функция Беллмана для последней переменной $\tilde{J}_T(\mathbf{z}_T)$, полученная на последнем шаге прямого хода алгоритма, непосредственно указывает ее оптимальное значение:

$$\hat{\mathbf{z}}_T = \arg \min_{\mathbf{z}_T} \hat{J}_T(\mathbf{z}_T), \mathbf{z}_T \in Z_T$$

Таким образом, в силу (34)мы получим, что найденное $\hat{\mathbf{z}}_T$ является решением задачи фильтрации.

На обратном ходе, по мере того, как процедура последовательно проходит от конца сигнала к началу $t = T, \dots, 2$, уже найденное оптимальное значение векторной переменной в каждой очередной вершине позволяет, в свою очередь, определить оптимальное значение переменной в непосредственно предшествующей вершине. Последовательное вычисление оптимальных значений переменных обеспечивается обратными рекуррентными соотношениями, найденными и сохраненными в процессе

прямого хода алгоритма либо непосредственно в виде (36) либо косвенно определяемыми функциями Беллмана(35), запоминаемыми вместо них:

$$\hat{\mathbf{z}}_{t-1} = \tilde{\mathbf{z}}_{t-1}(\hat{\mathbf{z}}_t)$$

Найденные оптимальные значения $\hat{\mathbf{z}}_1, \dots, \hat{\mathbf{z}}_T$ являются решением задачи интерполяции.

Поскольку в оптимизационной задаче(31)все составляющие парно-сепарабельной целевой функции являются квадратичными, то и функции Беллмана также являются квадратичными, их параметры можно легко пересчитывать, и, соответственно, может быть реализована процедура динамического программирования.

Следовательно, функции Беллмана будут определяться следующим выражением

$$\tilde{J}_t(\mathbf{z}_t) = (\mathbf{z}_t - \tilde{\mathbf{z}}_t)^T \bar{\mathbf{Q}}_t (\mathbf{z}_t - \tilde{\mathbf{z}}_t) + \tilde{c}_t, \quad (37)$$

параметры которого для момента $t=1$ будут принимать тривиальные значения:

$$\tilde{\mathbf{z}}_1 = \mathbf{z}_1^0; \bar{\mathbf{Q}}_1 = \mathbf{Q}; \tilde{c}_1 = 0.$$

Пусть

$$F_t(\mathbf{z}_t) = \min_{\mathbf{z}_{t-1}} \left[(\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1})^T \mathbf{U} (\mathbf{z}_t - \mathbf{A}\mathbf{z}_{t-1}) + (\mathbf{z}_{t-1} - \tilde{\mathbf{z}}_{t-1})^T \bar{\mathbf{Q}}_{t-1} (\mathbf{z}_{t-1} - \tilde{\mathbf{z}}_{t-1}) + \tilde{c}_{t-1} \right],$$

соответственно

$$\frac{\partial F_t(\mathbf{z}_t)}{\partial \mathbf{z}_{t-1}} = -2\mathbf{A}\mathbf{U}\mathbf{z}_t + 2\mathbf{A}\mathbf{U}\mathbf{A}\mathbf{z}_{t-1} + 2\bar{\mathbf{Q}}_{t-1}\mathbf{z}_{t-1} - 2\bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1} = (\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}).$$

Согласно необходимому условию экстремума получаем:

$$\begin{aligned} F_t(\mathbf{z}_t) = & \left[\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right]^T \mathbf{U} \left[\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right] + \\ & + \left[(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) - \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \left[(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) - \tilde{\mathbf{z}}_{t-1} \right] + \tilde{c}_{t-1} \end{aligned}$$

где выделим две составляющие:

$$\begin{aligned} & \left(\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right)^T \mathbf{U} \left(\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right) = \\ & = \left[\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right]^T \mathbf{U} \left[\mathbf{z}_t - \mathbf{A}(\mathbf{A}\mathbf{U}\mathbf{A} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{A}\mathbf{U}\mathbf{z}_t + \bar{\mathbf{Q}}_{t-1}\tilde{\mathbf{z}}_{t-1}) \right] \quad (38) \end{aligned}$$

и

$$\begin{aligned} & \left(\mathbf{z}_t - \ddot{\mathbf{z}}_t \right)^T \ddot{\mathbf{Q}}_t \left(\mathbf{z}_t - \ddot{\mathbf{z}}_t \right) = \\ & = \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \left(\mathbf{AUz}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right) - \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \left(\mathbf{AUz}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right) - \tilde{\mathbf{z}}_{t-1} \right] \end{aligned} \quad (39)$$

Преобразуем **Ошибка! Закладка не определена.**

$$\begin{aligned} & \left(\mathbf{z}_t - \dot{\mathbf{z}}_t \right)^T \dot{\mathbf{Q}}_t \left(\mathbf{z}_t - \dot{\mathbf{z}}_t \right) = \\ & = \left[\left(\mathbf{I} - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right) \mathbf{z}_t - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right]^T \mathbf{U} \cdot \\ & \quad \cdot \left[\left(\mathbf{I} - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right) \mathbf{z}_t - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \right] \end{aligned}$$

откуда:

$$\begin{aligned} \dot{\mathbf{Q}}_t &= \left[\mathbf{I} - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \mathbf{U} \left[\mathbf{I} - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right] \\ \dot{\mathbf{z}}_t &= \dot{\mathbf{Q}}_t^{-1} \cdot \left(\mathbf{I} - \mathbf{A} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right)^T \mathbf{UA} \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1}. \end{aligned}$$

Из **Ошибка! Закладка не определена.** следует:

$$\begin{aligned} & \left(\mathbf{z}_t - \ddot{\mathbf{z}}_t \right)^T \ddot{\mathbf{Q}}_t \left(\mathbf{z}_t - \ddot{\mathbf{z}}_t \right) = \\ & = \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AUz}_t - \left(\mathbf{I} - \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \right]^T \bar{\mathbf{Q}}_{t-1} \cdot \\ & \quad \cdot \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AUz}_t - \left(\mathbf{I} - \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \right] \end{aligned}$$

откуда в свою очередь

$$\begin{aligned} \ddot{\mathbf{Q}}_t &= \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right] \\ \ddot{\mathbf{z}}_t &= \ddot{\mathbf{Q}}_t^{-1} \cdot \left[\left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left(\mathbf{I} - \left(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1} \right)^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} \end{aligned}$$

Подставляя **Ошибка! Закладка не определена.** с учетом **Ошибка!**

Закладка не определена. определяем параметры функций Беллмана (37) на каждом шаге

$$\begin{aligned}
\bar{\mathbf{Q}}_t &= \mathbf{Q}_t + \dot{\mathbf{Q}}_t + \ddot{\mathbf{Q}}_t = \\
&= \left[\mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \mathbf{U} \left[\mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \\
&\quad + \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \mathbf{Q}_t = \\
&= \mathbf{U} - 2\mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUU} + \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} + \\
&\quad + \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right]^T \bar{\mathbf{Q}}_{t-1} \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right] + \mathbf{Q}_t = \\
&= \mathbf{U} - 2\mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AUU} + \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1}) (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} + \mathbf{Q}_t = \\
&= \mathbf{U} - \mathbf{AUA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t = \bar{\mathbf{Q}}_{t-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t
\end{aligned}$$

и

$$\begin{aligned}
\tilde{\mathbf{z}}_t &= \bar{\mathbf{Q}}_t^{-1} \left(\mathbf{Q}_t \mathbf{z}_t^0 + \dot{\mathbf{Q}}_t \tilde{\mathbf{z}}_t + \ddot{\mathbf{Q}}_t \tilde{\mathbf{z}}_t \right) = \bar{\mathbf{Q}}_t^{-1} \left[\left(\mathbf{I} - \mathbf{A}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right)^T \mathbf{UA}(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \right. \\
&\quad \left. + \left((\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \right)^T \bar{\mathbf{Q}}_{t-1} \left(\mathbf{I} - (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \bar{\mathbf{Q}}_{t-1} \right) \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right] = \\
&= \bar{\mathbf{Q}}_t^{-1} \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} \cdot \left(2\mathbf{I} - (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1}) \right) + \mathbf{Q}_t \mathbf{z}_t^0 \right] = \\
&= \bar{\mathbf{Q}}_t^{-1} \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right]
\end{aligned}$$

Таким образом, окончательно

$$\begin{aligned}
\bar{\mathbf{Q}}_t &= \bar{\mathbf{Q}}_{t-1} (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{U} + \mathbf{Q}_t \\
\tilde{\mathbf{z}}_t &= \bar{\mathbf{Q}}_t^{-1} \left[(\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} \mathbf{AU} \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1} + \mathbf{Q}_t \mathbf{z}_t^0 \right]
\end{aligned}$$

Значение целевой переменной в последний момент времени $\tilde{\mathbf{z}}_T$ непосредственно представляет решение задачи фильтрации. Задача интерполяции решается путем рекуррентного пересчета при условии сохранения в памяти значений $\bar{\mathbf{Q}}_{t-1}$, $t = T, \dots, 1$

$$\hat{\mathbf{z}}_{t-1} = (\mathbf{AUA} + \bar{\mathbf{Q}}_{t-1})^{-1} (\mathbf{AU} \hat{\mathbf{z}}_t + \bar{\mathbf{Q}}_{t-1} \tilde{\mathbf{z}}_{t-1})$$

6. Экспериментальное исследование

6.1. Экспериментальное исследование на модельных данных

Для исследования работы метода нами было создано искусственное множество данных, образованное двумя нормальными распределениями. Для получения свойства нестационарности центры этих распределений поворачиваются относительно начала координат. В начальный момент времени дисперсии распределений одинаковы и равны 1, математические ожидания – 3.5 и 6.5, соответственно. Создаваемые объекты описываются двумя признаками и индексом класса $\{-1;1\}$. Обучающее множество составили признаковые описания объектов 50-и моментов времени, по 100 объектов в каждом. Контрольная выборка содержит 2000 объектов, соответствующих 51-у отсчету. В Таблице 1 содержится процент ошибочной классификации объектов каждого класса.

Таблица 1 – Экспериментальные результаты:

модельные данные

| Значения параметров d и d' | Ошибка классификации объектов класса -1, % | Ошибка классификации объектов класса +1, % |
|-------------------------------------|--|--|
| $d \rightarrow 0; d' \rightarrow 0$ | $\rightarrow 0$ | $\rightarrow 100$ |
| $d=1; d'=1$ | 1.0092 | 3.1257 |
| $d=10^{-8}; d'=10^{-8}$ | 1,0801 | 2,846 |

Первые две строки таблицы описывают экстремальные случаи:

- $d \rightarrow 0; d' \rightarrow 0$ – большое значение штрафа в критерии не позволяет адаптироваться к происходящим в генеральной совокупности изменениям;
- $d=1; d'=1$ – модель сразу «забывает» информацию, содержащуюся в ранее полученных данных (поскольку данные соответствуют идеальному случаю для выбранной модели ошибка невелика).

6.2. Экспериментальное исследование на реальных данных

Для следующего эксперимента мы выбрали коллекцию данных, собранных с почтовых фильтров, и описывающих электронные письма[26].

В этом множестве содержатся записи о 4601 электронном сообщении, каждое из которых описывается 58 признаками. Значения признаков, характеризующих объекты-письма, являются непрерывными и показывают частоту встречаемости отдельных элементов (слов или символов) в тексте письма или длину непрерывной последовательности прописных букв. Кроме этого, о каждом письме известно, является ли оно рекламным («спам») или же нет, что описывается меткой 1 или 0 соответственно. Процентное соотношение объектов в классах: спам – 1813 (39.4%), не-спам – 2788 (60.6%).

Предварительно была проведена замена меток класса 0 на -1, данные были стандартизованы. Обучающая выборка составлена из 3600 объектов с сохранением исходного соотношения классов. Предполагалось, что каждый момент времени подавалось по 400 объектов. Ошибка классификации вычислялась как процент ошибочно классифицированных объектов к размеру контрольной выборки.

Полученные на контрольном множестве результаты сравнивались с результатами некоторых алгоритмов для распознавания при смещении решающего правила из программного пакета Massive Online Analysis (MOA) [27]:

- OzaBagASHT [19] – метод bagging с адаптивными деревьями Хевдинга; после достижения максимального размера дерево строится заново, начиная с корня. Была проведена серия экспериментов при различных значениях параметра, определяющего максимальное количество листьев в дереве, результаты представлены в Таблице 2.

Таблица 2 – Величина ошибки классификации
при настройке OzaBagASHT

| -s | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 | 28 | 30 |
|--------|-------|-------|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Ошибка | 39,46 | 30,57 | 30,1 | 31,77 | 28,87 | 30,87 | 25,77 | 30,17 | 30,17 | 22,28 | 31,47 | 32,47 | 31,17 |

- OzaBagAdwin [**Ошибка! Закладка не определена.**]- метод agging с обнаружением изменений в данных по алгоритму ADWIN [13]; в качестве классификатора были выбраны решающие деревья с адаптивным байесовским правилом. Была проведена серия экспериментов при различных значениях параметра, определяющего количество классификаторов в наборе, результаты представлены в Таблице 3.

Таблица 3 – Величина ошибки классификации
при настройке OzaBagAdwin

| -s | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Ошибка | 22,278 | 23,278 | 22,877 | 23,776 | 26,773 | 22,977 | 20,879 | 22,078 | 22,677 | 22,278 | 24,177 |

- SingleClassifierDrift[28] – одиночный классификатор; использовалось обнаружение изменений по методу EDDM, в качестве классификатора было выбрано решающее дерево с адаптивным байесовским правилом.
- AdaHoeffdingOptionTree – адаптивное дерево Хевдинга с дополнительными (option)узлами; максимальное количество таких узлов – 50, в ходе испытаний не выявлено влияния этого параметра на величину ошибки.
- LimAttClassifier [20] – ансамбль ограниченных деревьев Хевдинга, каждое из которых строится на своем наборе признаков;

результат получается объединением предсказанных каждым деревом вероятностей классов с использованием сигмовидного персептрона. При обнаружении изменений использовался подход bagging (позволило увеличить точность классификации). Количество признаков в наборе для каждого классификатора равно 2 (при 4 признаках в наборе результат не изменялся).

- Для определения значений параметров представленного нами метода было проведено несколько пробных экспериментов, в результате: $d=10^{-8}$; $d'=10^{-8}$, $C=1$.

В таблице 4 приведены лучшие из полученных результатов классификации для каждого метода.

Таблица 4 – Экспериментальные данные:
база данных электронных писем

| Алгоритм | Ошибка классификации, % |
|------------------------|-------------------------|
| OzaBagASHT | 22,278 |
| OzaBagAdwin | 20,879 |
| SingleClassifierDrift | 39.361 |
| AdaHoeffdingOptionTree | 23.876 |
| LimAttClassifier | 29,271 |
| Предложенный метод | 14,785 |

Как видно, на реальных данных разработанный алгоритм показал хорошие результаты распознавания по сравнению с другими методами, что доказывает его практическую применимость в реальных задачах при условии большого объема поступающих данных для обучения. Кроме того, время, затраченное на обучение алгоритма, оказалось ниже, чем для методов из программного пакета MOA.

Заключение

В данной работе рассмотрена задача обучения распознаванию образов в нестационарной генеральной совокупности. Описание генеральной совокупности построено на модели логистической регрессии. Свойство нестационарности понимается как разделяющая гиперплоскость, параметры которой изменяются во времени. В представленной постановке задачи обучения эти параметры описываются как марковские случайные процессы. Для оценивания параметров применяется байесовский подход к классификации. Нахождение их оптимальных значений в каждый момент времени осуществляется на основании процедуры динамического программирования.

Построенный алгоритм обучения распознаванию образов в нестационарной генеральной совокупности обладает линейной вычислительной сложностью относительно длины обучающей совокупности. Исследование метода на искусственных данных подтвердило его приспособляемость к происходящим в генеральной совокупности изменениям. Сравнение с методами для задач со смещением концепта из состава программного пакета Massive Online Analysis (MOA) на данных, описывающих электронные письма, показало приемлемую вычислительную эффективность предложенного метода.

Список использованных источников

1. Айзерман М.А., Браверман Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. М.: Наука, 1970, 384 с.
2. Журавлев Ю.И., Никифоров В.В. Алгоритмы распознавания, основанные на вычислении оценок. Кибернетика, 1971, № 3.
3. Вапник В.Н. Восстановление зависимостей по эмпирическим данным. М.: Наука, 1979, 447 с.
4. Vapnik V. Statistical Learning Theory. John-Wiley&Sons, Inc. 1998.
5. Wiener N. Extrapolation, Interpolation, and Smoothing of Stationary Random Time Series with Engineering Applications. Technology Press of MIT, John Wiley&Sons, 1949, 163 p.
6. Шавловский М.Б., Красоткина О.В, Моттль В.В. Задача обучения распознаванию образов в нестационарной генеральной совокупности. Математические методы распознавания образов-13, 2007,- С. 226-230
7. Де Гроот М. Оптимальные статистические решения. М., Мир, 1974, 196 с.
8. Красоткина О. В., Моттль В. В., Турков П. А., Байесовский подход к задаче обучения распознаванию образов в нестационарной генеральной совокупности// Интеллектуализация обработки информации: 8-я международная конференция. Республика Кипр, г.Пафос, 17-24 октября 2010 г.: Сборник докладов. – М.: МАКС Пресс, 2010. – 556с. с.379-382.
9. Шавловский М.Б. Задача обучения распознаванию образов в нестационарной генеральной совокупности. Выпускная квалификационная работа магистра.- МФТИ. 2009
- 10 J. Gama, P. Medas, G. Castillo, and P. Rodrigues, “Learning with drift detection,” in Advances in Artificial Intelligence (Lecture Notes in Computer Science), vol. 3171. New York: Springer-Verlag, 2004, pp. 286–295.
- 11 P. Vorburger and A. Bernstein, “Entropy-based concept shift detection,” in Proc. 6th Int. Conf. Data Min., 2006, pp. 1113–1118.
- 12 Widmer, G. and Kubat, M.; Learning in the presence of concept drift and hidden contexts. Machine Learning 23 (1996) 69-101.
- 13 Albert Bifet and Ricard Gavaldà. Learning from time-changing data with adaptive windowing. In SIAM International Conference on Data Mining, 2007.
- 14 Salganicoff M., Tolerating concept and sampling shift in lazy learning using prediction error context switching, AI Review, Special Issue on Lazy Learning, 11 (1-5), 1997, 133-155.
15. Maloof, M. A. and Michalski, R. S.; Incremental learning with partial instance memory. Artificial Intelligence 154 (2004) 95-126.
16. Black, M. and Hickey, R.; Learning classification rules for telecom customer call data under concept drift. Soft Computing - A Fusion of Foundations, Methodologies and Applications 8 (2003) 102-108.
17. Street, W.N., Kim, Y.: A streaming ensemble algorithm (SEA) for large-scale classification. In: KDD, pp. 377–382. ACM Press (2001)

- 18 Wang, H., Fan, W., Yu, P.S., Han, J.: Mining concept-drifting data streams using ensemble classifiers. In: Getoor, L., Senator, T.E., Domingos, P., Faloutsos, C. (eds.) KDD, pp. 226–235. ACM Press (2003)
19. Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Richard Kirkby, Ricard Gavaldà. New ensemble methods for evolving data streams. In 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009
20. Albert Bifet, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer Accurate Ensembles for Data Streams: Combining Restricted Hoeffding Trees using Stacking. In Journal of Machine Learning Research — Proceedings Track 13, 225-240 (2010)
- 21 . Татарчук А.И., Сулимова В.В., Моттль В.В., Уиндридж Д. Метод релевантных потенциальных функций для селективного комбинирования разнородной информации при обучении распознаванию образов на основе байесовского подхода. // Всероссийская конференция ММРО-14.М.: МАКС Пресс, 2009.С. 188–191.
- 22 .
Турков П.А. Разработка алгоритма обучения распознаванию образов в нестационарной генеральной совокупности данных. Выпускная квалификационная работа бакалавра.- ТулГУ. 2010.
- 23 . Красоткина О.В. Алгоритмы оценивания моделей нестационарных сигналов при наличии ограничений. Диссертация на соискание ученой степени к.ф.-м.н., М.: Тульский Государственный Университет - 2003.
24. С. М. Bishop Pattern Recognition and Machine Learning (Information Science and Statistics) // Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2006.
25. Беллман Р., Калаба Р., Динамическое программирование и современная теория управления. – М.: Наука, 1969, - 118.
- 26 . <http://archive.ics.uci.edu/ml/datasets/Spambase>, UCI Repository, Spambase Data Set
27. A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, MOA: Massive Online Analysis <http://sourceforge.net/projects/moa-datastream/>. Journal of Machine Learning Research (JMLR), 2010.
28. Manuel Baena-Garcia, Jose del Campo-Avila, Raul Fidalgo, Albert Bifet, Ricard Gavaldà, and Rafael Morales-Bueno. Early drift detection method. In Fourth International Workshop on Knowledge Discovery from Data Streams, 2006.