Ю.И.Журавлев, В.В.Рязанов, О.В.Сенько

РАСПОЗНАВАНИЕ

Математические методы. Программная система. Практические применения.

ИЗДАТЕЛЬСТВО ФАЗИС

МОСКВА 2005

Введение

В различных областях человеческой деятельности (экономике, финансах, медицине, бизнесе, геологии, химии, и др.) повседневно возникает необходимость решения задач анализа, прогноза и диагностики, выявления скрытых зависимостей и поддержки принятия оптимальных решений. Вследствие бурного роста объема информации, развития технологий ее сбора, хранения и организации в базах и хранилищах данных (в том числе интернет-технологий), точные методы анализа информации и моделирования исследуемых объектов зачастую отстают от потребностей реальной жизни. Здесь требуются универсальные и надежные подходы, пригодные для обработки информации из различных областей, в том числе для решения проблем, которые могут возникнуть в ближайшем будущем. В качестве подобного базиса могут быть использованы технологии и подходы математической теории распознавания и классификации /19, 25, 26/.

Действительно, данные подходы в качестве исходной информации используют лишь наборы описаний-наблюдений объектов, предметов, ситуаций или процессов (выборки прецедентов), при этом каждое отдельное наблюдение-прецедент записывается в виде вектора значений отдельных его свойств-признаков. Выборки признаковых описаний являются простейшими стандартизованными представлениями первичных исходных данных, которые возникают в различных предметных областях в процессе сбора однотипной информации, и которые могут быть использованы для решения следующих задач:

- распознавание (классификация, диагностика) ситуаций, явлений, объектов или процессов с обоснованием решений;
- прогнозирование ситуаций, явлений, процессов или состояний по выборкам динамических данных;
- кластерный анализ и исследование структуры данных;
- выявление существенных признаков и нахождение простейших описаний;
- нахождение эмпирических закономерностей различного вида;
- построение аналитических описаний множеств (классов) объектов;
- нахождение нестандартных или критических случаев;
- формирование эталонных описаний образов.

Первые работы в области теории распознавания и классификации по прецедентам появились в 30-х годах прошлого столетия и были связаны с байесовской теорией принятия решений (работы Неймана, Пирсона /74/), применением разделяющих функций

к задаче классификации (Фишер /63/), решением вопросов проверки гипотез (Вальд /85/). В 50-х годах появились первые нейросетевые модели распознавания (перцептрон Розенблата /48/), связанные с успехами в моделировании головного мозга. К концу 60-х годов уже были разработаны и детально исследованы различные подходы для решения задач распознавания в рамках статистических, перцептронных моделей, и моделей с разделяющими функциями. Итоги данных и последующих исследований были представлены в ряде монографий /1, 2, 8, 11, 25, 30, 31, 33, 41, 45, 48, 55, 57, 58, 64, 73, 75/. Большой вклад в развитие теории распознавания и классификации внесли советские и, в последующем, российские ученые: Айзерман, Браверман, Розоноэр (метод потенциальных функций /2/), Вапник, Червоненкис (статистическая теория распознавания, метод «обобщенный портрет» /11/), Мазуров (метод комитетов /42, 43, 45/), Ивахненко (метод группового учета аргументов /33/), Загоруйко (алгоритмы таксономии и анализа знаний /30, 31/), Лбов (логические методы распознавания и поиска зависимостей /41/). Интенсивные исследования проводились с конца 60-х годов в ВЦ АН СССР (в настоящее время ВЦ им А.А.Дородницына РАН). Еще в начале 60-х академиком РАН Журавлевым был предложен тестовый алгоритм распознавания – логический метод эффективного решения задач распознавания при малом числе обучающих прецедентов /15/. В дальнейшем на базе этого алгоритма Журавлевым был построен новый класс распознающих процедур – алгоритмы вычисления оценок /27/, а затем введена и исследована алгебраическая теория распознавания /26, 28/. В этом направлении фундаментальные результаты получили также чл.корр. РАН Рудаков (общая теория проблемно-ориентированного алгебраического синтеза корректных алгоритмов /49/, чл.корр. РАН Матросов (статистическое обоснование алгебраического подхода /44/), Рязанов (оптимизация моделей классификации /50/, коллективные решения задач кластерного анализа /51,52/), Дюкова (асимптотически-оптимальные логические алгоритмы /21,22/), Сенько (алгоритмы взвешенного статистического распознавания /56/), Асланян (логические алгоритмы распознавания) /60/, Донской (решающие деревья /16, 17/) и многие другие исследователи России, СНГ и дальнего зарубежья.

Разработки программных систем анализа данных и прогноза по прецедентам также активно ведутся в России и ведущих зарубежных странах. Прежде всего, это статистические пакеты обработки данных и визуализации (SPSS, STADIA, STATGRAPHICS, STATISTICA, SYSTAT, Олимп:СтатЭксперт Prof., Forecast Expert, и другие), в основе которых лежат методы различных разделов математической статистики – проверка статистических гипотез, регрессионный анализ, дисперсионный анализ, анализ временных рядов, и др. Использование статистических программных продуктов стало

стандартным и эффективным инструментом анализа данных, и, прежде всего, начального этапа исследований, когда находятся значения различных усредненных показателей, проверяется статистическая достоверность различных гипотез, находятся регрессионные зависимости. Вместе с тем статистические подходы имеют и существенные недостатки. Они позволяют оценить (при выполнении некоторых условий) статистическую достоверность значения прогнозируемого параметра, гипотезы или зависимости, однако сами методы вычисления прогнозируемых величин, выдвижения гипотез или нахождения зависимостей имеют очевидные ограничения. Прежде всего находятся усредненные по выборке величины, что может быть достаточно грубым представлением об анализируемых или прогнозируемых параметрах. Любая статистическая модель использует понятия «случайных событий», «функций распределения случайных величин» и т.п., в то время как взаимосвязи между различными параметрами исследуемых объектов, ситуаций или явлений являются детерминированными. Само применение статистических методов подразумевает наличие определенного числа наблюдений для обоснованности конечного результата, в то время как данное число может быть существенно больше имеющегося или возможного. Т.е. в ситуациях анализа в принципе непредставительных данных, или на этапах начала накопления данных, статистические подходы становятся неэффективными как средство анализа и прогноза.

В последние годы появились узкоспециализированные пакеты интеллектуального анализа данных. Для данных пакетов часто характерна ориентация на узкий круг практических задач, а их алгоритмической основой является какая-либо одна из альтернативных моделей, использующая нейронную сеть, решающие деревья, ограниченный перебор, и т.п. /20/. Ясно, что подобные разработки существенно ограничены при практическом использовании. Во-первых, заложенные в них подходы не является универсальными относительно размерностей задач, типа, сложности и структурированности данных, величины шума, противоречивости данных, и т.п. Во-вторых, созданные и «настроенные» на решение определенных задач, они могут оказаться совершенно бесполезными для других. Наконец, множество задач, представляющих интерес практическому пользователю, обычно шире возможностей отдельного подхода. Например, пользователю может быть важно иметь численную характеристику надежности некоторого прогноза, но «решающее дерево» ее не вычисляет. «Нейронная сеть» выступает в роли «черного ящика», предлагающего некоторый прогноз без его обоснования. Логические методы распознавания позволяют выявлять логические закономерности в данных и использовать их при прогнозировании, но при наличии

линейных зависимостей между признаками и прогнозируемой величиной точность прогноза, сделанного «линейной машиной», может быть заметно выше.

Таким образом, на настоящем уровне развития методов решения задач анализа данных и распознавания, представляется предпочтительным путь создания программных средств, включающих основные существующие разнообразные подходы. В данном случае повышаются шансы подбора из имеющихся алгоритмов такого алгоритма, который обеспечит наиболее точное решение интересующих пользователя задач на новых данных. Другим важным атрибутом систем анализа и классификации должно быть наличие средств автоматического решения задач распознавания и классификации коллективами алгоритмов. Действительно, стандартной ситуацией является наличие нескольких альтернативных алгоритмов или решений, равнозначных для пользователя. Для выбора из них одного наиболее предпочтительного не хватает информации. Тогда естественной альтернативой выбору является создание на базе имеющихся алгоритмов или решений новых, более предпочтительных.

Теоретические основы практической реализации идеи решения задач анализа данных коллективами алгоритмов были разработаны в ВЦ РАН в рамках алгебраического подхода для решения задач распознавания (логическая и алгебраическая коррекция алгоритмов) в 1976-1980 /25, 26, 28/ и комитетного синтеза классификаций для задач кластерного анализа (автоматической классификации) в 1981-1982 годах /51,52/. Позднее появились исследования в данной области и в других странах.

В алгебраическом подходе новые алгоритмы распознавания строятся в виде полиномов над исходными алгоритмами (применение алгебраических корректоров) или в виде специальных булевских функций (логических корректоров). Теоретическим базисом является теорема о существовании для произвольного алгоритма распознавания ему эквивалентного стандартного алгоритма, представимого в виде произведения распознающего оператора и решающего правила /26/. Это позволяет описать основные результаты вычислений произвольных алгоритмов распознавания в стандартном виде с помощью числовых матриц оценок («мер принадлежности» объектов к классам) и информационных матриц окончательных ответов (классификаций). Матрицы оценок различных распознающих алгоритмов являются «исходным материалом» для синтеза в виде полиномов новых матриц оценок, которые задают основу нового скорректированного решения задачи распознавания. Алгебраический подход позволяет строить алгоритмы, безошибочные на «обучающем» материале или совершающие меньшее число ошибок, чем каждый из исходных алгоритмов.

В настоящее время существует множество разнообразных подходов и конкретных эвристических алгоритмов для решения задач кластерного анализа (таксономии, или классификации без учителя), когда требуется найти естественные группировки похожих объектов (кластеры) по заданной выборке их векторных признаковых описаний. Решения, найденные различными алгоритмами, могут существенно отличаться друг от друга и даже фактически не соответствовать заложенной в данных действительности. Поиск наилучшего решения затруднен отсутствием общепризнанных универсальных критериев качества решений. Методы построения оптимальных коллективных решений в задачах кластерного анализа позволяют находить такие группировки объектов, которые являются эквивалентными с позиций сразу нескольких исходных алгоритмов. Оптимальные кластеризации находятся в результате решения специальных дискретных оптимизационных задач на перестановках.

В настоящей монографии представлено современное состояние в области практических методов распознавания, классификации и анализа данных, и приведено краткое описание программной системы «РАСПОЗНАВАНИЕ», включающей основные подходы.

Книга ориентирована на круг читателей из различных предметных областей, интересующихся применением современных практических методов анализа данных и распознавания. Поскольку данные приложения возникают в технических и гуманитарных областях, в науке и производстве, бизнесе и финансах, авторы хотели изложить суть методов и подходов максимально простым языком, доступным широкому кругу читателей, избегая излишней символики и научной строгости. При описании отдельных подходов авторы стремились выразить прежде всего их основную алгоритмическую суть, понимание которой является полезным для более эффективного использования системы. Следует отметить, что хотя детализированным описаниям теории и практики распознавания посвящены сотни статей и монографий, многие представленные в настоящей монографии материалы публикуются впервые.

В первой главе рассмотрена задача распознавания (классификации с учителем) и современное состояние в области практических методов для ее решения. Рассмотрены основные этапы в развитии теории и практики распознавания: создание эвристических алгоритмов, модели распознавания и оптимизация моделей, алгебраический подход к коррекции моделей. Приведены краткие описания основных подходов (основанных на построении разделяющих поверхностей, потенциальных функций, статистические и нейросетевые модели, решающие деревья, и другие). Расширенные описания методов, включенных в систему РАСПОЗНАВАНИЕ, приведены при необходимости в третьей

главе. Более подробно описаны основные подходы и алгоритмы комбинаторно-логических методов распознавания (модели вычисления оценок или алгоритмы, основанные на принципе частичной прецедентности), разработанные в ВЦ РАН. В основе данных моделей лежит идея поиска важных частичных прецедентов в признаковых описаниях исходных данных (информативных фрагментов значений признаков, или представительных наборов). Для вещественных признаков находятся оптимальные окрестности информативных фрагментов. В другой терминологии, данные частичные прецеденты называют знаниями или логическими закономерностями, связывающими значения исходных признаков с распознаваемой или прогнозируемой величиной. Найденные знания являются важной информацией об исследуемых классах (образах) объектов. Они непосредственно используются при решении задач распознавания или прогноза, дают наглядное представление о существующих в данных взаимозависимостях, что имеет самостоятельную ценность для исследователей и может служить основой при последующем создании точных моделей исследуемых объектов, ситуаций, явлений или процессов. По найденной совокупности знаний вычисляются также значения таких практически полезных величин, как степень важности (информативности) признаков и объектов, логические корреляции признаков и логические описания классов объектов, и решается задача минимизации признакового пространства.

Во второй части первой главы приведены основные понятия алгебраического подхода для решения задач распознавания, общие выражения для записи алгебраических и логических корректоров.

Вторая глава посвящена методам решения основной задачи кластерного анализа (классификации без учителя) – нахождению группировок объектов (кластеров) в заданной выборке многомерных данных. Приведен краткий обзор основных подходов для решения задачи кластерного анализа и описание комитетного метода синтеза коллективных решений.

В третьей главе представлена первая версия универсальной программной системы интеллектуального анализа данных, распознавания и прогноза РАСПОЗНАВАНИЕ. В основу требований к системе положены идеи универсальности и интеллектуальности. Под универсальностью системы понимается возможность ее применения к максимально широкому кругу задач (по размерностям, по типу, качеству и структуре данных, по вычисляемым величинам). Под интеллектуальностью понимается наличие элементов самонастройки и способности успешного автоматического решения задач неквалифицированным пользователем. Для достижения данных показателей были проведены работы по объединению различных подходов в рамкой единой системы, в

частности, по унификации обозначений, форматов, пользовательских интерфейсов, единых форм представления результатов обработки данных и обеспечения в результате единого комфортного языка общения пользователя с различными методами распознавания и кластерного анализа.

В рамках Системы РАСПОЗНАВАНИЕ разработана библиотека программ, реализующих линейные, комбинаторно-логические, статистические, нейросетевые, гибридные методы прогноза, классификации и извлечения знаний из прецедентов, а также коллективные методы прогноза и классификации.

1. **Алгоритмы распознавания, основанные на вычислении оценок.** Распознавание осуществляется на основе сравнения распознаваемого объекта с эталонными по различным наборам признаков, и использования процедур голосования. Оптимальные параметры решающего правила и процедуры голосования находятся из решения задачи оптимизации модели распознавания - определяются такие значения параметров, при которых точность распознавания (число правильных ответов на обучающей выборке) является максимальной /25,26, 80/.

2. **Алгоритмы голосования по тупиковым тестам.** Сравнение распознаваемого объекта с эталонными осуществляется по различным «информативным» подмножествам признаков. В качестве подобных подсистем признаков используются тупиковые тесты (или аналоги тупиковых тестов для веществennозначных признаков) различных случайных подтаблиц исходной таблицы эталонов /15, 25, 26, 80/.

3. **Алгоритмы голосования по логическим закономерностям.**

По обучающей выборке вычисляются множества логических закономерностей каждого класса – наборы признаков и интервалы их значений, свойственные каждому классу. При распознавании нового объекта вычисляется число логических закономерностей каждого класса, выполняющихся на распознаваемом объекте. Каждое отдельное «выполнение» считается «голосом» в пользу соответствующего класса. Объект относится в тот класс, нормированная сумма «голосов» за который является максимальной. Настоящий метод позволяет оценивать веса признаков, логические корреляции признаков, строить логические описания классов, находить минимальные признаковые подпространства /76, 77, 81/.

4. **Алгоритмы статистического взвешенного голосования.**

По данным обучающей выборки находятся статистически обоснованные логические закономерности классов. При распознавании новых объектов вычисляется оценка вероятности принадлежности объекта к каждому из классов, которая является взвешенной суммой «голосов» /37, 56, 70, 78, 80/.

5. **Линейная машина.**

Для каждого класса объектов находится некоторая линейная функция. Распознаваемый объект относится в тот класс, функция которого принимает максимальное значение на данном объекте. Оптимальные линейные функции классов находятся в результате решения задачи поиска максимальной совместной подсистемы системы линейных неравенств, которая формируется по обучающей выборке. В результате находится специальная кусочно-линейная поверхность, правильно разделяющая максимальное число элементов обучающей выборки /19, 46/.

6. **Линейный дискриминант Фишера.**

Классический статистический метод построения кусочно-линейных поверхностей, разделяющих классы /19/. Благоприятными условиями применимости линейного дискриминанта Фишера являются выполнение следующих факторов: линейная отделимость классов, дихотомия, «простая структура» классов, невырожденность матриц ковариаций, отсутствие выбросов. Созданная модификация линейного дискриминанта Фишера позволяет успешно использовать его и в «неблагоприятных» случаях.

7. **Метод к-ближайших соседей.**

Классический статистический метод. Распознаваемый объект относится в тот класс, из которого он имеет максимальное число «соседей». Оптимальное число соседей и априорные вероятности классов оцениваются по обучающей выборке /19/.

8. **Нейросетевая модель распознавания с обратным распространением**

Создана модификация известного метода обучения нейронной сети распознаванию образов (метод обратного распространения ошибки). В качестве критерия качества текущих параметров нейронной сети используется гибридный критерий, учитывающий как сумму квадратов отклонений значений выходных сигналов от требуемых, так и количество ошибочных классификаций на обучающей выборке /57/.

9. **Метод опорных векторов.**

Метод построения нелинейной разделяющей поверхности с помощью опорных векторов. В новом признаковом пространстве (спрямляющем пространстве) строится оптимальная разделяющая гиперплоскость, представляющая собой нелинейную поверхность в исходном признаковом пространстве. Построение данной поверхности сводится к решению задачи квадратичного программирования /62/.

10. **Алгоритмы решения задач распознавания коллективами различных распознающих алгоритмов.**

Задача распознавания решается в два этапа. Сначала независимо применяются различные алгоритмы Системы. Далее автоматически находится оптимальное коллективное решение

с помощью специальных методов-«корректоров». В качестве корректирующих методов используются различные подходы /25, 26, 69, 84/.

11. **Методы кластерного анализа (автоматической классификации или обучения без учителя) /19/.**

Используются следующие известные подходы:

- алгоритмы иерархической группировки;
- кластеризация с критерием минимизации суммы квадратов отклонений;
- метод к-средних.

Возможно решение задачи классификации как при заданном, так и неизвестном числе классов.

12. **Алгоритм построения коллективных решений задачи кластеризации.**

Задача кластеризации решается в два этапа. Сначала находится набор различных решений (в виде покрытий или разбиений) при фиксированном числе кластеров с помощью различных алгоритмов Системы. Далее находится оптимальная коллективная кластеризация в результате решения специальной дискретной оптимизационной задачи /50/.

Поскольку для практического пользователя важнейшим моментом является оценка точности распознавания алгоритмов при решении новых задач, разработаны программные средства автоматического контроля качества распознавания и прогноза, а также оценки знаний. Система разработана для персональных компьютеров Pentium 200 и выше под управлением операционных систем Windows 95 OSR2, Windows 2000, Windows XP.

Система ориентирована на широкий круг пользователей различной квалификации для поддержки принятия оптимальных прогнозных и диагностических решений в различных областях производственной и социальной сферы:

- обработка данных социологических опросов;
- прогнозирование тенденций изменения макроэкономических показателей;
- анализ финансовых данных и прогноз финансовых показателей;
- оценка экономического состояния предприятий и перспектив их инвестирования;
- проблемы прогнозирования экологических последствий по малым выборкам прецедентов;
- широкий круг задач медицины, связанных с созданием систем поддержки принятия диагностических решений, обработкой медицинской статистики, анализа эффективности лекарств и прогноза последствий лечения;
- задачи геологического прогнозирования;

- задачи экспериментальной физики, связанные с анализом накопленного экспериментального материала на этапах выявления качественных взаимосвязей между физическими параметрами и созданием приближенных математических моделей;
- задачи прогнозирования свойств новых органических соединений в химии на основе имеющегося банка исследованных органических соединений;
- обработка и анализ данных в биологии, с целью оптимизации селекционных и генетических исследований;
- обширный круг задач распознавания изображений.

Основные отличительные признаки созданного продукта от имеющихся аналогов состоят в следующем:

- разнообразие имеющихся методов и более широкие их практические возможности;
- способность автоматической обработки больших массивов данных;
- возможность решения задач прогноза и распознавания редких или уникальных событий и процессов (исследование непредставительных или мало репрезентативных выборок);
- способность обработки разнотипных, частично-противоречивых и неполных данных;
- использование оригинальных разработок авторов (алгоритмы вычисления оценок, модели частичной прецедентности, коллективные методы распознавания и кластеризации, и другие), отсутствующих в аналогах данного проекта;
- способность автоматического анализа данных и выработки прогнозных решений.

Настоящая система и ее предшественники (пакеты прикладных программ ПАРК /29/, ОБРАЗ /9/, диалоговая система ДИСАРО /23/, системы ЛОРЕГ /6/ и TaxonSearch /79/) использовались для решения и исследования многочисленных прикладных задач, многие из которых получили практическое внедрение.

Приложения в области бизнеса и финансов:

- оценка стоимости квартир по ее внутренним и внешним характеристикам (жилая площадь, строительный материал дома, местонахождение, этаж, удаленность от станции метро, и др.;
- оценка экономического состояния предприятий легкой промышленности по комплексу финансовых показателей и структуре рабочего персонала;
- подтверждение кредитных карточек;
- оценка стоимости жилья в частном секторе г.Бостона;
- распознавание типа движущихся объектов по комплексу акустических и сейсмических признаков с целью создания автоматических охранных систем.

Приложения в медицине и здравоохранении:

- кластеризация возрастных распределений населения

- распознавание рака груди;
- прогноз результатов лечения остеогенной саркомы;
- прогноз динамики депрессивных синдромов;
- диагностика инсульта;
- прогноз результатов лечения рака мочевого пузыря;
- прогноз состояния пациента через год после сердечного приступа по данным эхокардиограмм;
- диагностика сердечных сосудов;
- прогноз летального исхода при гепатите;
- прогноз диабета;
- диагностика меланомы по комплексу геометрических и радиологических признаков;
- оценка степени тяжести заболевания пневмонией.

Задачи технической диагностики:

- распознавание солеобразования в нефтедобывающем оборудовании;
- контроль состояния технических устройств.

Приложения в сельском хозяйстве:

- прогноз урожайности пшеницы по состоянию посевов за 2 и за 4 недели до созревания;
- распознавание преобладающих пород в лесных массивах по данным дистанционного зондирования.

Приложения в химии, физике и биологии:

- распознавание мест локализации протеина;
- прогноз свойств твердых сплавов стали;
- прогноз свойств новых неорганических соединений;
- распознавание наличия особенностей в ионосфере по виду отраженного сигнала.

Приложения в геологии:

- распознавание месторождений редких металлов и нефти.

Приложения в области обработки изображений:

- распознавание рукописных цифр;
- распознавание трехмерных объектов.

Иллюстративные примеры практических применений системы РАСПОЗНАВАНИЕ приведены в четвертой главе.

Настоящая монография выполнена при поддержке Российского фонда фундаментальных исследований (проекты №02-01-08007 инно, 02-01-00558, 03-01-00580, 02-07-90134, 02-07-90137) и Программы №17 «Параллельные вычисления и многопроцессорные вычислительные системы» Президиума РАН.

Часть книги написана молодыми исследователями Бирюковым А.С. (глава 5), Ветровым Д.П. (разд. 1.4.4, 3.6, 3.8, 3.10, 3.11, 3.12), Кропотовым Д.А. (разд. 3.7, 3.8, 3.10, 3.11, 3.13). В авторский коллектив программной системы входят также и другие исследователи: Докукин А.А., Катериночкина Н.Н., Обухов А.С., Романов М.Ю., Рязанов И.В., Челноков Ф.Б.

Практическая реализация системы РАСПОЗНАВАНИЕ выполнена в ООО «Центр технологий анализа и прогнозирования «РЕШЕНИЯ» при поддержке Фонда содействия развитию малых форм предприятий в научно-технической сфере. В следующей версии системы РАСПОЗНАВАНИЕ предполагается расширить пакет методов распознавания и кластерного анализа. Будут включены решающие деревья в задачах распознавания, новые алгоритмы поиска логических закономерностей, алгоритм восстановления компонент смесей нормальных выборок и проекционный метод кластерного анализа, алгоритмы генерации новых информативных сложных признаков и минимизации признаковых пространств, новые средства визуализации данных и знаний. Система будет адаптирована для решения задач прогнозирования дискретных событий по выборкам многомерных данных. Дополнительная информация о системе, новости по ее модификации, а также сведения о возможности ее приобретения размещаются по адресу http://www.solutions-center.ru.

Авторы надеются, что книга и приложенная демо-версия системы РАСПОЗНАВАНИЕ будут полезны широкому кругу исследователей, бизнесменов, инженеров, студентов и аспирантов, интересующимся современными компьютерными средствами анализа информации и извлечения знаний, а программная система РАСПОЗНАВАНИЕ окажется эффективным инструментом для решения многих научных, производственных, финансовых и социальных практических задач.

Глава1. Математические методы распознавания (классификации с учителем) и прогноза.

1.1. Задачи распознавания или классификации с учителем.

Исходной информацией являются описания объектов (ситуаций, предметов, явлений или процессов) *S* в виде векторов значений признаков для рассматриваемых объектов $S = (x_1(S), x_2(S),..., x_n(S))$ и значений некоторого «основного свойства» *y(S)* объекта *S*. Свойство *y(S)* принимает конечное число значений. Значения признаков $x_i, i = 1,2,...,n$, характеризующих различные стороны-свойства объектов *S* **,** для некоторых объектов $S_1, S_2,..., S_m$ считаются известными. Предполагается, что существует функциональная связь между признаками и основным свойством (неизвестная пользователю). Задача распознавания (прогноза, идентификации, «классификации с учителем») состоит в определении значения свойства *y(S)* некоторого объекта *S* по информации $S_1, S_2,..., S_m, y(S_1), y(S_2),..., y(S_m)$ (обучающей или эталонной выборке). Таким образом, задача распознавания может быть представлена как специальная задача экстраполяции функции, зависящей от конечного числа разнотипных переменных – признаков, и заданной в виде таблицы значений в конечном числе точек. Задачу построения алгоритма вычисления значений данной неизвестной функции в новых точках по известной совокупности ее значений в конечном числе точек называют задачей обучения распознаванию (построение распознающего алгоритма), а вычисление значения функции для произвольного нового набора признаков – задачей распознавания. Обычно вместо термина «основное свойство объекта» используют термин «класс объекта». Объекты, имеющие равные значения основного свойства считаются принадлежащими одному множеству (образу, классу объектов), и задача распознавания по прецедентам формулируется как задача отнесения объекта к одному из классов.

Следует осознавать, что множество значений отдельного признака может быть изначально весьма сложно устроенным. Так, значением признака может быть функция одной вещественной переменной (например, электрокардиограмма), определенная на данном отрезке числовой оси и имеющая не более, чем заданное число точек разрыва первого рода или изображение фиксированного района, полученное при аэрофотосъемке и съемке из космоса. Как правило, существенная доля признаков имеет более простую природу, допускает только значения "да", "нет", "неизвестно", выражается числом или числовым вектором (результат измерения или измерений) или имеет большее, чем три, но конечное число градаций, например, "неизвестно", "определенно нет", "вероятнее нет, чем да", "вероятнее да, чем нет", "определенно да", - пять градаций.

Формирование системы признаков и определение множества допустимых значений каждой части (признака) практически не поддается формализации. Это - работа эксперта-специалиста или группы экспертов. На этом этапе возможно множество альтернативных решений: можно выделить, например, небольшое число признаков со сложно устроенными множествами значений - изображения, сигналы и т.п., а можно "аппроксимировать" сложные признаки наборами простых. Так, вместо изображения или сигнала часто вводится совокупность его относительно простых характеристик, значения которых и запоминаются при описании ситуации. При этом число признаков увеличивается, но "сложность" отдельной части уменьшается. Так, вместо функции с конечным числом точек разрыва можно запоминать первые коэффициенты разложения в ряд Фурье, вместо полного оттиска пальца - число типовых элементарных локальных конфигураций. Именно проблемы замены сложных признаков - сигналов, изображений - на множество простых числовых признаков вызвали к жизни такие развитые отрасли прикладной математики как обработка сигналов, обработка и распознавание изображений. Если первая отрасль считается традиционной и полностью оформившейся, то вторая быстро развивается именно в последние годы.

Мы будем далее считать, что признаки принимают числовые значения, выражающие степень выраженности какого-то свойства. Случаи простого наличия или отсутствия какого-то свойства (бинарные признаки) будут кодироваться значениями 1 и 0. В случаях, когда признак принимает конечное число значений (к-значные признаки), значения признаков будут кодироваться *0, 1, 2, ..., к-1*. Бинарные и *к*-значные признаки будут рассматриваться как частный случай числовых признаков. Данные признаковые описания в виде числовых векторов являются в настоящее время практически общепринятыми и именно они используются в системе «РАСПОЗНАВАНИЕ». Заметим, что этап описания объектов в виде набора числовых признаков обычно успешно решается специалистами соответствующих предметных областей и фактически давно используется при начальной систематизации данных. Обычным в практике является также отсутствие по какой-либо причине информации о значениях части признаков у некоторых объектов. В данных случаях «пробелы» или «пропуски» значений признаков кодируются специальным символом. Алгоритмы распознавания решают задачи распознавания объектов по признакам, значения которых для данного объекта известны. При этом учитывается наличие пропусков и их количество.

Далее будем считать, что информация $S_1, S_2, ..., S_m, y(S_1), y(S_2), ..., y(S_m)$ задана в виде таблицы обучения $T_{nml} = (a_{ij})_{m \times n}$, где строки соответствуют признаковым описаниям

объектов длины n, строкам $S_1, S_2, ..., S_{m_1}$ соответствуют значения основного признака $y(S_i) = 1$ (объекты принадлежат классу K_1), строкам $S_{m_1+1}, S_{m_1+2}, ..., S_{m_2}$ соответствуют значения основного признака $y(S_i) = 2$ (объекты принадлежат классу K_2), и т.д. Строкам $S_{m_{l-1}+1}, S_{m_{l-1}+2}, ..., S_m$ соответствуют значения основного признака $y(S_i) = l$ (объекты принадлежат классу K_l), т.е. $S_{m_{i-1}+1}, S_{m_{i-1}+2}, ..., S_{m_i} \in K_i, i = 1,2,...,l, m_0 = 1, m_l = m$.

$$\left.\begin{array}{cccc} a_{11} & a_{12} & ... & a_{1n} \\ a_{21} & a_{22} & ... & a_{2n} \\ . & . & . & . \\ a_{m_1 1} & a_{m_1 2} & ... & a_{m_1 n} \end{array}\right\} \in K_1,$$

$$\left.\begin{array}{cccc} a_{m_1+1,1} & a_{m_1+1,2} & ... & a_{m_1+1,n} \\ a_{m_1+2,1} & a_{m_1+2,2} & ... & a_{m_1+2,n} \\ . & . & . & . \\ a_{m_2 1} & a_{m_2 2} & ... & a_{m_2 n} \end{array}\right\} \in K_2,$$

.....

$$\left.\begin{array}{cccc} a_{m_{l-1}+1,1} & a_{m_{l-1}+1,2} & ... & a_{m_{l-1}+1,n} \\ a_{m_{l-1}+2,1} & a_{m_{l-1}+2,2} & ... & a_{m_{l-1}+2,n} \\ . & . & . & . \\ a_{m1} & a_{m2} & ... & a_{mn} \end{array}\right\} \in K_l,$$

где $a_{ij} = x_j(S_i)$.

Таким образом, для решения задачи распознавания по прецедентам требуется на основе таблицы обучения T_{nml} создать алгоритм, который будет правильно определять класс, которому принадлежит предъявленный к распознаванию объект. Формально алгоритм распознавания будем записывать в следующем виде:

$$A(S) = (\alpha_1^A(S), \alpha_2^A(S), ..., \alpha_l^A(S)), \alpha_i(S) \in \{0,1,\Delta\}, i = 1,2,...,l. \qquad (1.1)$$

Здесь $\alpha_i^A(S) = 1$ означает отнесение алгоритмом объекта S в класс K_i, $\alpha_i^A(S) = 0$ означает решение алгоритма «объект S не принадлежит классу K_i», $\alpha_i^A(S) = \Delta$ означает отказ от классификации объекта S данным алгоритмом относительно класса K_i.

В заключение параграфа отметим, что здесь не будут рассматриваться случаи с использованием других видов обучающей информации (в качестве основной или дополнительной к обучающей выборке). Например, подобной информацией могут быть заданные экспертами правила, связывающие признаки, значения признаков и классы, или функции принадлежности объектов к классам. Данный вид обучающей информации используется в структурных методах распознавания и алгоритмах, основанных на

применении нечетких множеств. В принципе, подобная дополнительная информация может быть включена в стандартные таблицы обучения в качестве дополнительных признаков /26/.

1.2. Алгоритмы распознавания по прецедентам (классификация с учителем).

В настоящее время существует множество различных подходов для решения задачи распознавания по прецедентам. Происхождение каждого из них связано с тем или иным «естественным» представлением о том, что «из себя представляют образы и как надо решать задачу распознавания». В настоящем параграфе кратко рассматриваются такие основные подходы и приводятся ссылки на конкретные алгоритмы. Следует отметить, что, как правило, по каждому подходу имеются специальные монографии, изучение которых требует специальной подготовки.

1.2.1. Статистические алгоритмы распознавания.

Предполагается, что объекты обучающей выборки и распознаваемые объекты принадлежат к одной и той же генеральной совокупности. Считается, что объективно существует совместное вероятностное распределение элементов данной генеральной совокупности по классам и в признаковом пространстве. В случаях, когда такое распределение известно, существует простое оптимальное решение задачи распознавания принадлежности объектов классам $K_1,...,K_l$. Предположим, что нам необходимо классифицировать объект S, признаковое описание которого представлено вектором $\mathbf{x}(S)$. Для каждого из классов $K_1,...,K_l$ вычисляется условная вероятность принадлежности $\mathbf{P}(K_i \mid \mathbf{x})$. Объект S относится к тому классу, для которого условная вероятность принадлежности максимальна. Данное решающее правило минимизирует вероятность ошибочной классификации. В литературе его принято называть байесовским решающим правилом или оптимальным байесовским классификатором. К сожалению, байесовское решающее правило не может быть реализовано в подавляющем большинстве практических задач из-за того, что вероятностное распределение неизвестно.

Однако мы можем попытаться оценить условные вероятности принадлежности $\mathbf{P}(K_i \mid \mathbf{x})$, используя информацию, содержащуюся в обучающей выборке.

Метод k-ближайших соседей. Простым, но достаточно эффективным подходом здесь является метод k-ближайших соседей. Оценка условных вероятностей $\mathbf{P}(K_i \mid \mathbf{x})$ в методе k-ближайших соседей ведется по ближайшей окрестности V_k точки $\mathbf{x}(S)$ в признаковом пространстве, содержащей по крайней мере k признаковых описаний объектов

обучающей выборки. В качестве оценки $\mathbf{P}(K_i \mid \mathbf{x})$ выступает отношение $\frac{k_i}{k}$, где k_i - число объектов из класса K_i в окрестности V_k. Естественно, что поиск ближайшей окрестности должен основываться на использовании некоторой функции расстояний, заданной на множестве пар точек признакового пространства. В качестве такой функции расстояний в частности может выступать евклидова метрика. Точность распознавания методом k-ближайших соседей существенно зависит от числа k, оптимизация которого может производится по обучающей выборке. При этом в качестве оптимального берется то число ближайших соседей, при котором оценка точности распознавания с использованием режима скользящего контроля максимальна. Основным недостатком метода k-ближайших соседей является снижение его эффективности при малых объемах выборки и высокой размерности признакового пространства.

Аппроксимация с помощью многомерных нормальных распределений. Следует отметить, что условная вероятность $\mathbf{P}(K_i \mid \mathbf{x})$ связана с априорными вероятностями классов $\mathbf{P}(K_1), \ldots, \mathbf{P}(K_l)$ и с плотностями вероятностей $f_1(\mathbf{x}), \ldots, f_l(\mathbf{x})$ по формуле Байеса

$$\mathbf{P}(K_i \mid \mathbf{x}) = \frac{f_i(\mathbf{x})\mathbf{P}(K_i)}{\sum_{i=1}^{l} f_i(\mathbf{x})\mathbf{P}(K_i)} \quad , \qquad (1.2)$$

Оценки условных вероятностей $\mathbf{P}(K_i \mid \mathbf{x})$ могут быть получены по формуле (1.2) из оценок вероятностей $\mathbf{P}(K_1), \ldots, \mathbf{P}(K_l)$ и плотностей $f_1(\mathbf{x}), \ldots, f_l(\mathbf{x})$. При этом в качестве оценок априорных вероятностей классов $\mathbf{P}(K_1), \ldots, \mathbf{P}(K_l)$ могут быть взяты доли объектов соответствующих классов в обучающей выборке. Плотности вероятностей $f_1(\mathbf{x}), \ldots, f_l(\mathbf{x})$ восстанавливаются исходя из предположения об их принадлежности фиксированному типу распределения.

Наиболее часто используемым видом распределения является многомерная нормальная плотность, которая в общем виде представляется выражением

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{n/2} \mid \Sigma \mid^{1/2}} \exp\left[-\tfrac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t \Sigma^{-1} (\mathbf{x}-\boldsymbol{\mu})\right],$$

где n - размерность признакового пространства, $\boldsymbol{\mu}$ - математическое ожидание вектора признаков $\mathbf{x}$, Σ - матрица ковариаций компонент вектора $\mathbf{x}$, $|\Sigma|$ - детерминант матрицы Σ. Для построения распознающего алгоритма достаточно оценить вектор математических ожиданий $\boldsymbol{\mu}$ и матрицу ковариаций Σ для каждого из классов. Оценка $\hat{\boldsymbol{\mu}}_i$ математического ожидания вектора $\mathbf{x}$ вычисляется как вектор среднеарифметических значений компонент вектора $\mathbf{x}$ по всем объектам обучающей выборки, принадлежащим

классу K_i или $\hat{\boldsymbol{\mu}}^i = \frac{1}{m_i - m_{i-1}} \sum_{s_j \in K_i} \mathbf{x}(s_j)$, где $m_i - m_{i-1}$ - число объектов класса K_i в обучающей выборке.

Оценка элемента $\sigma^i_{kk'}$ матрицы ковариаций Σ_i вычисляется соответственно по формуле $\hat{\sigma}^i_{kk'} = \frac{1}{m_i - m_{i-1}} \sum_{s_j \in K_i} [x_k(s_j) - \mu^i_k][x_{k'}(s_j) - \mu^i_{k'}]$, $k, k' \in \{1, \ldots, n\}$. Матрицу, состоящую из элементов $\hat{\sigma}^i_{kk'}$ обозначим $\hat{\Sigma}_i$. В оптимальном байесовском классификаторе объект S относится в тот класс, для которого условная вероятность $\mathbf{P}(K_i \mid \mathbf{x})$ максимальна. Поскольку знаменатель в правой части формулы (1.2) одинаков для всех классов, то максимум $\mathbf{P}(K_i \mid \mathbf{x})$ (или любой монотонной функции от $\mathbf{P}(K_i \mid \mathbf{x})$) достигается для тех же самых классов, для которых достигается максимум $f_i(\mathbf{x})\mathbf{P}(K_i)$. Используя вместо плотностей $f_i(\mathbf{x})$ их приближения $\hat{N}_i(\mathbf{x})$ многомерным нормальным распределением и долю $\nu_i = \frac{m_i - m_{i-1}}{m}$ вместо $\mathbf{P}(K_i)$, можно построить распознающий алгоритм, аппроксимирующий оптимальный байесовский классификатор. В качестве оценки $g_i(\mathbf{x})$ за класс K_i удобнее использовать натуральный логарифм произведения $\hat{N}_i(\mathbf{x})$ и ν_i:

$$g_i(\mathbf{x}) = \ln[\nu_i \hat{N}_i(\mathbf{x})] = -\tfrac{1}{2}(\mathbf{x}^t \hat{\Sigma}_i^{-1}\ \mathbf{x}) + \mathbf{w}_i^t \mathbf{x} + g_i^0,$$

где $\mathbf{w}_i = \hat{\Sigma}_i^{-1} \hat{\boldsymbol{\mu}}_i$, $g_i^0 = -\tfrac{1}{2} \boldsymbol{\mu}_i^t \hat{\Sigma}_i^{-1}\ \boldsymbol{\mu}_i - \tfrac{1}{2} \ln(|\ \hat{\Sigma}_i\ |) + \ln(\nu_i) - \tfrac{n}{2} \ln(2\pi)$ - не зависящее от вектора $\mathbf{x}$ слагаемое. Распознаваемый объект S, относится к тому классу, оценка $g_i[\mathbf{x}(S)]$ за который максимальна. Следует отметить, что полученное приближение оптимального байесовского решающего правила является квадратичным по признакам для случаев, когда ковариационные матрицы для разных классов отличаются друг от друга. В случае, если ковариационные матрицы для разных классов совпадают, различия между их оценками стремятся к 0 с ростом объема обучающей выборки. При этом слагаемое $-\tfrac{1}{2}(\mathbf{x}^t \hat{\Sigma}_i^{-1}\ \mathbf{x})$ оказывается практически одинаковым для всех классов и рассматриваемая аппроксимация байесовского классификатора превращается в метод, использующий для разделения классов линейные поверхности. Основными недостатками метода, основанного на аппроксимации плотностей распределений классов многомерными нормальными распределениями, является его низкая эффективность при отклонении от нормальности реальных распределений. Особенно эта проблема усугубляется при

высокой размерности n признакового пространства, что связано с необходимостью оценивания $\frac{n(n+1)}{2}$ элементов матрицы Σ.

Линейный дискриминант Фишера. Большую популярность среди исследователей приобрел метод, предложенный Фишером еще в 1936 году. В основе метода лежит попытка разделить объекты двух классов, построив в многомерном признаковом пространстве такую прямую, чтобы проекции на нее описаний объектов этих двух классов были максимально разделены. Пусть вектор $\mathbf{w}$ задает направление некоторой прямой. Проекцией произвольного вектора $\mathbf{x}$ на направление, задаваемое $\mathbf{w}$, является отношение $\mathbf{w}^t\mathbf{x}/|\mathbf{w}|$, где $|\mathbf{w}|$- абсолютная величина вектора $\mathbf{w}$, которая реально является несущественным масштабным коэффициентом. В качестве меры различий проекций двух классов K_1 и K_2 на направление $\mathbf{w}$ было предложено использовать функционал

$$J(\mathbf{w}) = \frac{[\bar{y}_1(\mathbf{w}) - \bar{y}_2(\mathbf{w})]^2}{d_1(\mathbf{w}) + d_2(\mathbf{w})},$$

где $\bar{y}_i(\mathbf{w})$- среднее значение проекций векторов описаний объектов обучающей выборки из класса K_i, т.е. $\bar{y}_i(\mathbf{w}) = \frac{1}{(m_i - m_{i-1})|\mathbf{w}|}\sum_{s_j \in K_i}\mathbf{w}^t\mathbf{x}(s_j)$, а $d_i(\mathbf{w})$- выборочная дисперсия проекций векторов описаний объектов обучающей выборки из класса K_i,

$$d_i(\mathbf{w}) = \frac{1}{(m_i - m_{i-1})}\sum_{s_j \in K_i}[\frac{\mathbf{w}^t\mathbf{x}(s_j)}{|\mathbf{w}|} - \bar{y}_i]^2,\ i \in \{1,2\}.$$

Смысл функционала $J(\mathbf{w})$ с точки зрения разделения двух классов ясен из его структуры. Он возрастает при увеличения отношения квадрата различия между средними значениями проекций двух классов к сумме внутриклассовых выборочных дисперсий.

Можно показать /19/, что функционал $J(\mathbf{w})$ достигает своего максимума при $\mathbf{w} = D_w^{-1}(\hat{\boldsymbol{\mu}}_1 - \hat{\boldsymbol{\mu}}_2)$, где D_w-матрица разброса внутри классов, $\hat{\boldsymbol{\mu}}_i = \frac{1}{(m_i - m_{i-1})}\sum_{s_j \in K_i}\mathbf{x}(s_j)$- выборочный центр класса K_i, $i \in \{1,2\}$. Матрица разброса внутри классов определяется как сумма матриц разброса по каждому из классов или $D_w = D_1 + D_2$, где $D_i = \frac{1}{(m_i - m_{i-1})}\sum_{s_j \in K_i}[\mathbf{x}(s_j) - \hat{\boldsymbol{\mu}}_i][\mathbf{x}(s_j) - \hat{\boldsymbol{\mu}}_i]^t$. Сравнивая линейный дискриминант Фишера с аппроксимацией оптимального байесовского классификатора с помощью многомерных нормальных распределений, нетрудно увидеть, что оба правила практически идентичны для случаев, когда мы можем пренебречь различиями ковариационных матриц классов.

Линейный дискриминант Фишера может быть распространен на случаи большего, чем 2 числа классов.

1.2.2. Алгоритмы распознавания, основанные на построении разделяющих поверхностей.

В основе данных подходов лежат геометрические модели классов. Предполагается, что множеству объектов каждого класса соответствует определенная область в *n*-мерном признаковом пространстве. Данные области имеют достаточно простую форму и их можно разделить «простой» поверхностью (прежде всего линейной, кусочно-линейной или квадратичной). Рассмотрим примеры данных алгоритмов. Будем считать для простоты, что имеются лишь два класса объектов.

Задача построения **линейной разделяющей поверхности** (гиперплоскости) состоит в вычислении некоторой линейной относительно признаков функции

$$f(\mathbf{x}) = a_1 x_1 + a_2 x_2 + \ldots + a_n x_n + a_{n+1} \qquad (1.3)$$

и использовании при классификации следующего упрощенного решающего правила:

$$\alpha^A(S) = \begin{cases} 1, & f(S) > 0, \\ \Delta, & f(S) = 0, \\ 0, & f(S) < 0. \end{cases} \qquad (1.4)$$

Здесь $\alpha^A(S) = 1$ означает отнесение объекта в первый класс, $\alpha^A(S) = 0$ - отнесение во второй, $\alpha^A(S) = \Delta$ - отказ от классификации объекта.

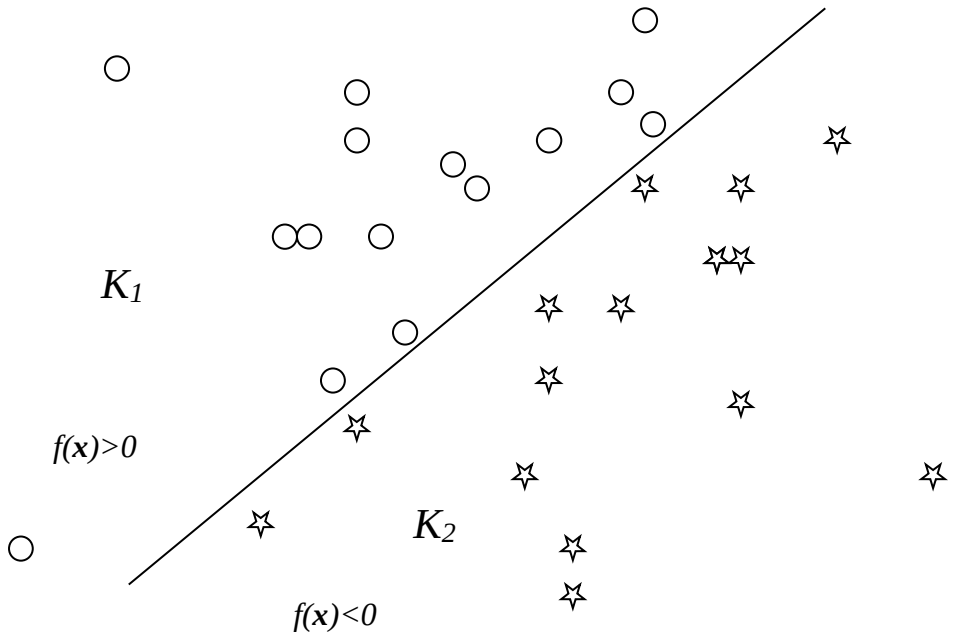

Рис. 1. Безошибочное разделение двух классов гиперплоскостью

Существуют различные варианты математических постановок критериев выбора $f(\mathbf{x})$. Основной постановкой является поиск такой функции (т.е. значений неизвестных коэффициентов $a_1, a_2, ..., a_n, a_{n+1},$), для которых число невыполненных неравенств в системе

$$\begin{aligned}
a_1x_1(S_1) + a_2x_2(S_1) + ... + a_nx_n(S_1) + a_{n+1} &> 0,\\
a_1x_1(S_2) + a_2x_2(S_2) + ... + a_nx_n(S_2) + a_{n+1} &> 0,\\
&...\\
a_1x_1(S_{m_1}) + a_2x_2(S_{m_1}) + ... + a_nx_n(S_{m_1}) + a_{n+1} &> 0,\\
&... \quad ... \quad ...\\
a_1x_1(S_{m_1+1}) + a_2x_2(S_{m_1+1}) + ... + a_nx_n(S_{m_1+1}) + a_{n+1} &< 0,\\
a_1x_1(S_{m_1+2}) + a_2x_2(S_{m_1+2}) + ... + a_nx_n(S_{m_1+2}) + a_{n+1} &< 0,\\
&...\\
a_1x_1(S_m) + a_2x_2(S_m) + ... + a_nx_n(S_m) + a_{n+1} &< 0,
\end{aligned} \qquad (1.5)$$

является минимальным. В данном случае при классификации по решающему правилу (1.4) достигается минимальное число ошибок (т.е. отнесений не в свой класс или отказ от классификации). Если система (1.5) совместна, тогда достаточно найти произвольное ее решение относительно неизвестных $a_1, a_2, ..., a_n, a_{n+1},$ для чего можно использовать релаксационные методы решения систем линейных неравенств /24/, метод конечных приращений /19/, алгоритмы линейного программирования и другие. Однако заранее не известно, совместна данная система или нет. Обычно системы (1.5) являются именно несовместными, т.е. обучающие выборки невозможно безошибочно разделить гиперплоскостью. Поэтому на методы решения систем (1.5) обычно накладывают следующее естественное ограничение: если система является совместной, тогда метод находит некоторое ее решение, если система несовместна – находится некоторое «обобщенное» решение системы. Под обобщенным решением понимают решение некоторой ее максимальной совместной подсистемы (находится такой набор $a_1, a_2, ..., a_n, a_{n+1},$ при котором будет выполнено максимальное число неравенств системы (1.5)), либо такой набор $a_1, a_2, ..., a_n, a_{n+1},$ при котором «нарушенность» системы (1.5) будет

минимальна /19/.

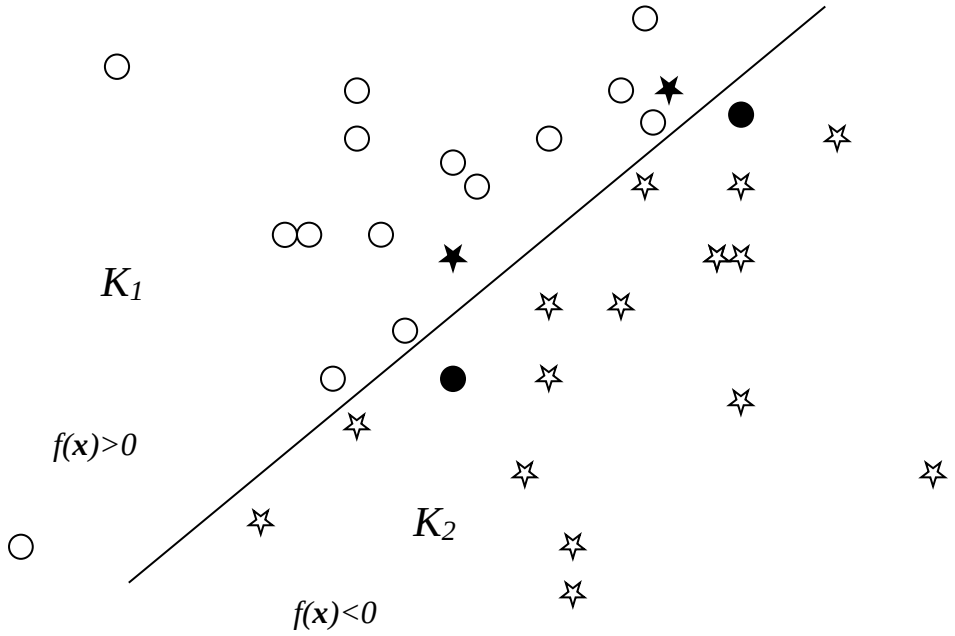

Рис. 2.Разделение двух классов гиперплоскостью с минимальным числом ошибок. Черным цветом отмечены объекты обучения, на которых оптимальная гиперплоскость совершает ошибки

Методы классификации с помощью разделяющей гиперплоскости просты и эффективны для относительно простых практических задач. Если конфигурация классов такова, что оптимальная гиперплоскость допускает слишком большое число ошибок на обучающей выборке, следует строить кусочно-линейные разделяющие поверхности, или применять другие подходы.

Для непротиворечивых таблиц обучения (т.е. при отсутствии равных признаковых описаний, принадлежащих различным классам) **метод комитетов** позволяет строить кусочно-линейную поверхность, безошибочно разделяющую объекты обучающей выборки /42, 43, 45/.

Рассмотрим для простоты задачу с двумя классами. Пусть дана некоторая совокупность линейных функций

$$f_i(\mathbf{x}) = a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{in}x_n + a_{i,n+1},\ i=1,2,\ldots,k. \qquad (1.6)$$

Условие правильной классификации всех объектов обучающей выборки некоторой функцией из (1.6) записывается в виде системы (1.7)

$$f_i(S_j) = a_{i1}x_1(S_j) + a_{i2}x_2(S_j) + \ldots + a_{in}x_n(S_j) + a_{i,n+1} > 0, \qquad j = 1,2,\ldots,m_1,$$

$$f_i(S_j) = a_{i1}x_1(S_j) + a_{i2}x_2(S_j) + \ldots + a_{in}x_n(S_j) + a_{i,n+1} < 0, \qquad j = m_{1+1}, m_{1+2},\ldots,m. \qquad (1.7)$$

Совокупность функций (1.6) называется комитетом для системы (1.7) , если каждому неравенству в системе (1.7) удовлетворяет более половины функций из (1.6) .

Пусть для заданной обучающей выборки построен некоторый комитет (1.6). Тогда решающее правило может быть записано в следующем виде:

$$\alpha^A(S) = \begin{cases} 1, & \sum_{i=1}^{k} sign(f_i(S)) > 0, \\ \Delta, & \sum_{i=1}^{k} sign(f_i(S)) = 0, \\ 0, & \sum_{i=1}^{k} sign(f_i(S)) < 0. \end{cases}$$

То есть объект относится в первый класс, если более половины функций положительны и во второй класс, если более половины функций отрицательны. В противном случае происходит отказ от распознавания.

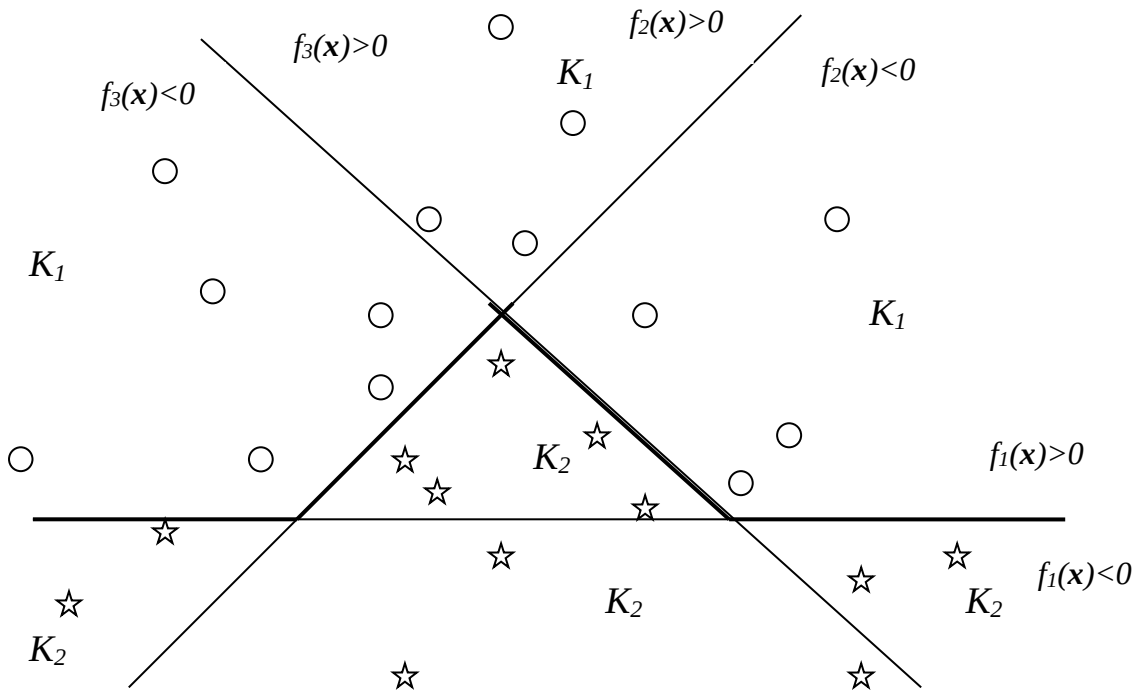

Рис. 3. Разделение двух классов комитетом из трех линейных функций

В работах /42, 43, 45/ доказано существование комитетов для непротиворечивых таблиц обучения, исследованы задачи построения минимальных комитетов, исследованы обобщения комитетов на нелинейные случаи.

Другим примером построения кусочно-линейных разделяющих поверхностей являются алгоритмы обобщенного портрета /11/.

С помощью алгоритмов кластерного анализа исходное пространство признаковых описаний делится на *p* «простых» непересекающихся областей таких, что классы линейно разделимы по объектам обучающей выборки для каждой из областей. Значение *p* заранее не известно. Для каждой области строится линейная разделяющая классы функция, максимально удаленная от классов. Классификация осуществляется в два этапа. Сначала определяется, какой области принадлежит распознаваемый объект S. Затем для его классификации применяется соответствующая линейная функция.

Прямым обобщением линейных и кусочно-линейных разделяющих поверхностей являются полиномиальные, в частности, квадратичные поверхности. Разделяющие функции ищутся в виде полиномов степени не выше k, где k задается заранее или вычисляется одновременно с коэффициентами. Степень k полинома (1.8) определяется максимальной суммой степеней параметров x_i в отдельных слагаемых (1.8)

$$f(x) = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i,j=1}^{n} a_{ij} x_i x_j + \sum_{i,j,k=1}^{n} a_{ijk} x_i x_j x_k + \ldots \qquad (1.8)$$

Обозначив $z_1 = 1, z_2 = x_1, z_3 = x_2, \ldots, z_{n+1} = x_n, z_{n+2} = x_1^2, z_{n+3} = x_1 x_2, \ldots, z_t = x_i x_j x_k, \ldots$ задача сводится к построению линейной разделяющей функции $f(\mathbf{z}) = \mathbf{a}^t \mathbf{z}$ в спрямляющем пространстве размерности *N* относительно переменных $(z_1, z_2, \ldots, z_N)$.

1.2.3. Метод потенциальных функций.

Метод основан на аналогии с задачами электростатики. Рассмотрим случай двух классов. Предполагается, что каждый элемент обучающей выборки S_i первого класса имеет положительный «заряд» $+q_i$ а элемент S_j второго класса - отрицательный заряд $-q_j$. Объекты первого класса создают «потенциал» $+q_i K(S, S_i)$ в каждой точке *S* пространства признаковых описаний, а объекты второго класса – потенциал $-q_i K(S, S_i)$. Здесь $K(S, S_i)$ является величиной потенциала, создаваемого в точке *S* единичным зарядом. Тогда значение потенциальной функции в точке *S* определяется как

$$g(S) = \sum_{S_i \in K_1} q_i K(S, S_i) - \sum_{S_i \in K_2} q_i K(S, S_i).$$

При классификации некоторого объекта *S* вычисляется значение потенциальной функции $g(S)$. Классификация проводится согласно решающему правилу

$$\alpha^A(S) = \begin{cases} 1, & g(S) > 0, \\ \Delta, & g(S) = 0, \\ 0, & g(S) < 0. \end{cases}$$

В качестве основных требований к виду функций $K(S, S_i)$ предъявляются следующие два:

1. $K(S_i, S_i) = \max_S K(S_i, S)$;

2. $K(S, S_i) > K(S, S_j)$ при $\|S_j - S\| > \|S_i - S\|$ (т.е. $K(S, S_i)$ монотонно убывает при $\|S_i - S\| \to +\infty$).

Распространенными примерами выбора $K(S, S_i)$ являются следующие:

1. $K(S, S_i) = \dfrac{\sigma^2}{\sigma^2 + \|S - S_i\|^2}$, где σ - числовой параметр;

2. $K(S, S_i) = \exp(-\dfrac{1}{2\sigma^2}\|S - S_i\|^2)$;

Таким образом, для классификации достаточно выбрать конкретный вид функции и задать неизвестные значения параметров $\sigma, q_1, q_2, ..., q_m$.

Вычисление значений параметров $q_1, q_2, ..., q_m$ при фиксированном σ (а следовательно и функции $g(S)$) осуществляется в процессе обучения согласно правилу коррекции (1.9) /2/. Пусть на некотором шаге имеется функция $g(S)$. Для классификации предъявляется некоторый объект S_i обучающей выборки. Если результат классификации правильный, предъявляется для классификации следующий объект обучающей выборки. Если классификация является неправильной, осуществляется коррекция функции $g(S)$ согласно правилу (1.9) .

$$g^*(S) = \begin{cases} g(S) + K(S, S_i), & g(S_i) \le 0, \quad S_i \in K_1, \\ g(S) - K(S, S_i), & g(S_i) \ge 0, \quad S_i \in K_2. \end{cases} \qquad (1.9)$$

Объекты обучающей выборки предъявляются, например, циклически или в случайном порядке. Процесс обучения продолжается до достижения безошибочного распознавания всех объектов обучающей выборки полученной функцией $g(S)$ или «стабилизации» - достижения ситуации, когда среднее число ошибок перестает уменьшаться за заданное число итераций.

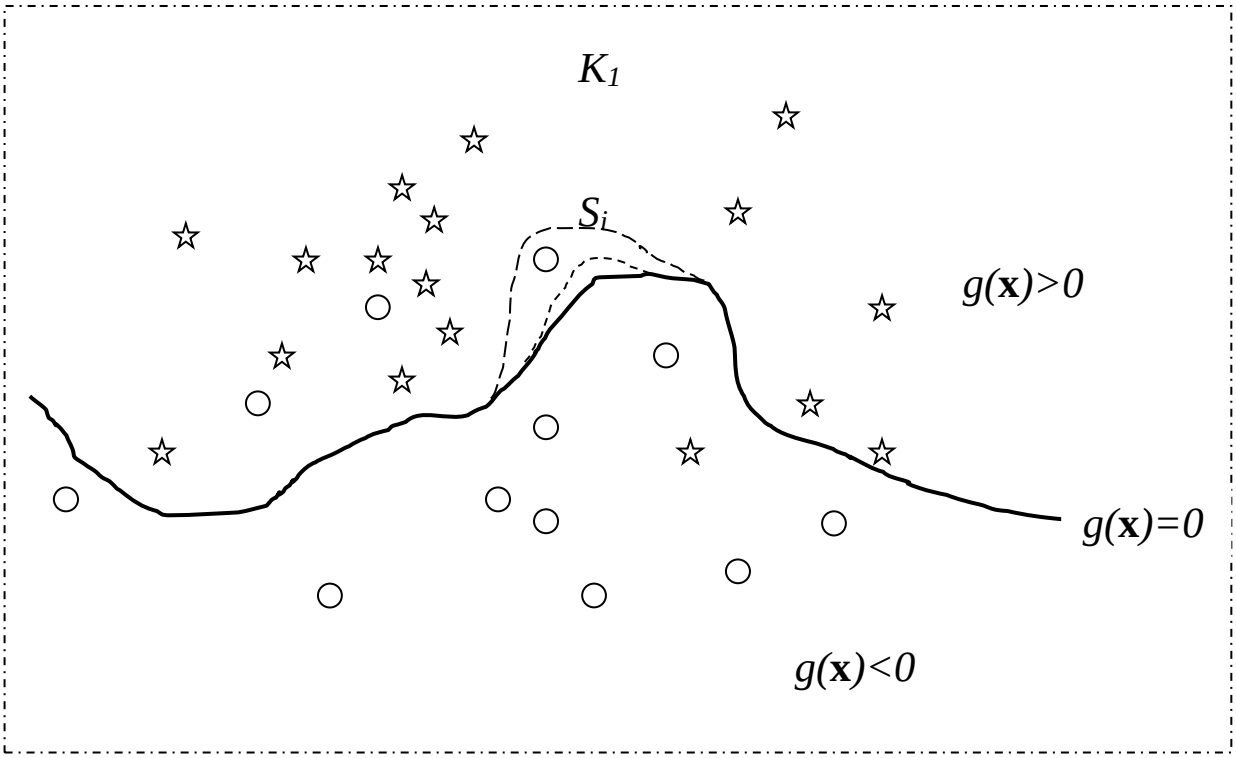

Рис.4. Примеры возможной коррекции функции $g(S)$ при условии $g(S_i) > 0$, $S_i \in K_2$.

На рис.4 проиллюстрирована коррекция потенциальной функции $g(S)$ (множество точек $g(S)=0$ изображено жирной линией) при предъявлении некоторого объекта из второго класса, неправильно классифицируемого данной функцией. Функция $g^*(S)$ будет отличаться от $g(S)$ практически лишь в окрестности S_i (участки заметного отличия отмечены пунктирами). При повторном предъявлении S_i потенциальная функция может правильно классифицировать объект (жирный пунктир) или опять неправильно. Во втором случае (тонкий пунктир), тем не менее, коррекция $g(S)$ будет сделана «в нужную сторону», $g^*(S_i) < g(S_i)$ /2, 19/.

1.2.4. Нейросетевые модели распознавания.

Данные алгоритмы являются попыткой моделирования способности человеческого мышления, в частности, способности обучаться и решать задачи распознавания по прецедентам. Они основаны на достижениях биологии и медицины - простейших моделях человеческого мозга, созданных в середине прошлого века. Биологический мозг рассматривается как множество элементарных элементов - нейронов, соединенных друг с другом многочисленными связями. Нейроны бывают трех типов: рецепторы (принимающие сигналы из внешней среды и передающие другим нейронам), внутренние нейроны (принимающие сигналы от других нейронов, преобразующие их и передающие другим нейронам) и реагирующие нейроны (принимающие сигналы от нейронов и вырабатывающие сигналы во внешнюю среду).

Рассмотрим простейшую и наиболее распространенную модель человеческого мозга, ориентированную на решение задач распознавания – искусственную многослойную нейронную сеть.

Элементарной ячейкой нейронной сети является модель искусственного нейрона (рис. 5).

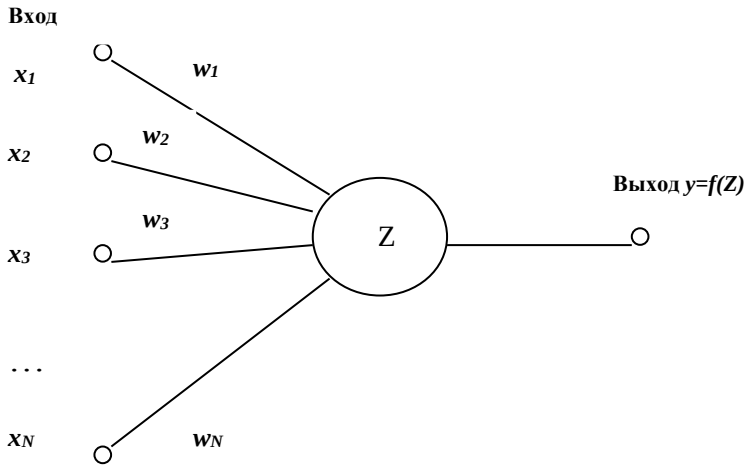

Рис. 5. Модель искусственного нейрона

Каждый внутренний или реагирующий нейрон имеет множество входных связей (синапсов), по которым поступают сигналы от других внутренних нейронов или рецепторов, и одну выходящую связь (аксон). Каждая связь имеет некоторый «вес» w_i. При поступлении на вход нейрона совокупности сигналов $x_1, x_2, ..., x_N$ они «усиливаются» с соответствующими весами $w_1, w_2, ..., w_N$. Нейрон переходит в состояние, числовая оценка которого вычисляется как $Z = \sum_{i=1}^{N} w_i x_i$. Величина выходного сигнала вычисляется как $f(Z)$, где f - активационная функция. Примеры активационных функций приведены на рис. 6-9.

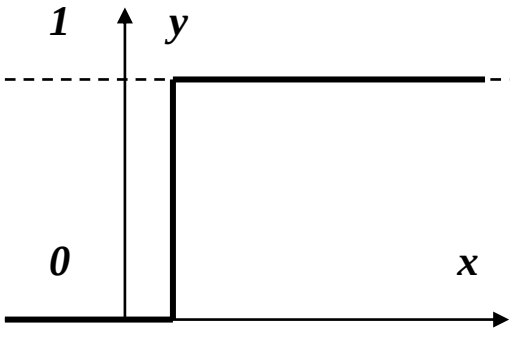

Рис. 6. Функция единичного скачка

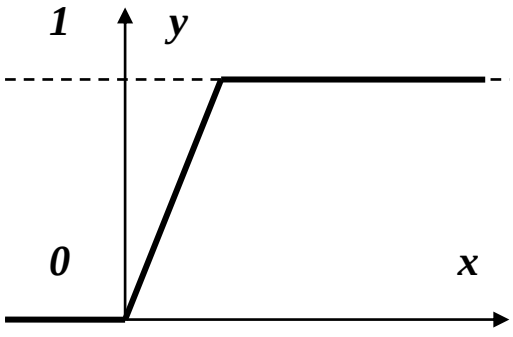

Рис. 7. Единичный скачок с линейным порогом

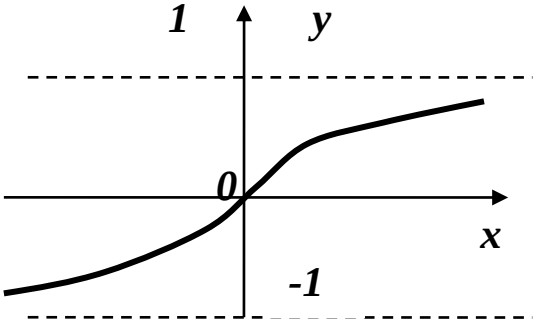

Рис. 8. Гиперболический тангенс *f(x)=th(x)*

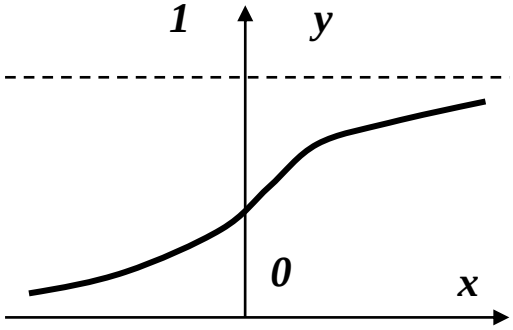

Рис.9. Функция сигмоида *f(x)=1/(1+exp(-ax))*

Нейрон считается «возбужденным», если выходной сигнал отличен от нуля, а величина $y = f(\sum_{i=1}^{N} w_i x_i)$ характеризует степень возбуждения. Вид функций и область их изменения отражают априорные представления о функционировании биологических нейронов: величина возбуждения зависит монотонно от состояния, ограничена снизу и сверху, и «сильно» меняется в небольшом интервале значений $Z = \sum_{i=1}^{N} w_i x_i$.

Наиболее простыми, распространенными и исследованными являются многослойные нейронные сети прямого действия. Общий вид подобной сети изображен на рис. 10. Сеть состоит из N слоев, каждые слой состоит из n_i нейронов, каждый нейрон j-го уровня связан с каждым нейроном j+1 – го уровня. Фиктивный нулевой слой состоит из n входных нейронов, на каждый из которых подается значение некоторого признака x_i. Результатами классификации являются выходные значения нейронов N -го слоя.

Распознаваемый объект S поступает на 0-й слой. Далее поступивший сигнал (признаковое описание) последовательно преобразуется по слоям согласно заданным фиксированным весам синаптических связей и выбранной активационной функции. Если $y_j^N(S), j = 1,2,...,l,$ есть значение j-го нейрона выходного слоя, то информационный вектор результатов классификации S вычисляется согласно (1.10)

$$\alpha_j^A(S) = \begin{cases} 1, & y_j^N(S) > 0, \\ \Delta, & y_j^N(S) = 0, \\ 0, & y_j^N(S) < 0. \end{cases} \qquad (1.10)$$

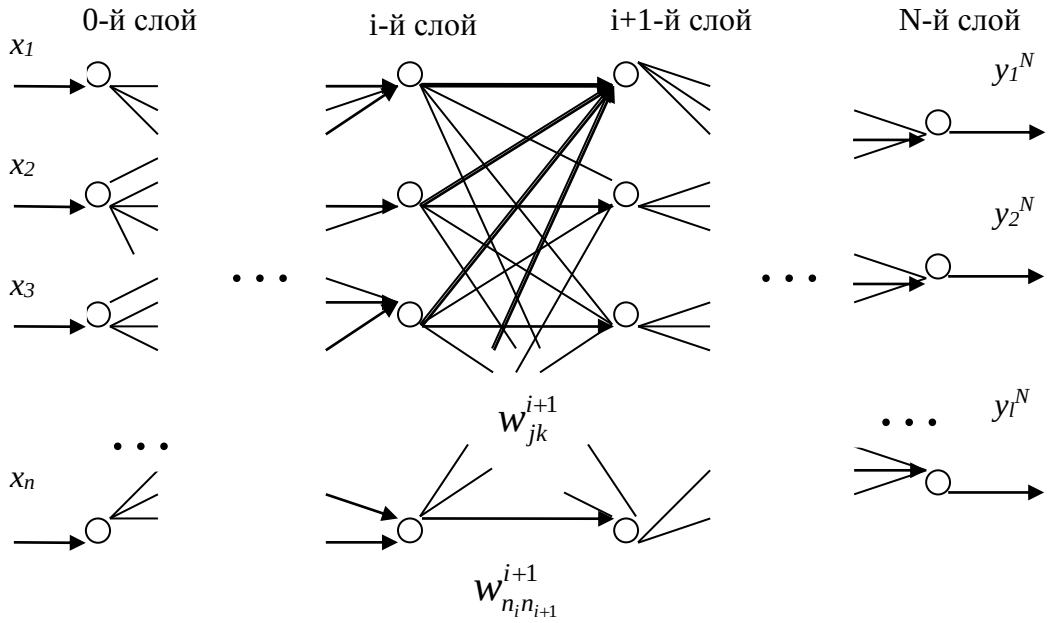

Рис.10. Структурная схема нейронной классифицирующей сети прямого действия

Значения неизвестных весовых коэффициентов находятся в результате процесса обучения сети. Начальные значения весовых коэффициентов задаются случайно. Объекты обучения последовательно поступают на вход сети. Если предъявленный объект классифицируется правильно, коэффициенты остаются прежними и на вход сети поступает следующий объект. Если при классификации объекта происходит ошибка, весовые коэффициенты изменяются определенным образом. Для однослойной сети они меняются согласно простой итерационной формуле подобно используемой в методе потенциальных функций. Для многослойной сети используются специальные рекуррентные формулы пересчета весовых коэффициентов от последнего уровня до первого (метод «обратного распространения ошибки»). Обучение заканчивается, если изменение коэффициентов не приводит к дальнейшему уменьшению суммарного числа ошибок на обучающей выборке /57/.

1.2.5. Решающие деревья.

Методы распознавания, основанные на построении решающих деревьев, относятся к типу логических методов. В данном классе алгоритмов распознавание объекта осуществляется как прохождение по бинарному дереву из корня в некоторую висячую вершину. В каждой вершине вычисляется определенная логическая функция. В зависимости от полученного значения функции происходит переход далее по дереву в левую или правую вершину следующего уровня. Каждая висячая вершина связана с

одним из классов, в который и относится распознаваемый объект, если путь по дереву заканчивается в данной вершине.

Бинарным корневым деревом (БД) называется дерево, имеющее следующие свойства:

а) каждая вершина (кроме корневой) имеет одну входящую дугу;

б) каждая вершина имеет имеет либо две, либо ни одной выходящей дуги.

Вершины, имеющие две выходящие дуги, называются внутренними, а остальные – терминальными или листьями.

Пусть задано *N* предикатов $\mathrm{P} = \{y_1 = P_1(S), y_2 = P_2(S), ..., y_N = P_N(S)\}$, определенных на множестве допустимых признаковых описаний *{S}*, именуемые признаковыми предикатами. Каждый предикат отвечает на вопрос, выполняется ли некоторое свойство на объекте *S* или нет. Примерами признаковых предикатов могут быть

$$P_1(S) = \begin{cases} 1, & \text{если } (1.5 \le x_2(S) \le 3.2) \,\&\, (x_5(S) = 0) \\ 0, & \text{в противном случае,} \end{cases}$$

$$P_2(S) = \begin{cases} 1, & \text{если } 0.5x_2(S) + 1.2x_7(S) < 1 \\ 0, & \text{в противном случае.} \end{cases}$$

Каждой строке $S_i = (a_{i1}, a_{i2}, ...a_{in})$ таблицы обучения $T_{nml} = \left(a_{ij}\right)_{m \times n}$ поставим в соответствие бинарную строку значений предикатов на описании $S_i \Rightarrow (y_{i1}, y_{i2}, ..., y_{iN}), y_{ij} = P_j(S_i)$. В результате, таблице $T_{nml} = \left(a_{ij}\right)_{m \times n}$ будет соответствовать бинарная таблица $T^0_{Nml} = \left(y_{ij}\right)_{m \times N}$ бинарных «вторичных описаний» объектов обучения.

Бинарное дерево называется решающим, если выполнены следующие условия:

1. каждая внутренняя вершина помечена признаковым предикатом из P;
2. выходящие из вершин дуги помечены значениями, принимаемыми предикатами в вершине;
3. концевые вершины помечены метками классов;
4. ни в одной ветви дерева нет двух одинаковых вершин.

Пример подобного дерева для конфигурации из трех классов (рис.11) приведен на рис. 12.

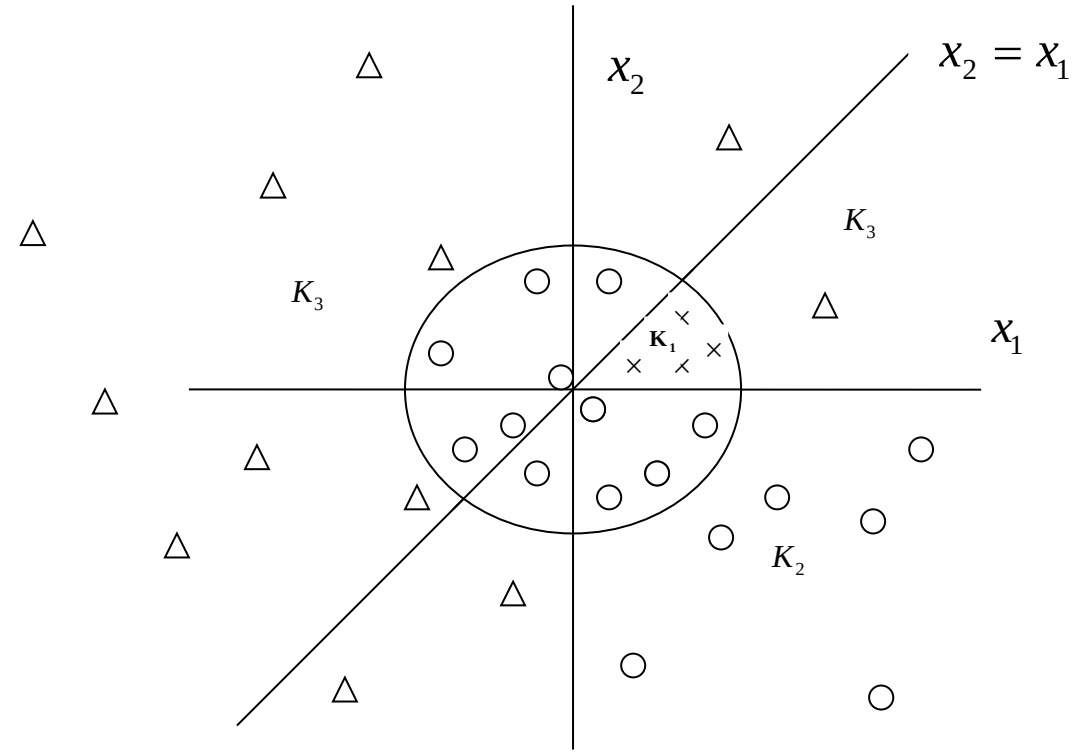

Рис. 11. Конфигурация объектов трех классов. Объекты классов K_1, K_2, K_3 обозначены, соответственно, символами $\times$, O, Δ

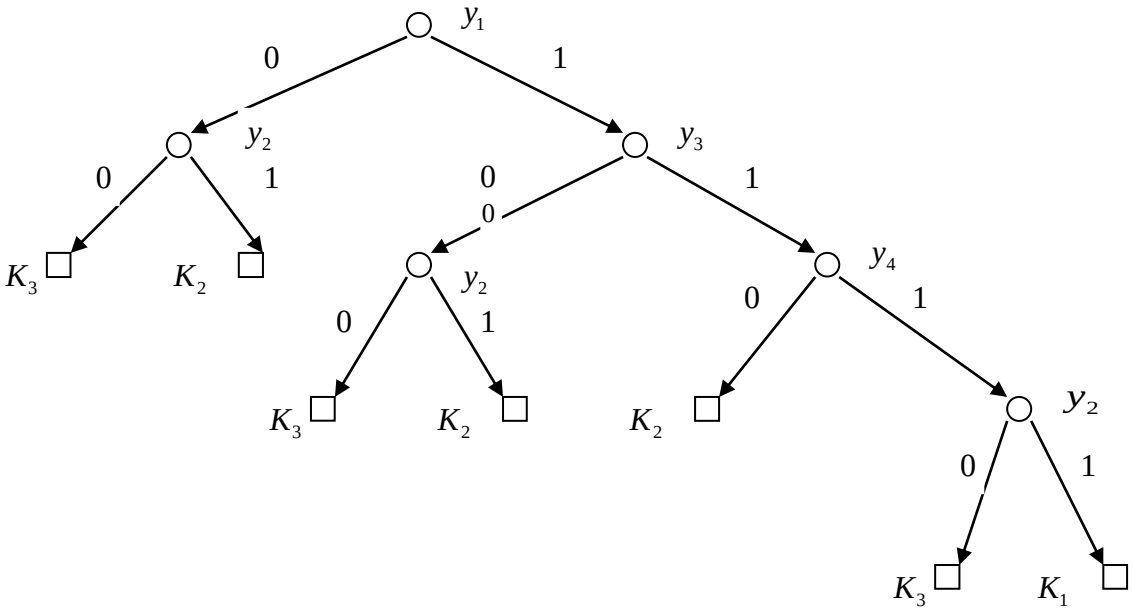

Рис.12. Решающее дерево для классов рис. 11

На рис.12 приведено некоторое решающее дерево, позволяющее правильно распознавать объекты трех классов, изображенных на рис.11. Вершины дерева помечены следующими предикатами: $y_1 = "x_1 > 0", y_2 = "x_1^2 + x_2^2 < 1", y_3 = "x_1 > x_2", y_4 = "x_2 > 0".$ Данному решающему дереву соответствуют характеристические функции классов $f_1(S) = y_1 y_2 y_3 y_4$, $f_2(S) = \bar{y}_1 y_2 \vee y_1 y_2 \bar{y}_3 \vee y_1 y_3 \bar{y}_4$, $f_3(S) = \bar{y}_1 \bar{y}_2 \vee y_1 \bar{y}_2 \bar{y}_3 \vee y_1 \bar{y}_2 y_3 y_4$, принимающие значение 1 на объектах «своего» класса и 0 на объектах остальных классов.

Приведем еще один пример решающего дерева (рис.13), построенного непосредственно по таблице обучения: $T_{5.4,2} = \begin{pmatrix} 00111 \\ 11101 \\ 10111 \\ 01010 \end{pmatrix} \begin{matrix} \in K_1 \\ \\ \\ \in K_2 \end{matrix}$. (1.11)

В качестве признаковых предикатов используются $y_1 = x_1$, $y_2 = x_2$.

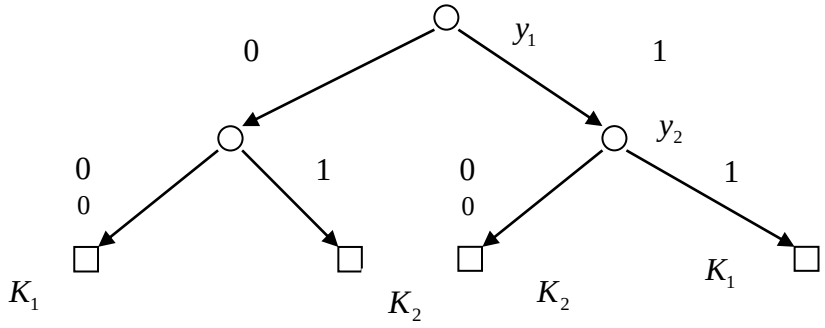

Рис.13. Решающее дерево бинарной таблицы обучения (1.11)

Здесь в качестве приближений для характеристических функций классов выбраны функции $f_1(S) = x_1x_2 \vee \bar{x}_1\bar{x}_2$, $f_2(S) = \bar{x}_1x_2 \vee x_1\bar{x}_2$. В данном примере, для построения решающего дерева использованы только два признака.

Задача построения решающего дерева по обучающим данным решается неоднозначно, методам построения решающих деревьев посвящена обширная литература /16, 17, 41, и др./.

1.3. Алгоритмы распознавания, основанные на принципе частичной прецедентности

Настоящий класс алгоритмов выделен в отдельный раздел по нескольким причинам. Представляя исторически логические подходы в теории распознавания, в рамках данного подхода разработана также теория алгоритмов вычисления оценок, объединяющая все существующие методы распознавания и положенная в основу алгебраического подхода в теории распознавания. Теоретические основы алгоритмов частичной прецедентности (вычисления оценок, голосования, или комбинаторно-логических алгоритмов) описаны в многочисленных научных публикациях /25-27/, и другие. В настоящем разделе описаны некоторые алгоритмы данного класса, широко используемые в практическом распознавании.

Принципиальная идея данных алгоритмов основана на отнесении распознаваемого объекта *S* в тот класс, в котором имеется большее число «информативных» фрагментов

эталонных объектов («частичных прецедентов»), приблизительно равных соответствующим фрагментам объекта *S*. Вычисляются близости – «голоса» (равные 1 или 0) распознаваемого объекта к эталонам некоторого класса по различным информативным фрагментам объектов класса. Данные близости («голоса») суммируются и нормируются на число эталонов класса. В результате вычисляется нормированное число голосов, или оценка объекта *S* за класс $\Gamma_j(S)$ – эвристическая степень близости объекта *S* к классу K_j. После вычисления оценок объекта за каждый из классов, осуществляется отнесение объекта к одному из классов с помощью порогового решающего правила.

1.3.1. Тестовый алгоритм.

Тестовый алгоритм распознавания был опубликован в /15/ и базируется на понятии теста, введенного в 1956 году С.В.Яблонским. В классическом изложении тестовый алгоритм был предназначен для бинарных и к-значных признаков. Пусть $x_i(S) \in \{0,1,...,k-1\}, i = 1,2,...,n.$

Определение 1. Тестом таблицы T_{nml} называется совокупность столбцов $\{i_1, i_2,..., i_k\}$ таких, что после удаления из T_{nml} всех столбцов, за исключением имеющих номера $\{i_1, i_2,..., i_k\}$, в полученной таблице $T_{n-k,m,l}$ все пары строк, принадлежащих разным классам, различны. Тест $\{i_1, i_2,..., i_k\}$ называется тупиковым, если никакая его истинная часть не является тестом /59/.

Пусть найдено множество $\{T\}$ тупиковых тестов таблицы T_{nml} и $T = \{i_1, i_2,..., i_k\} \in \{T\}$. Выделим в описании распознаваемого объекта *S* фрагмент описания $(a_{i_1}, a_{i_2},..., a_{i_k})$, соответствующий признакам с номерами $i_1, i_2,..., i_k$. Сравним $(a_{i_1}, a_{i_2},..., a_{i_k})$ со всеми фрагментами $(a_{ti_1}, a_{ti_2},..., a_{ti_k})$ объектов $S_t, t = 1,2,...,m,$ таблицы T_{nml}.

Число совпадений $(a_{i_1}, a_{i_2},..., a_{i_k})$ с $(a_{ti_1}, a_{ti_2},..., a_{ti_k})$, $t = m_{j-1}+1, m_{j-1}+2,..., m_j$, для каждого $j = 1,2,...,l$, $m_0 = 0, m_l = m$, обозначим через $G_j(T)$.

Величина $\Gamma_j(S) = \dfrac{1}{m_j - m_{j-1}} \sum_{T \in \{T\}} G_j(T)$ (1.12)

называется оценкой *S* по классу K_j. Имея оценки $\Gamma_1(T), \Gamma_2(T),..., \Gamma_l(T)$ объект *S* легко классифицируется с помощью решающих правил. Например, он относится в тот класс, за который получена его максимальная оценка. В случае наличия нескольких классов с максимальными оценками, происходит отказ от его классификации.

Если отдельное совпадение в (1.12) назвать «голосом» в пользу класса K_j, то выражение (1.12) будет нормированным числом голосов за данный класс. В связи с данной интерпретацией, алгоритмы с подобной схемой вычисления оценок называют также алгоритмами голосования.

Тупиковые тесты широко используются для оценки информативности (важности) признаков. Пусть N_i - число тупиковых тестов таблицы T_{nml}, содержащих *i*-й столбец , а $N = |\{T\}|$ - общее число тупиковых тестов таблицы. Тогда вес *i*-го признака определяется как $p_i = \frac{N_i}{N}$. Чем больше вес, тем важнее признак для описания объектов. Это утверждение основывается на следующем рассуждении. Таблица T_{nml} является исходным описанием классов и признаков. Тупиковый тест есть несжимаемое далее по признакам описание классов, сохраняющее разделение классов. Чем в большее число таких неизбыточных описаний входит *i*-й признак, тем он важнее.

Задача нахождения множества тупиковых тестов таблицы T_{nml} сводится к вычислению всех тупиковых покрытий строк некоторой бинарной матрицы ее столбцами. Пусть $T_{nml} = \|a_{ij}\|_{m\times n}$, где $a_{ij} = x_j(S_i) \in \{0,1,...,k-1\}$. Таблице T_{nml} ставится в соответствие матрица сравнения по признакам всевозможных пар объектов из различных классов $C = \|c_{ij}\|_{N\times n}$, где $c_{ij} = \begin{cases} 1, & a_{vj} \neq a_{\mu j}, \\ 0, & a_{vj} = a_{\mu j}, \end{cases} i = i(v,\mu)$, $S_v \in K_u, S_\mu \in K_v, u \neq v$,

$N = \sum\limits_{\substack{i>j \\ i,j=1,2,...,l}} (m_i - m_{i-1})(m_j - m_{j-1})$.

Столбцы $(i_1, i_2, \ldots, i_k)$ образуют покрытие строк матрицы $C = \|c_{ij}\|_{N\times n}$, если $\forall i = 1,2,...,N, \quad \exists j \in \{i_1, i_2, ..., i_k\}: \quad c_{ij} = 1$. Покрытие называется тупиковым, если произвольное его собственное подмножество не является покрытием. Каждому тупиковому тесту соответствует тупиковое покрытие строк матрицы сравнения и наоборот. Нахождение всех тупиковых тестов является сложной вычислительной задачей. Тем не менее здесь существуют подходы, эффективные для таблиц средней размерности /21/. При решении практических задач достаточно нахождение лишь части тупиковых тестов для вычисления оценок согласно (1.12). В случае наличия вещественнозначных признаков используют обычно один из следующих двух подходов. Область определения каждого вещественнозначного признака разбивается на *k* интервалов. Значение признака

из *i*- го интервала полагается равным *i-1*. Далее задача распознавания решается относительно полученной *k*- значной таблицы обучения и *k*- значных описаний распознаваемых объектов. Другой подход связан с введением числовых пороговых параметров для каждого признака. Для веществennозначных таблиц вводится аналог тупиковых тестов, в котором требование несовпадения значений признаков заменяется требованием их различия не менее чем на соответствующий порог. В обоих подходах при выборе значности новой таблицы или значений пороговых параметров необходимо учитывать, что соответствующие матрицы сравнения не должны содержать нулевых строк, поэтому значность (или пороговые параметры) следует выбирать из требования отделимости *k*- значных «образов» эталонных объектов различных классов (или отделимости с точностью до пороговых значений).

1.3.2. Алгоритмы распознавания с представительными наборами.

Другими алгоритмами данного вида являются алгоритмы типа “Кора” /3, 10/, в которых опорные множества связаны со значениями признаков конкретных объектов.

Пусть $S_\nu \in K_j$, а признаки принимают значения 0 или 1. Набор $u = \{x_{i_1}(S_\nu), x_{i_2}(S_\nu), ..., x_{i_k}(S_\nu),\}$ назовем представительным набором для класса K_j, если для любого $S_\mu \in T_{nml}, S_\mu \notin K_j$ хотя бы одно из равенств $x_{i_t}(S_\nu) = x_{i_t}(S_\mu), t = 1,2,...,k,$ не выполнено. Представительный набор называется тупиковым, если любой его собственный поднабор не является представительным набором, т.е. для любого его поднабора существует равный ему поднабор в каком-либо другом классе. Таким образом, тупиковый представительный набор некоторого класса K_j есть несократимый фрагмент некоторого эталона данного класса, для которого нет равных ему фрагментов эталонов других классов.

Пусть вычислено V_j −множество тупиковых представительных наборов для класса K_j. Тогда степень близости распознаваемого объекта S к классу K_j по заданному множеству тупиковых представительных наборов V_j можно оценить по формуле

$$\Gamma_j(S) = \frac{1}{|V_j|} \sum_{u \in V_j} B(S,u) \qquad (1.13)\ ,$$

где $B(S,u)$ равно 1, если в признаковом описании объекта *S* имеется фрагмент *u*. Далее классификация объекта осуществляется по максимальной из вычисленных оценок, как и в

тестовом алгоритме. Таким образом, в данном алгоритме с представительными наборами распознаваемый объект относится в тот класс, для которого он имеет максимальную часть тупиковых представительных наборов в своем признаковом описании. Отметим, что в модели с представительными наборами (как и для тестового алгоритма) возможны и другие способы вычисления оценок (1.13). Например, другие нормировки, весовые коэффициенты перед $B(S,u)$, значения которых равны числу эталонов класса K_j, в которых встречается *u,* или значения которых находятся в результате оптимизации модели распознавания (см. 1.3.4) .

Для нахождения тупиковых представительных наборов, содержащихся в некотором эталоне S_j, формируются матрицы сравнения S_j со всеми эталонами других классов. Тупиковые покрытия данных матриц сравнения и определяют тупиковые представительные наборы. Однако, если для поиска тупиковых тестов таблицы T_{nml} требуется решать задачу на покрытия для матрицы из $N = \sum_{\substack{i>j \\ i,j=1,2,...l}} (m_i - m_{i-1})(m_j - m_{j-1})$ строк и *n* столбцов, то для поиска тупиковых представительных наборов объекта $S_i \in K_j$ задача на покрытия решается для матрицы из $(m - m_j)$ строк и *n* столбцов. Таким образом, нахождение множества всех тупиковых представительных наборов таблицы T_{nml} требует решения *m* задач на покрытия но «малой » размерности. Эффективные алгоритмы решения данных задач разработаны Дюковой /22/.

Вопрос обобщения алгоритмов распознавания с представительными наборами на случаи *k*- значной и вещественнозначной информации информации решается аналогично тестовому алгоритму. Множество допустимых значений некоторого признака делится на конечное число интервалов, каждому из которых приписывается целое число 0, 1, 2,…, или *k-1.* Таблице T_{nml} и распознаваемым объектам ставятся в соответствие строки новых целочисленных значений признаков. Далее процесс определения тупиковых представительных наборов и распознавания полностью идентичен бинарному случаю. Другое распространение на вещественнозначные признаки связано с введением пороговых параметров $\varepsilon_1, \varepsilon_2, ..., \varepsilon_n$ и следующей модификацией понятия представительного набора: набор $u = \{x_{i_1}(S_\nu), x_{i_2}(S_\nu), ..., x_{i_k}(S_\nu),\}$ называется представительным набором для класса K_j, если для любого $S_\mu \in T_{nml}, \cdot S_\mu \notin K_j$ хотя бы одно из неравенств $\left|x_t(S_\nu) - x_t(S_\mu)\right| \le \varepsilon_t, t = i_1, i_2, ..., i_k$ будет невыполненным.

Заметим, что дискретизация признаков или введение пороговых параметров $\varepsilon_1, \varepsilon_2, ..., \varepsilon_n$ здесь могут быть индивидуальны для каждого эталона. Главным требованием их выбора является отделимость рассматриваемого объекта $S_i \in K_j$ относительно эталонов из CK_j.

Отметим, что в отличие от тестового алгоритма, описанная модель с представительными наборами допускает пересечение классов. В последнем случае оказываются «бесполезными» объекты, по которым классы пересекаются, поскольку данные объекты не содержат представительных наборов. Впрочем, существуют модификации понятия представительного набора, допускающие и пересечение классов.

1.3.3. Алгоритмы распознавания, основанные на вычислении оценок.

Идеи распознавания по частичным прецедентам, первоначально заложенные в тестовом алгоритме распознавания, были обобщены в моделях распознавания, основанных на вычислении оценок. Алгоритмы данных моделей определяются заданием шести последовательных этапов, для которых могут быть использованы различные конкретные способы определения или выполнения. Тестовый алгоритм и алгоритмы с представительными наборами могут быть представлены как частные случаи более общей конструкции. Ниже будут приведены лишь некоторые основные способы выполнения данных этапов. Подробно данные вопросы рассмотрены в /25 - 27/.

1. **Задание системы опорных множеств алгоритма.** Первым шагом определения алгоритмов вычисления оценок (АВО) является задание множества подсистем признаков, по которым осуществляется сравнение объектов. Пусть Ω_A - некоторая система подмножеств множества *{1,2,...,n}*, называемая системой опорных множеств алгоритма *A*. Элементы $\Omega=\{i_1, i_2, ..., i_k\} \in \Omega_A$ называются опорными множествами алгоритма. Они определяют номера признаков, по которым сравниваются части эталонных и распознаваемых объектов. Примером выбора системы опорных множеств Ω_A является множество тупиковых тестов. Каждому подмножеству $\Omega=\{i_1, i_2, ..., i_k\}$ можно поставить во взаимно однозначное соответствие характеристический булевский вектор $\omega = \{\omega_1, \omega_2, ..., \omega_n\}$, в котором $\omega_j = 1,\ j = i_1, i_2, ..., i_k$, а остальные компоненты равны нулю. В силу данного соответствия $\Omega \leftrightarrow \omega$, использование данных величин будет для нас равнозначным.

Множество всех n-мерных булевских векторов определяет дискретный единичный куб $E^n = \{\omega : \omega = (\omega_1, \omega_2, ..., \omega_n)\}$, $\omega_i \in \{0,1\}, i = 1,2,...,n$. Число элементов куба равно 2^n.

Теоретические исследования свойств тупиковых тестов для случайных бинарных таблиц показали, что характеристические векторы «почти всех тупиковых тестов» имеют асимптотически (при неограниченном возрастании размерности таблицы обучения) приблизительно одну и ту же длину. Это явилось одним из обоснований выбора в качестве множества Ω_A всевозможных подмножеств *{1,2,…,n}* длины *k*. Значение *k* находится из решения задачи обучения (оптимизации модели) или задается экспертом. В итоге, широко распространенными подходами к выбору Ω_A являются (наряду с тупиковыми тестами) следующие два:

a) $\Omega_A = \{\Omega : |\Omega| = k\}$;

b) $\Omega_A = \{\Omega\}, \Omega \subseteq \{1,2,...,n\}, \Omega \neq \varnothing.$

Второй способ выбора системы опорных множеств, как всевозможных подсистем *{1,2,…,n},* не требует нахождения подходящего значения параметра *k*.

2. Задание функции близости. Пусть фиксировано некоторое опорное множество Ω и соответствующий ему характеристический вектор ω. Фрагмент $x_{i_1}(S), x_{i_2}(S), ...x_{i_k}(S)$ объекта $S = (x_1(S), x_2(S), ...x_n(S)$, соответствующий всем единичным компонентам вектора ω, называется ω-частью объекта, и обозначается ω*S*. Под функцией близости $B_\Omega(S_i, S_j)$ будет пониматься функция от соответствующих ω-частей сравниваемых объектов, принимающая значение 1 («объекты близки») или 0 («объекты далеки»). Приведем примеры подобных функций.

a) $B_\Omega(S_\nu, S_\mu) = \begin{cases} 1, & |x_i(S_\nu) - x_i(S_\mu)| \le \varepsilon_i, \forall i : \omega_i = 1, \omega \leftrightarrow \Omega, \\ 0, & \text{иначе.} \end{cases}$

Здесь $(\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)$ - неотрицательные параметры, именуемые «точности измерения признаков».

b) $B_\Omega(S_\nu, S_\mu) = \begin{cases} 1, & \sum\limits_{i \in \Omega} |x_i(S_\nu) - x_i(S_\mu)| \le \varepsilon, \\ 0, & \text{иначе.} \end{cases}$

Здесь ε также некоторый неотрицательный параметр алгоритма.

3. Оценка близости объекта *S* к эталонному объекту S_i для заданной ω-части. Данная числовая величина формируется на основе функции близости и, возможно, дополнительных параметров.

a) $\Gamma_\Omega(S_i, S) = B_\Omega(S_i, S)$.

b) $\Gamma_\Omega(S_i, S) = w_\Omega B_\Omega(S_i, S)$, где w_Ω - «вес» опорного множества.

c) $\Gamma_\Omega(S_i, S) = \gamma_i(\sum_{j:\omega_j=1} p_j) B_\Omega(S_i, S)$. Здесь γ_i - параметры, характеризующие степень важности объекта S_i (информативность объекта), а $p_1, p_2, \ldots p_n$ - веса (информативность) признаков.

4. Оценка объекта *S* за класс K_j для заданной ω-части.

а) Функция $\Gamma_j^\Omega(S) = \frac{1}{(m_j - m_{j-1})} \sum_{S_i \in K_j} \Gamma_\Omega(S_i, S)$ является примером естественной оценки близости объекта к классу для заданного подмножества признаков.

5. Оценка объекта *S* за класс K_j .

Данная функция задает суммарную степень близости распознаваемого объекта *S* к классу K_j . Приведем обычно используемые выражения для ее вычисления.

a) $\Gamma_j(S) = \sum_{\Omega \in \Omega_A} \Gamma_j^\Omega(S)$.

b) $\Gamma_j(S) = \nu_j \sum_{\Omega \in \Omega_A} \Gamma_j^\Omega(S)$, где ν_j - «вес» класса K_j .

Например, в статистической теории распознавания аналогами параметров ν_j являются априорные вероятности классов, которые характеризуют, насколько часто встречаются объекты различных классов .

6. Решающее правило.

Решающее правило есть правило (алгоритм, оператор), относящее объект по вектору оценок $(\Gamma_1(S), \Gamma_2(S), \ldots, \Gamma_l(S))$ в один из классов, или вырабатывающее для объекта «отказ от распознавания». Отказ является более предпочтительным вариантом решения в случаях, когда оценки объекта малы за все классы (объект является принципиально новым, аналоги которого отсутствуют в обучающей выборке), или он имеет две или более близкие максимальные оценки за различные классы (объект лежит на границе классов).

В формальной постановке, решающее правило ***r*** вычисляет для распознаваемого объекта *S* по вектору оценок $\Gamma(S) = (\Gamma_1(S), \Gamma_2(S), \ldots, \Gamma_l(S))$ вектор

$r(\Gamma(S)) = (\alpha_1^A(S), \alpha_2^A(S), ..., \alpha_l^A(S)), \alpha_i(S) \in \{0,1,\Delta\}, i = 1,2,...,l.$ Интерпретация обозначений приведена ранее в (1.1).

a) Пример простейшего решающего правила – отнесение объекта в единственный класс, за который он имеет максимальную оценку.

$$\alpha_j^A(S) = \begin{cases} 1, & \Gamma_j(S) > \Gamma_i(S), i = 1,2,...,l, i \neq j, \\ 0, & \text{иначе.} \end{cases}$$

b) Для «осторожного» принятия решения относительно новых объектов, существенно непохожих на объекты обучающей выборки, или находящихся на границе двух и более классов, обычно достаточно введение в решающее правило двух пороговых параметров

$$\alpha_j^A(S) = \begin{cases} & \Gamma_j(S) > \Gamma_i(S) + \delta_1, i = 1,2,...,l, i \neq j, \\ 1, & \Gamma_j(S) > \delta_2 \sum_{i=1}^{l} \Gamma_i(S), \\ 0, & \text{иначе.} \end{cases}$$

Здесь $\delta_1, \delta_2 \geq 0$.

c) Линейное решающее правило в пространстве R^l оценок определяется как

$$\alpha_i^A(S) = \begin{cases} 1, & \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) \geq c_1^i, \\ \Delta, & c_1^i > \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) > c_2^i, \\ 0, & \sum_{j=1}^{l} \delta_j^i \Gamma_j(S) \leq c_2^i, \end{cases} \qquad (1.14)$$

Здесь $\delta_1, \delta_2, \delta_j^i, c_1^i, c_2^i$ - параметры алгоритма. В решающем правиле (1.14) наличие двух или более единиц интерпретируется как «объект принадлежит нескольким классам». Когда вектор $(\alpha_1^A(S), \alpha_2^A(S), ..., \alpha_l^A(S))$ состоит из одних нулей говорят, что данный объект – выброс, он не похож ни на один из классов, т.е. близких ему аналогов ранее не наблюдалось.

Использование решающего правила означает фактически переход из признакового пространства в пространство оценок, в котором в качестве разделяющих классы функций используются гиперплоскости, проходящие через начало координат симметрично

относительно новых координатных осей (случай **a**), пары гиперплоскостей (случай **b**), и наборы из l гиперплоскостей.

Варьируя в моделях вычисления оценок правила определения этапов 1-6 (часть примеров их определения приведена выше), можно получить различные модели распознающих алгоритмов типа вычисления оценок.

Если конкретные правила этапов (1-5) определены, то после последовательной подстановки выражений на этапах 2-5 могут быть получены различные общие формулы для вычисления оценок $\Gamma_j(S)$. Выбирая первые варианты реализации этапов (2-5) будет получена следующая общая формула для вычисления оценок объекта S за классы K_j , $j=1,2,...,l$.

$$\Gamma_j(S) = \frac{1}{(m_j - m_{j-1})} \sum_{\nu=m_{j-1}+1}^{m_j} \sum_{\Omega \in \Omega_A} B_\Omega(S_\nu, S) . \qquad (1.15)$$

При выборе системы опорных множеств согласно вариантам a) или b) прямое вычисление оценок (1.15) представляется весьма трудоемким (при вычислении оценок (1.15) согласно a) требуется mC_n^k вычислений значений функции близости). В действительности нет необходимости выполнения всех данных вычислений, поскольку при многих вариантах реализации этапов 2-5 и различных системах опорных множеств существуют эффективные комбинаторные формулы вычисления оценок.

Например, при использовании в качестве системы опорных множеств $\Omega_A = \{\Omega : |\Omega| = k\}$ и вариантов a) выполнения этапов (2-5), справедлива формула

$$\Gamma_j(S) = \frac{1}{(m_j - m_{j-1})} \sum_{S_i \in K_j} C^k_{d(S,S_i)} , \qquad (1.16)$$

где $d(S,S_i) = \left| \left\{ \nu : |x_\nu(S_i) - x_\nu(S)| \le \varepsilon_\nu, \nu = 1,2,...,n \right\} \right|$.

При использовании вариантов a) выполнения этапов (2-5) и b) для первого этапа, справедлива формула

$$\Gamma_j(S) = \frac{1}{(m_j - m_{j-1})} \sum_{S_i \in K_j} (2^{d(S,S_i)} - 1) . \qquad (1.17)$$

Другие, более сложные и общие способы определения этапов (1-5), а также соответствующие им эффективные формулы вычисления оценок, приведены в /25, 26/.

1.3.4. Оптимизация многопараметрических моделей распознавания.

Процесс распознавания во многих моделях вычисления оценок предполагает знание числовых параметров модели (веса признаков, веса эталонов, пороговые параметры, и т.п.). Их значения могут быть выбраны непосредственно пользователем исходя из содержательных или эвристических соображений, поскольку многие параметры имеют естественную интерпретацию. Основным же подходом к их вычислению является процесс обучения или оптимизации модели. Желаемым результатом в обоих случаях является нахождение таких значений параметров, при которых будет обеспечена высокая точность распознавания.

Поиск значений параметров, как процесс «обучения с учителем», используется в нейросетевых подходах, методе потенциальных функций, построении линейных разделяющих гиперплоскостей. Применяется следующая общая схема обучения. Задаются начальные значения параметров (например, случайные из некоторого интервала). Алгоритму предъявляется один из обучающих объектов, класс которого известен. Если объект распознается правильно, предъявляется для распознавания следующий объект. Если объект классифицируется неправильно, происходит коррекция параметров «в нужном направлении». Процесс продолжается до достижения стабилизации работы алгоритма, когда последующее обучение не уменьшает общее число ошибок на обучающей выборке.

Более общая постановка процесса «настройки» алгоритмов связана с решением стандартной оптимизационной задачи оптимизации модели.

Пусть дано параметрическое множество распознающих алгоритмов $\{A(y\}, y \in D\}$ и на нем определен числовой функционал $\varphi(A)$ качества алгоритма. Требуется найти такой алгоритм $A^* \in \{A\}$, который доставляет экстремум функционалу: $\varphi(A^*) = \underset{A\in\{A\}}{extr}\,\varphi(A)$.

Так, например, модель вычислении оценок со способами выполнения этапов (a,a,c,a,a,c) является следующим параметрическим семейством алгоритмов:

$$\{A_k(k,\varepsilon,p,\gamma,\delta,c_1,c_2),1 \geq k \geq 0, k - \text{целое}, \varepsilon \geq 0,1 \geq p \geq 0,1 \geq \gamma \geq 0, c_1 > c_2\}$$

Стандартная постановка проблема оптимизации параметрической модели распознавания состоит в следующем.

Пусть задана таблица контрольных объектов T'_{nql}, аналогичная таблице обучения, т.е. состоящая из разбитых на *l* классов *m* числовых строк – признаковых описаний объектов

$$S'_i = (x_1(S'_i), x_2(S'_i), \ldots, x_n(S'_i)),\ a'_{it} = x_t(S'_i) \qquad (1.18)$$

Для определенности считаем, что

$$S'_i \in K_j, i = q_{j-1}+1, q_{j-1}+2, \ldots, q_j, q_0 = 0, q_l = q.$$

Пусть $\alpha_{ij} = \begin{cases} 1, & S'_i \in K_j, \\ 0, & S'_i \notin K_j. \end{cases}$ Обозначим $\alpha_{ij}^A = \alpha_j(S'_i)$.

Определение 2. Стандартным функционалом качества распознавания называется функционал $\varphi(A) = \frac{1}{ql}\sum_{i=1}^{q}\sum_{j=1}^{l}\left|\alpha_{ij} - \alpha_{ij}^A\right|$.

В статистической теории распознавания данный критерий называют эмпирическим риском. Очевидными эквивалентными ему вариантами являются «доля правильных ответов» или «число правильных ответов».

Постановка задачи оптимизации моделей распознавания может быть записана в терминах систем неравенств. Для простоты ограничимся случаем двух классов и моделью (a,a,c,a,a,a) вычисления оценок.

Условием правильного распознавания некоторого контрольного объекта $S'_i \in K_j$ является выполнение неравенства $\Gamma_1(S'_i) > \Gamma_2(S'_i)$, если объект из первого класса, и $\Gamma_2(S'_i) > \Gamma_1(S'_i)$, если объект из второго класса. Тогда число правильно распознанных объектов при некотором варианте выбора параметров модели будет равно числу выполненных неравенств системы.

$\Gamma_1(S'_i) > \Gamma_2(S'_i)$, $i = 1,2,\ldots,m_1$,

$\Gamma_2(S'_i) > \Gamma_1(S'_i)$, $i = m_1+1, m_1+2, \ldots, m$.

Учитывая, что оценки являются билинейными формами от параметров $p_1, p_2, \ldots p_n$ и $\gamma_1, \gamma_2, \ldots \gamma_m$, задача оптимизации модели может быть сформулирована следующим образом: «Найти максимальную совместную подсистему системы (1.19) и некоторое ее решение, удовлетворяющие соответствующим ограничениям на параметры модели».

$$\sum_{j=1}^{m}\sum_{i=1}^{n} b_{ij}^t(k,\varepsilon) p_i \gamma_j > 0,\ t = 1,2,\ldots,q. \qquad (1.19)$$

Данная задача является сложной оптимизационной задачей даже для частного случая линейной системы, когда в (1.19) фиксированы или параметры $k, \varepsilon_1, \varepsilon_2, \ldots \varepsilon_n, p_1, p_2, \ldots p_n$, или параметры $k, \varepsilon_1, \varepsilon_2, \ldots \varepsilon_n, \gamma_1, \gamma_2, \ldots \gamma_m$.

Фундаментальные теоретические результаты, связанные с исследованием задачи поиска максимальных совместных подсистем, получены в Уральском Университете (Мазуров, Хачай) /43/. Комбинаторные алгоритмы для задач малой размерности созданы в ВЦ РАН Катериночкиной Н.Н. /34/ В системе «РАСПОЗНАВАНИЕ» используется эвристический алгоритм, основанный на релаксационном спуске. Оптимизация стандартного функционала качества как последовательность вспомогательных оптимизационных задач в пространстве параметров ε при фиксированных p,γ и пространстве p,γ, при фиксированных ε, рассматривалась в /82/.

1.3.5. Статистическое взвешенное голосование.

Процедура статистически взвешенного голосования по системам подобластей признакового пространства лежит в основе метода распознавания образов «Статистически взвешенные синдромы» (СВС). Процедура статистически взвешенного голосования используется при прогнозировании произвольных стохастических функций, зависящих от набора непрерывных прогностических переменных и принимающих значения из некоторого подмножества точек действительной оси.

Пусть Y - прогнозируемая функция, $\tilde{Q}$ - система подобластей многомерного признакового пространства, $\tilde{S}_y$ -обучающая выборка, представляющая собой множество объектов вида $(y_j, \mathbf{x}_j)$, где y_j - значение функции Y , а $\mathbf{x}_j$ - соответствующий вектор значений прогностических переменных. Процедура статистически взвешенного голосования предназначена для построения по системе подобластей $\tilde{Q}$ и обучающей выборке $\tilde{S}_y$ детерминировано зависящей от прогностических переменных функции $\tilde{Y}$, которая достаточно сильно коррелирована с Y. Назовем данную функцию $\tilde{Y}$ взвешенной оценкой функции Y. Пусть $\mathbf{x}$ -некоторая точка в многомерном признаковом пространстве, принадлежащая к подобластям $q_1,\ldots,q_p$ из $\tilde{Q}$. Значение функции $\tilde{Y}$ в точке $\mathbf{x}$ вычисляется по формуле (1.20)

$$\tilde{Y}(\mathbf{x}) = \frac{\sum_{i=1}^{p} w_i \bar{y}_i}{\sum_{i=1}^{p} w_i}, \qquad (1.20)$$

где $\bar{y}_i$ - среднее значение функции Y на объектах обучающей выборки из подобласти q_i, w_i -так называемый вес i-ой подобласти. Метод вычисления весов подобластей, основанный на максимизации специального функционала правдоподобия, был предложен

в работе /56/. В результате была получена следующая формула для расчета весов: $w_i = \frac{1}{d_i}\frac{m_i}{m_i\kappa_i + 1}$, где m_i -число объектов из $\tilde{S}_y$ с описаниями $\mathbf{x}$, попавшими в подобласть q_i; d_i - дисперсия функции Y в подобласти q_i; κ_i - коэффициент детерминированности функции Y на подобласти q_i, который определяется как отношение $\kappa_i = \frac{d_i^{\xi}}{d_i}$, где $d_i^{\xi} = \int_{Q_i} \{\xi[\mathbf{x}(\omega)] - \mathbf{M}(Y \mid Q_i)\}^2 \mathbf{P}(d\omega)$ - дисперсия на подобласти q_i функции $\xi(\mathbf{x})$, которая определяется равенством $\xi(\mathbf{x}) = \mathbf{M}(Y \mid \mathbf{x})$. Коэффициент детерминированности возрастает с уменьшением доли случайной составляющей в зависимости функции Y от $\mathbf{x}$ внутри подобласти q_i.

В методе СВС в качестве оценок за классы $K_1,...,K_l$ для некоторого распознаваемого объекта S выступают рассчитанные с помощью процедуры взвешенного голосования взвешенные оценки индикаторных функций классов $\alpha_1(S),...,\alpha_l(S)$. При этом в качестве подобластей голосования используются так называемые синдромы.

Под синдромом мы в данном случае понимаем подобласть в пространстве прогностических признаков, внутри которой содержание объектов одного из классов значительно отличается от среднего содержания по выборке.

1.4. Алгебраический подход для решения задач распознавания и прогноза

1.4.1. Этапы развития теории распознавания и классификации по прецедентам.

Анализ истории развития теории распознавания и исследование существующих подходов позволяют выделить три основных этапа в ее развитии.

1. Первый этап характеризуется появлением разнообразных эвристических методов и алгоритмов как универсальных, предназначенных для решения широкого спектра задач, так и специальных, ориентированных на обработку информации заданного типа. С их помощью решались прикладные задачи в самых различных областях человеческой деятельности. Примерами успешных частных алгоритмов являются алгоритмы «ближайший сосед», «тестовый алгоритм», «алгоритм Кора», дискриминант Фишера, и многие другие.

2. Второй этап связывают с разработкой на базе отдельных эвристических методов распознавания параметрических моделей и решением задач оптимизации моделей - поиска наилучших алгоритмов в пределах фиксированных моделей /25/. Данный этап связан с естественным желанием исследователя в максимально возможном обобщении

метода распознавания, реализацией идеи параметрической подстройки метода под новые данные. К тому же обилие алгоритмов распознавания создало проблему их сравнения и автоматического выбора наилучшего из заданного множества. Приведенные в разделах 1.2, 1.3 основные подходы и представляют современные и достаточно хорошо изученные модели.

В дальнейшем, опыт решения практических задач выявил и слабые места при работе с фиксированной моделью. Выяснилось, что нахождение и использование наилучшего по точности алгоритма данной модели для заданной контрольной задачи далеко не всегда является лучшим способом использования имеющегося исходного множества алгоритмов. Стандартной является ситуация, когда найден набор эвристических алгоритмов $A_1, A_2, \ldots, A_k$ примерно равного качества, но правильно распознающих различные подмножества контрольной выборки. В то же время в рамках данной модели не существует алгоритмов, превосходящих $A_1, A_2, \ldots, A_k$ по точности распознавания. Таким образом, «назревала» необходимость создания формализаций, объединяющих все известные методы и модели, а не основанные лишь на каком-то одном принципе. Требовалось создание математического аппарата, позволяющего не просто выбирать и применять при решении прикладной задачи какой-либо алгоритм из имеющегося множества, а строить на базе известных новые более предпочтительные алгоритмы.

В 1976-1978 годах Журавлевым был разработан подобный алгебраический формализм. Было предложено решать задачи распознавания не одним, а множеством алгоритмов в два этапа. Сначала для распознавания произвольных объектов независимо применяются алгоритмы $A_1, A_2, \ldots, A_k$. Далее результаты их применения специальным образом обрабатываются и формируется окончательное коллективное решение об отнесении объектов к одному из классов. С данным подходом связана надежда на взаимную компенсацию ошибок отдельных алгоритмов и получение в итоге более точного решения заданной задачи распознавания.

Данная идея склеивания алгоритмов и методы ее реализации лежат в основе алгебраической теории распознавания. Было показано, что все многообразие распознающих алгоритмов может быть описано в стандартном единообразном виде. Каждый алгоритм может быть представлен как произведение (последовательное выполнение) двух операторов - распознающего оператора и решающего правила. Распознающий оператор переводит совокупность q описаний объектов распознаваемой выборки в числовую матрицу оценок $\|g_{ij}\|, i = 1,2,\ldots,q, j = 1,2,\ldots,l,$ характеризующих

меры близости объектов к каждому из классов. Решающее правило переводит числовую матрицу оценок в информационную матрицу вычисленных значений предикатов $P_j(S) = "S \in K_j"$. Показана определяющая роль распознающего оператора. Над множествами распознающих операторов определены алгебраические операции, позволяющие строить корректные алгоритмы (безошибочно распознающие элементы заданной контрольной выборки), получены необходимые и достаточные условия существования корректных алгоритмов, разработаны схемы и численные методы их непосредственного построения в виде полиномов от отдельных алгоритмов одной или нескольких моделей /25-28/.

1.4.2. Алгебраическая коррекция множеств распознающих алгоритмов.

Предположим, что у нас имеется семейство $\{A\}$ эвристических алгоритмов распознавания, которые вообще говоря не являются корректными для некоторой конкретной задачи. Под корректностью распознающего алгоритма понимается правильность распознавания им объектов контрольной выборки. Возникает вопрос о возможности построения корректного алгоритма с использованием уже имеющегося множества алгоритмов $\{A\}$.

Рассмотрим более общую постановку задачи распознавания с пересекающимися классами. Пусть имеется некоторое множество допустимых объектов $\{S\}$, являющееся объединением классов $K_1,...,K_l$. Через $P_i(S)$ обозначим заданный на множестве допустимых объектов $\{S\}$ предикат $"S \in K_i"$. Пусть задан конечный набор допустимых объектов $S_1,...,S_q$.

Определение 1. Матрица $\|\alpha_{ij}\|_{q \times l}$, где $\alpha_{ij} = P_j(S_i)$ называется информационной матрицей набора $S_1,...,S_q$ по системе предикатов $P_1,...,P_l$. Строка $(\alpha_{i1},...,\alpha_{il})$ называется информационным вектором объекта S_i.

Задача распознавания состоит в том, чтобы по начальной информации I, принадлежащей к множеству допустимых информаций $\{I\}$ о классах $K_1,...,K_l$, и предъявленной для распознавания выборке $\tilde{S}^q = \{S_1, S_2,..., S_q\}$ построить информационную матрицу $\|\alpha_{ij}\|_{q \times l}$. Обозначим данную задачу для краткости как задача $Z(I, S_1,...,S_q, P_1,..., P_l)$ или просто *Z*. Примером начальной информации о классах является таблица признаковых описаний эталонных объектов классов и их информационная матрица.

Предположим, что у нас имеется множество алгоритмов $\{A\}$, переводящих пару $(I, S_1,...,S_q)$, $I \in \{I\}$, $\{S_1,...,S_q\} \subseteq \{S\}$ в матрицы $\|\beta_{ij}\|_{q\times l}$, составленные из элементов $0,1,\Delta$. $A(I,S_1,...,S_q) = \|\beta_{ij}\|_{q\times l}$, где значения β_{ij} интерпретируются обычным образом. Если $\beta_{ij} \in \{0,1\}$, то β_{ij} - значение предиката P_j на допустимом объекте S_i, вычисленное алгоритмом A. Если алгоритм A не вычислил значение предиката $P_j(S_i)$, то принимается $\beta_{ij} = \Delta$ □ □.

Определение 2. Алгоритм A называется корректным для задачи Z, если выполнено равенство $A(I,S_1,...,S_q,P_1,...,P_l) = \|\alpha_{ij}\|_{q\times l}$.

Алгоритм не являющийся корректным для задачи Z называется некорректным. В качестве $\{A\}$ в дальнейшем рассматривается совокупность алгоритмов, составленная вообще говоря из некорректных алгоритмов.

Основной целью является:

А)- введение алгебраических операций над алгоритмами из $\{A\}$, позволяющих строить алгебраические замыкания $[\{A\}]$ множеств $\{A\}$;

В)- выяснение ограничений для $\{I\},\{S\},\{A\}$, при выполнении которых для любой задачи Z в $[\{A\}]$ существует алгоритм, корректный для Z. В этом случае замыкание $[\{A\}]$ называется корректным относительно $\{Z\}$.

Теорема 1. Каждый алгоритм $A \in \{A\}$ представим как последовательное выполнение алгоритмов B и C, где $B(I,S_1,...,S_q) = \|a_{ij}\|_{q\times l}$, a_{ij}-действительные числа $C(\|a_{ij}\|_{q\times l}) = \|\beta_{ij}\|_{q\times l}$, $\beta_{ij} \in \{0,1,\Delta\}$, $B = B(A), C = C(A)$.

Из теоремы 1 следует, что множество $\{A\}$ порождает множества $\{B\}$ и $\{C\}$. Элементы из $\{B\}$ будем в дальнейшем называть распознающими операторами, или просто операторами, элементы из $\{C\}$- решающими правилами. Числовые матрицы $B(I,S_1,...,S_q) = \|a_{ij}\|_{q\times l}$ называют матрицами оценок объектов за классы, или просто матрицами оценок.

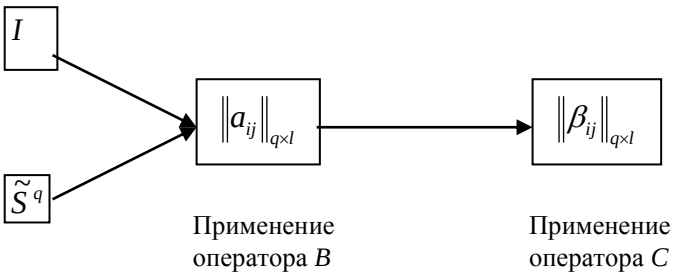

Рис. 14. Каждый алгоритм распознавания представим в виде произведения распознающего оператора и решающего правила

Определение 3. Решающее правило C называется корректным на $\{S\}$, если для всякого конечного набора $S_1,...,S_q$ из $\{S\}$ существует хотя бы одна числовая матрица $\|a_{ij}\|_{q\times l}$ такая, что $C(\|a_{ij}\|_{q\times l}) = \|\alpha_{ij}\|_{q\times l}$. Здесь $\|\alpha_{ij}\|_{q\times l}$ - информационная матрица элементов $S_1,...,S_q$ по системе предикатов $P_1,...,P_l$.

В множестве операторов $\{B\}$ вводятся следующие операции сложения, умножения и умножения на скаляр.

Пусть $B', B'' \in \{B\}$, $B'(I,S_1,...,S_q) = \|a'_{ij}\|_{q\times l}$, $B''(I,S_1,...,S_q) = \|a''_{ij}\|_{q\times l}$, b'-скаляр. Определим операторы $b' \bullet B$ (произведение на скаляр), $B' + B''$ (сумма операторов), $B' \bullet B''$ (произведение операторов) следующим образом:

$$b' \bullet B'(I,S_1,...,S_q) = \| b' \cdot a'_{ij} \|_{q\times l} \qquad (1.21)$$

$$(B' + B'')(I,S_1,...,S_q) = \|a'_{ij} + a''_{ij}\|_{q\times l} \qquad (1.22)$$

$$(B' \bullet B'')(I,S_1,...,S_q) = \|a'_{ij} \cdot a''_{ij}\|_{q\times l} \qquad (1.23)$$

Очевидно, что линейное замыкание $L\{B\}$ множества относительно операций (1.21), (1.22) является векторным пространством.

Замыкание $\mathbf{U}\{B\}$ множества $\{B\}$ относительно операций (1.21) - (1.23) является ассоциативной линейной алгеброй с коммутативным умножением.

Нетрудно увидеть, что операторы из $\mathbf{U}\{B\}$ могут быть представлены в виде многочленов от операторов из $\{B\}$. Если $B' \in \mathbf{U}\{B\}$, то $B' = \sum b_i B_{i1} \bullet ... \bullet B_{ik}$. Как обычно, степенью операторного многочлена называется максимальное число сомножителей в слагаемых $.B_{i1} \bullet ... \bullet B_{ik}$.

Определение 4. Множества $L\{A\}$ и $\mathbf{U}\{B\}$ алгоритмов $A = B \bullet C^*$ таких, что $B \in L\{B\}$ ($B \in \mathbf{U}\{B\}$), называются линейным (алгебраическим) замыканием $\{A\}$.

Зафиксируем информацию $I \in \{I\}$ и выборку $\tilde{S}^q = \{S_1,...,S_q\}$, $\tilde{S}^q \subseteq \{S\}$. Будем рассматривать задачи $(I,\tilde{S}^q)$, обладающие следующим свойством относительно множества $\tilde{\mathbf{B}}$ распознающих операторов.

Определение 5. Если множество матриц $\{B(I,\tilde{S}^q)\}$ (операторы B пробегают множество $\tilde{\mathbf{B}}$) содержит базис в пространстве числовых матриц размерности $q\times l$, то задача $Z(I,S_1,...,S_q,P_1,...,P_l)$ называется полной относительно $\tilde{\mathbf{B}}$.

Теорема 2. Если множество $\{Z\}$ состоит лишь из задач, полных относительно $\tilde{\mathbf{B}}$, то линейное замыкание $L\{\tilde{\mathbf{B}} \bullet C^*\}$, где C^*- произвольное фиксированное корректное решающее правило, является корректным относительно $\{Z\}$.

Следствие 1. Пусть $\{A\}$- совокупность некорректных алгоритмов, $\{B\}$ - соответствующее множество операторов, C^*-фиксированное корректное решающее правило. Тогда $L\{A\} = \{L\{B\} \bullet C^*\}$ является корректным относительно $\{Z\}$, если $\{Z\}$ состоит из задач, полных относительно $\{B\}$.

Следствие 2. Пусть выполнены все условия следствия 1 и, кроме того, $[\{B\}]$ есть замыкание $\{B\}$ относительно операций (1.21) - (1.23). Тогда $\mathbf{U}\{A\} = \{\mathbf{U}\{B\} \bullet C^*\}$ является корректным относительно $\{Z\}$, если $\{Z\}$ состоит из задач, полных относительно $\{B\}$.

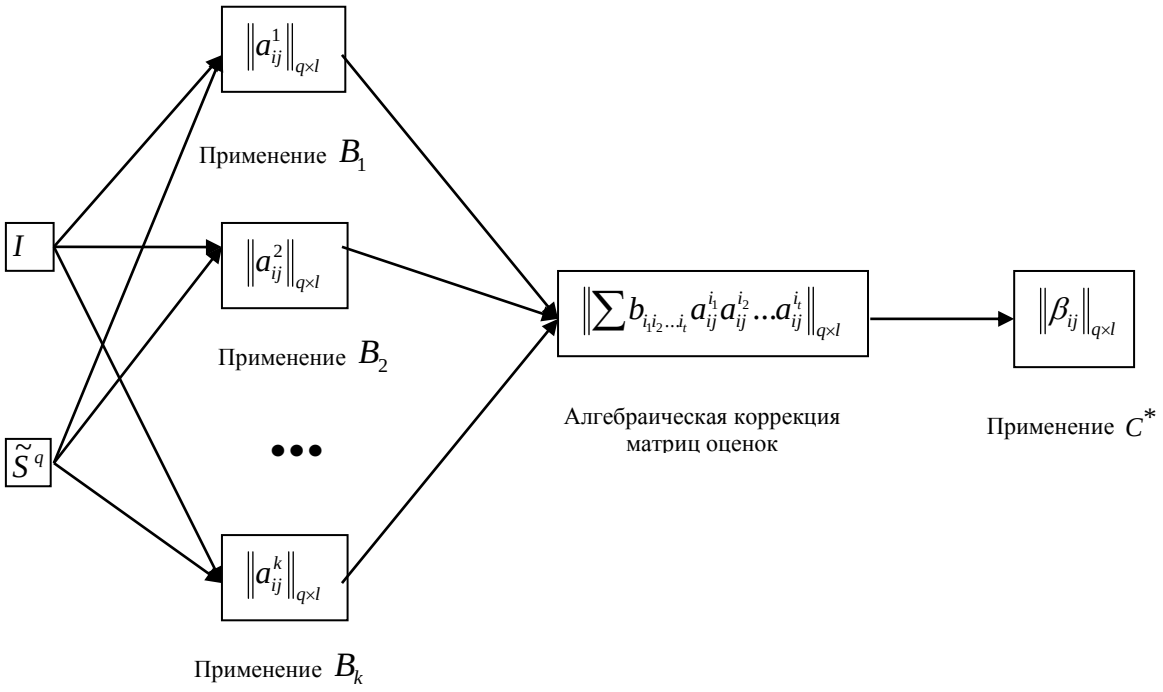

Рис.15. Алгебраическая коррекция эвристических алгоритмов

1.4.3. Логическая коррекция множеств распознающих алгоритмов.

Подобно тому, как было ранее введено сложение распознающих операторов и умножение их на скаляр, можно рассматривать операции над распознающими алгоритмами как операции над информационными матрицами. Однако если множество распознающих операторов образует линейное векторное пространство, то в множествах информационных матриц таких хороших операций нет /26/. Тем не менее использование многоместных операций над множествами информационных матриц позволяет создать средства синтеза эффективных распознающих алгоритмов, непосредственным входом

которых являются результаты независимого распознавания объектов алгоритмами из заданного конечного набора.

Рассмотрим для простоты случай отсутствия отказов при классификации. Пусть задан коллектив из n распознающих алгоритмов $A^1, A^2, ..., A^n$, $A^k(I, S_1, ..., S_q) = \| \beta_{ij}^k \|_{q \times l}$, $k = 1,2,...,n.$ Пусть матрица $\|\alpha_{ij}\|_{q \times l}$ является информационной матрицей набора $S_1, ..., S_q$ по системе предикатов $P_1, ..., P_l$. Множества матриц $\{\| \beta_{ij}^k \|_{q \times l}, k = 1,2,...,n, \| \alpha_{ij} \|_{q \times l}\}$ определяют l не всюду определенных булевых функций $f_j(y_1, y_2, ...y_n), j = 1,2,...,l$. При этом $f_j(\beta_{ij}^1, \beta_{ij}^2, ..., \beta_{ij}^n) = \alpha_{ij}, i = 1,2,...,q$, и функция $f_j(y_1, y_2, ...y_n)$ не определена на остальных $2^n - q$ наборах. Основная задача состоит в таком доопределении функции на весь дискретный единичный куб E^n, при котором будут максимально выполнены дополнительные «содержательно обоснованные» условия, обеспечивающие некоторый однозначный и оптимальный выбор такой функции. Данные функции будем называть логическими корректорами. Если подобные функции построены, то процесс распознавания произвольной новой выборки объектов $\tilde{\tilde{S}}^t = \{S'_1, S'_2, ..., S'_t\}$ алгоритмами $A^1, A^2, ..., A^n$ с последующей логической коррекцией может быть представлен схематически на рис. 16.

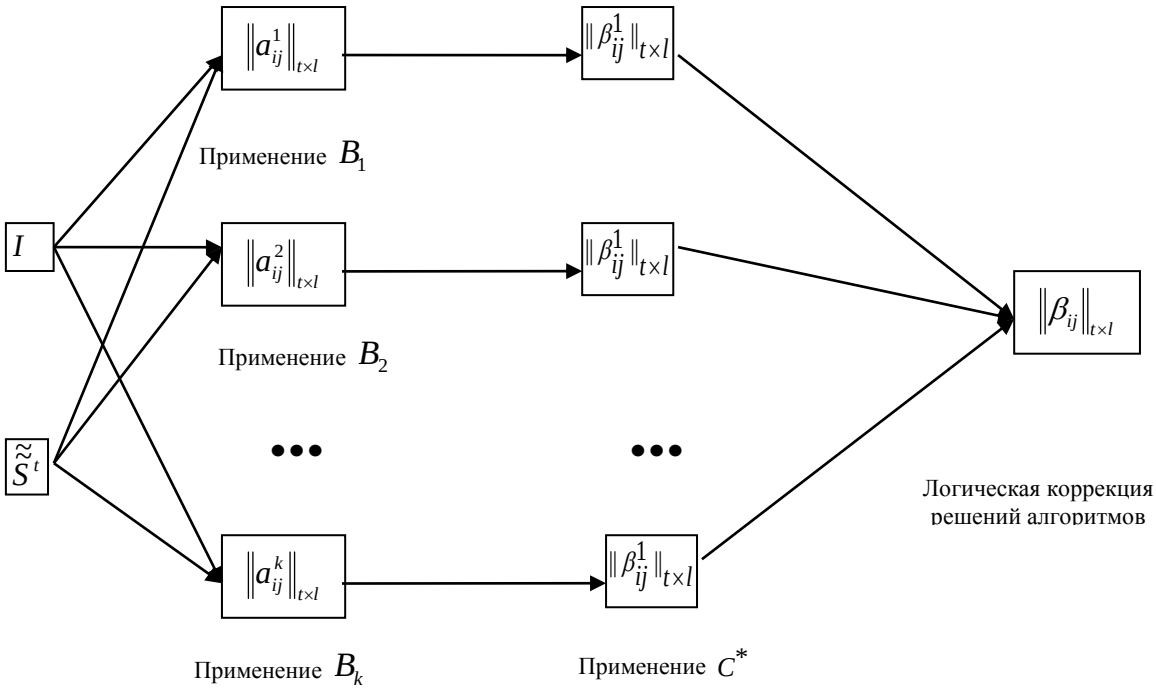

Рис.16. Логическая коррекция алгоритмов распознавания

Существуют разнообразные подходы к выборы типа логических корректоров и методы их поиска. Приведем в качестве примера «монотонный логический корректор». Здесь в основу положена следующая идея. Пусть известно, что при классификации некоторого объекта S, принадлежащего классу K_j, он зачисляется в данный класс алгоритмами $A_{i_1}, A_{i_2},..., A_{i_k}$. Тогда, если некоторый новый объект S' классифицируется данными алгоритмами как принадлежащий K_j, и он относится в K_j дополнительно некоторым алгоритмом $A_t, t \notin \{i_1, i_2,..., i_k\}$, тогда естественно считать коллективной классификаций для S' решение $S' \in K_j$. Математической формализацией данного правила монотонной коррекции алгоритмов будет следующая задача.

Дана частично определенная булева функция $\tilde{f}_j(y_1, y_2,..., y_n)$, причем $\tilde{f}_j(\beta_{ij}^1, \beta_{ij}^2,..., \beta_{ij}^n) = \alpha_{ij}, i = 1,2,..., q$. Требуется найти монотонную булеву функцию $f_j(y_1, y_2,..., y_n)$, для которой выполнено $f_j(\beta_{ij}^1, \beta_{ij}^2,..., \beta_{ij}^n) = \alpha_{ij}, i = 1,2,..., q$. Отметим, что данная задача не имеет решения в случае противоречивости некоторой пары наборов $\{\beta_{ij}^1, \beta_{ij}^2,..., \beta_{ij}^n; \alpha_{ij}\}$ и $\{\beta_{tj}^1, \beta_{tj}^2,..., \beta_{tj}^n; \alpha_{tj}\}$ ($\beta_{ij}^k \le \beta_{tj}^k, k = 1,2,..., n$, но $\alpha_{ij} > \alpha_{tj}$). В данном случае на практике обычно находится монотонная булева функция для непротиворечивого множества наборов по контрольной выборке и применяются эвристики, либо из каждой пары противоречивых наборов исключается «менее представительный».

Различные методы логической коррекции алгоритмов распознавания описаны в работах /32, 36, 60/.

1.4.4. Выпуклый стабилизатор.

Одной из наиболее острых проблем, возникающих при решении практических задач распознавания образов, является «перенастройка» алгоритмов распознавания. Алгоритм настраивается на заданную выборку, демонстрируя на ней высокое качество работы, но стремительно деградирует при предъявлении новых объектов. Для перенастроенных алгоритмов обычно крайне сложно получить оценку качества работы в силу неустойчивости их работы. Перенастройка связана с большим числом параметров, оптимизируемых при обучении, и происходит, как правило, на малых выборках.

При построении коллективных решений, удается повысить устойчивость получаемого алгоритма по сравнению с исходными, и, тем самым, значительно

уменьшить эффект перенастройки. В основе идеи стабилизации алгоритмов лежит понятие неустойчивости алгоритма распознавания на объекте.

Пусть имеется задача распознавания с *l* классами. Каждый объект представлен вектором в *n*-мерном действительном пространстве. Обозначим $\Gamma_i^k(S_j)$ оценку апостериорной вероятности принадлежности *j*-го объекта к *k*-му классу, полученную с помощью *i*-го исходного алгоритма распознавания. Тогда приняв за неустойчивость алгоритма на данном объекте величину

$$Z_B(S_j,\varepsilon)=\sum_{k=1}^{l}\frac{1}{\varepsilon_k^{\,2}}\sum_{i=1}^{n}[\Gamma_B^k(S_j+\varepsilon_k\cdot e_i)-\Gamma_B^k(S_j)]^2,$$

(*B* – распознающий оператор, соответствующий алгоритму *A*) получим выражение для неустойчивости данного алгоритма на контрольной выборке $\tilde{S}^q=\{S_1,S_2,...,S_q\}$

$$Z_B(\varepsilon)=\sum_{j=1}^{q}Z_B(S_j,\varepsilon).$$

Будем искать коллективное решение в виде выпуклой комбинации исходных алгоритмов

$$\Gamma_B^k(S)=\frac{\sum_{j=1}^{q}w_j(S)\Gamma_{B(F(j))}^k(S)}{\sum_{j=1}^{q}w_j(S)},\ k=1\ldots l\ ,\ S\in R^n,$$

где $F:\{1...q\}\to\{1...p\}$ - некоторая функция, определяющая номер распознающего оператора для каждого объекта контрольной выборки; а $w_j:R^n\to\bar{R}^+$ - весовые функции, обладающие следующими свойствами:

$$w_j(S)\to 0 \qquad \text{при } \rho(S,S_j)\to\infty,$$

$$\frac{w_i(S)}{\sum_{j=1}^{q} w_j(S)} \to 1 \qquad \text{при } \rho(S,S_i) \to 0 \qquad .$$

В качестве функции F будем использовать номер наиболее устойчивого на соответствующем объекте алгоритма, который его правильно распознает (если таковой имеется), и номер наиболее устойчивого алгоритма среди всех, если ни один исходный алгоритм не распознает данный объект верно. Тогда подобрав соответствующим образом весовую функцию, получим коллективное решение, названное выпуклым стабилизатором.

Было показано, что выпуклая стабилизация позволяет сохранить высокие дискриминационные свойства исходных алгоритмов, увеличивая при этом их устойчивость по отношению к изменениям положения объектов (а, следовательно, гарантируя перенесение хорошего качества работы на произвольные объекты). Для того чтобы использовать выпуклую стабилизацию алгоритмов, необходимо провести процедуру обучения по контрольной (возможно, совпадающей с обучающей) выборке. Из-за большого объема вычислений, выполняемых как на этапе обучения, так и на этапе распознавания, данный метод имеет смысл применять на небольших выборках /84/.

Другие возможные подходы и алгоритмы построения коллективных решений задачи распознавания описаны в /47, 69/. Подходы, используемые в Системе РАСПОЗНАВАНИЕ, описаны в главе 3.

Глава2. Математические методы кластерного анализа (классификации без учителя).

Важный раздел теории распознавания составляют методы кластеризации (или автоматической классификации, таксономии, самообучения, обучения без учителя, группировки), решающие задачи разбиения выборок объектов (при заданных признаковых пространствах или матрицах близостей объектов) на классы эквивалентности, причем эквивалентность объектов базируется на мерах близости, сходства и т.п. Из перечисленного набора используемых названий, близких друг другу и воспринимаемых почти как синонимы, будет далее использоваться обычно термин «кластеризация» - узкоспециализированный, но не допускающий двусмысленности как, например, термин «классификация». Соответственно, термин «кластер» будет использоваться как синоним термину «класс», но обозначать именно множество близких объектов, полученное как результат решения задачи кластерного анализа.

Методы кластерного анализа позволяют решать задачи минимизации числа эталонов, поиска эталонных описаний, выявления структурных свойств классов и многие другие вопросы анализа данных. Принципы, согласно которым объекты объединяются в один кластер, являются обычно “внутренним делом” конкретного алгоритма кластеризации. Пользователь, зная данные принципы, может в определенных пределах интерпретировать результаты каждого конкретного метода. В отличие от задач распознавания, различные методы кластеризации могут приводить к решениям, имеющим весьма существенные различия. Таким образом, кроме набора разнообразных методов кластеризации, практический интерес представляет наличие средств автоматической обработки результатов, полученных независимо различными алгоритмами /51, 52/. В настоящей главе будут описаны некоторые подходы для решения задачи кластерного анализа, а также метод комитетного синтеза оптимальных коллективных решений. Конкретные методы кластеризации, реализованные в системе РАСПОЗНАВАНИЕ, описаны дополнительно в разделе 3.13.

Предварительно следует сделать два замечания. Во-первых, различают задачи кластерного анализа (и соответственно алгоритмы) с заданным (или известным) числом кластеров, а также с не заданным (неизвестным) числом кластеров. В последнем случае оптимальная кластеризация и число кластеров находятся в результате решения единой задачи. Во-вторых, кроме обычной постановки задачи кластеризации, как задачи поиска разбиений, существуют постановки как задачи поиска покрытий и структур на заданном множестве прецедентов.

Далее основная задача кластеризации будет рассматриваться прежде всего как задача поиска разбиений выборки признаковых описаний $I(S_1),I(S_2),...,I(S_m)$, $I(S) = (x_1(S),x_2(S),...,x_n(S))$, заданной числовой таблицей T_{nm}. В «нестрогой» постановке данная задача формулируется как поиск разбиения выборки на группировки (классы, кластеры, таксоны) близких объектов, причем само искомое разбиение находится как решение некоторой оптимизационной задачи, как результат сходимости некоторой итерационной процедуры, как результат применения некоторой детерминированной процедуры, и т.п.

2.1. Кластеризация, как задача поиска оптимального разбиения.

Пусть рассматривается задача кластеризации на l кластеров. Выборку признаковых описаний объектов будем обозначать как $X = \{\mathbf{x}_1, \mathbf{x}_2,..., \mathbf{x}_m\}$, $\mathbf{x}_i = (x_{i1}, x_{i2},..., x_{in})$. Разбиением $\mathrm{K} = \{K_1, K_2,..., K_l\}$ выборки $X = \{\mathbf{x}_1, \mathbf{x}_2,..., \mathbf{x}_m\}$ на l групп является произвольная совокупность непересекающихся подмножеств множества X, покрывающая все объекты выборки: $K_i \subseteq X, i = 1,2,...,l$, $\bigcup_{i=1}^{l} K_i = X$, $K_i \cap K_j = \varnothing, i \neq j$. Пусть задан некоторый критерий качества $F(\mathrm{K})$ разбиения K. Тогда задача кластеризации будет состоять в нахождении разбиения K^*, доставляющего экстремум критерию $F(\mathrm{K})$: $F(\mathrm{K}^*) = \underset{\mathrm{K}\in\{\mathrm{K}\}}{extr} F(\mathrm{K})$.

Приведем примеры подобных критериев /19/.

1. *Сумма внутриклассовых дисперсий, или сумма квадратов ошибок.*

$F(\mathrm{K}) = \sum_{j=1}^{l} \sum_{\mathbf{x}_i \in K_j} \rho^2(\mathbf{x}_i, \mathbf{y}_j)$, где $\mathbf{y}_j = \frac{1}{n_j} \sum_{\mathbf{x}_i \in K_j} \mathbf{x}_i$, $n_j = |K_j|$ – число объектов в группе K_j.

Решением задачи кластерного анализа при данном критерии считается такое разбиение K^*, которое доставляет минимум функционалу $F(\mathrm{K})$.

2. *Критерии на базе матриц рассеяния.*

Матрица рассеяния для группы K_j определяется как $\Sigma_j = \sum_{\mathbf{x}_i \in K_j} (\mathbf{x}_i - \mathbf{y}_j)(\mathbf{x}_i - \mathbf{y}_j)^t$, а матрица внутригруппового рассеяния как $\Sigma = \sum_{i=1}^{l} \Sigma_j$ (здесь символ t означает символ транспонирования).

Известно несколько определений критериев кластеризации на базе матриц внутригруппового рассеяния. Например, выбор в качестве критерия определителя матрицы внутригруппового рассеяния $F(\mathrm{K}) = |\Sigma|$. Здесь решение задачи кластерного

анализа при данном критерии также находится в результате дискретной минимизации.

Поскольку число всевозможных разбиений $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ на l групп оценивается как $\frac{l^m}{l!}$ (например, при весьма «скромных» параметрах выборки m=100 , l=5, $\frac{l^m}{l!} \approx 10^{67}$ /19/), полный перебор разбиений здесь заведомо исключен. Поэтому обычно применяют методы частичного перебора с использованием случайного выбора начальных разбиений и последующей локальной оптимизацией. В методах локальной оптимизации (для определенности, минимизации) строится последовательность разбиений $\mathrm{K}_1, \mathrm{K}_2, ..., \mathrm{K}_i, ...,$ для которой $F(\mathrm{K}_1) > F(\mathrm{K}_2) >, ..., > F(\mathrm{K}_i) >, ...$ а разбиение K_{i+1} вычисляется непосредственно по K_i путем его «локального» изменения. Пусть $\mathrm{K}_i = \{K_1^i, K_2^i, ..., K_l^i\}$, $\mathrm{K}_{i+1} = \{K_1^{i+1}, K_2^{i+1}, ..., K_l^{i+1}\}$, тогда $K_j^{i+1} = K_j^i, j \neq \nu, \mu,$ $K_\nu^{i+1} = K_\nu^i \setminus \{\mathbf{x}_t\}, K_\mu^{i+1} = K_\mu^i \cup \{\mathbf{x}_t\}$. Некоторый элемент $\mathbf{x}_t \in K_\nu^i$ переносится в какую-то другую группу, если это приводит к уменьшению значения функционала на новом разбиении. Конечность данной процедуры локальной (итеративной) оптимизации очевидна. Более подробное описание метода для критерия суммы внутриклассовых дисперсий приведено в разделе 3.16.

Широко известен метод **«*k*- внутригрупповых средних».** В данном методе строится последовательность разбиений $\mathrm{K}_i = \{K_1^i, K_2^i, ..., K_l^i\}$, $i = 1,2,...$ как результат выполнения следующих однотипных итераций. Пусть разбиение K_1 выбрано случайно. Для группы K_1^i находится ее центр $\mathbf{y}_1 = \frac{1}{n_1} \sum_{\mathbf{x}_j \in K_1^i} \mathbf{x}_j$. Далее в группу K_1^{i+1} зачисляются все элементы выборки, которые ближе к $\mathbf{y}_1$ чем к аналогично полученным $\mathbf{y}_2, \mathbf{y}_3, ..., \mathbf{y}_l$. Группа K_2^{i+1} строится аналогично, но относительно множества объектов $X \setminus K_1^{i+1}$, и т.д. После вычисления $K_1^{i+1}, K_2^{i+1} ..., K_l^{i+1}$ их центры пересчитываются и вычислительный процесс повторяется (см. раздел 3.13).

Метод Форель является также представителем подходов, в котором кластеры находятся не в результате оптимизации некоторого критерия, а с помощью итерационных процедур – движения гипершаров фиксированного радиуса в сторону мест «сгущения» объектов /30, 31/.

Пусть фиксировано некоторое положительное число *R*. Выбирается случайный элемент $\mathbf{x}_t \in X$ и гипершар радиуса R с центром в $\mathbf{y}_1 = \mathbf{x}_t$: $\mathrm{R}_1 = \{\mathbf{x} : \rho(\mathbf{x}, \mathbf{y}_1) \leq R\}$. Полагаем $K_1^1 = \{\mathbf{x}_i : \mathbf{x}_i \in X \cap \mathrm{R}_1\}$. Вычисляется центр $\mathbf{y}_2 = \frac{1}{\left|K_1^1\right|} \sum_{\mathbf{x}_i \in K_1^1} \mathbf{x}_i$ новой сферы

$\mathrm{R}_2 = \{\mathbf{x} : \rho(\mathbf{x}, \mathbf{y}_2) \le R\}$ и группа $K_1^2 = \{\mathbf{x}_i : \mathbf{x}_i \in X \cap \mathrm{R}_2\}$. Процесс заканчивается вычислением такой группы объектов $K_1^t = \{\mathbf{x}_i : \mathbf{x}_i \in X \cap \mathrm{R}_t\}$, для которой $K_1^t = K_1^{t+1}$. Полученное множество объектов K_1^t объявляется первым кластером K_1. Оно исключается из X и вышеописанная процедура повторяется относительно оставшейся части выборки. Таким образом, последовательно находятся кластеры $K_2, K_3, ..., K_l$. Процесс кластеризации заканчивается при достижении ситуации $X \setminus \{K_1, K_2, ..., K_l\} = \varnothing$. Полученное число кластеров зависит от выбора радиуса *R,* который является параметром алгоритма.

2.2. Кластеризация, как задача поиска оптимального покрытия.

Данная постановка может оказаться в ряде практических случаев предпочтительнее задачи поиска оптимальных разбиений. Примерами подобных ситуаций являются случаи, когда возможность пересечения кластеров обусловлена самой природой практической задачи (объекты разных кластеров эквивалентны согласно различным отношениям эквивалентности), наличием «выбросов» и «шумовых» объектов. Под последними можно понимать те объекты, для которых отнесение в один из кластеров не является более предпочтительным, чем отнесение в другой. В таких случаях более правильным может быть решение об их принадлежности двум (или нескольким) кластерам, чем «искусственное» зачисление в один из кластеров. Наконец, решение задачи поиска оптимальных покрытий может оказаться проще, особенно в случаях зашумленных и неполных данных.

Рассмотрим один алгоритм решения данной задачи для заданного числа *l* кластеров. Считаем также предварительно заданными (или определенными) l векторов $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_l \in R^n$, которые являются центрами «сгущения» выборки и интерпретируются как центры кластеров. Они могут быть получены, например, как центры гипершаров после предварительной кластеризации выборки с помощью алгоритма Форель или с помощью метода ***k*- внутригрупповых средних**.

Введем в рассмотрение специальные семейства вложенных гипершаров.

Пусть ρ - некоторая метрика в пространстве R^n. Упорядочив по возрастанию значения расстояний от y_i до всех элементов из $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$, получим монотонно возрастающие последовательности $r_1^i < r_2^i < ... < r_{k_i}^i$, $k_i \le m$, $i = 1,2,...,l$. Через $\{R_j^i = \{\mathbf{x} : \rho(\mathbf{x}, \mathbf{y}_i) \le r_j^i\}, j = 1,2,...,k_i\}$, обозначим множество вложенных гипершаров с

центрами в $\mathbf{y}_i$. Множество данных гипершаров покрывает все объекты выборки. Обозначим $n_{ij} = \left|\{\mathbf{x}_t : \mathbf{x}_t \in R_j^i\}\right|$ - число объектов выборки, принадлежащих гипершару R_j^i.

Теперь можно поставить следующую задачу: «Найти l гипершаров $R_{t_1}^1, R_{t_2}^2, ..., R_{t_l}^l$, покрывающих все объекты из $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ и имеющих минимальное число общих объектов $\sum_{i=1}^{l} n_{it_i} \to \min$ ».

Данная известная задача поиска покрытия минимального веса формулируется в виде следующей задачи целочисленного линейного программирования.

$$\sum_{i=1}^{l}\sum_{j=1}^{k_i} n_{ij} z_{ij} \to \min, \tag{2.1}$$

$$\sum_{i=1}^{l}\sum_{j=1}^{k_i} c_{ij}^t z_{ij} \geq 1,\ t = 1,2,...,m, \tag{2.2}$$

$$\sum_{j=1}^{k_i} z_{ij} \geq 1, i = 1,2,...,l, \tag{2.3}$$

$$z_{ij} \in \{0,1\}. \tag{2.4}$$

где $c_{ij}^t = \begin{cases} 1, & x_t \in R_j^i, \\ 0, & \text{иначе}. \end{cases}$

Ограничения (2.2) выражают требование покрытия каждого объекта хотя бы одним шаром, а (2.3) – присутствие в решении основной задачи хотя бы одного шара с центром в y_j для каждого $j = 1,2,...,l$.

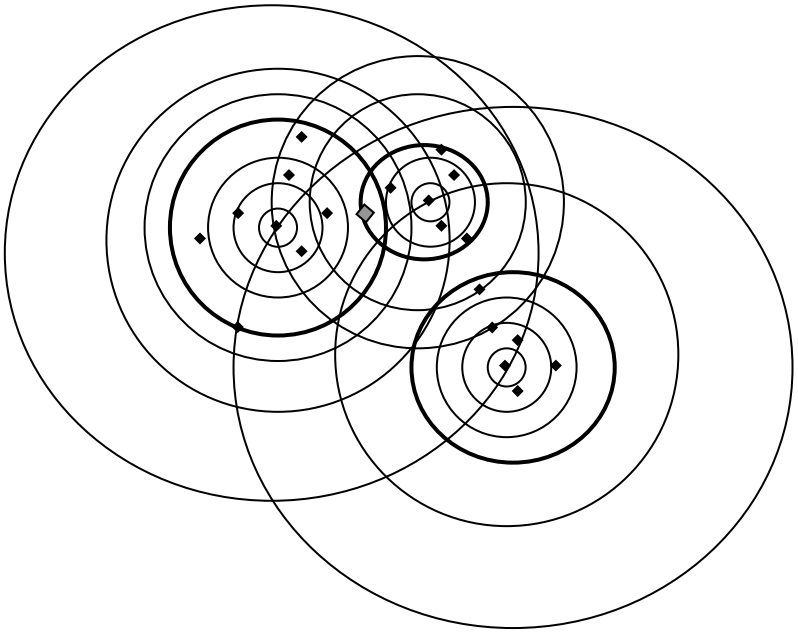

Рис. 17. Пример кластеризации в виде оптимального покрытия гипершарами

По построению, имеет место монотонность коэффициентов функции (2.1) и ограничений (2.2):

$n_{i,j} < n_{i,j+1},\ c_{i,j}^t \le c_{i,j+1}^t$.

Пусть $\{z_{ij}^*, i = 1,2,...,l,\ j = 1,2,...,k_i\}$ - решение задачи (2.1)- (2.4). В силу условий (2.3), положительности коэффициентов целевой функции (2.1) и свойств монотонности коэффициентов, будут выполнены равенства $\sum_{j=1}^{k_i} z_{ij}^* = 1, i = 1,2,...,l.$ Тогда совокупность шаров $R_{t_1}^1, R_{t_2}^2,..., R_{t_l}^l$, соответствующих всем единичным значениям $z_{1t_1}^*, z_{2t_2}^*,..., z_{2t_l}^*$, и образует искомое решение. Окончательно, в качестве решения задачи кластерного анализа принимается совокупность подмножеств $K_1, K_2,..., K_l$, $K_i = \{\mathbf{x}_j : \mathbf{x}_j \in R_{t_i}^i, j = 1,2,...,m\}$.

На Рис. 17 приведен пример задачи кластеризации на три кластера, в котором два кластера имеют один общий объект. Максимальный радиус шаров с центром в $\mathbf{y}_j$ ограничивался сверху величиной $\max_{t \ne j} \rho(\mathbf{y}_j, \mathbf{y}_t)$.

2.3. Кластеризация, как задача поиска структур в данных.

Классическим примером данной задачи является иерархическая группировка. Здесь задача кластеризации на l кластеров решается последовательным объединением ближайших групп (существуют и подходы, основанные на последовательном разбиении групп объектов). На первом шаге считается, что каждый объект образует «одноточечный» кластер: $K_i^1 = \{\mathbf{x}_i\}, i = 1,2,...,m$. На втором шаге два ближайших кластера объединяются в один. Процесс объединения повторяется до нахождения l кластеров. Приведем наиболее распространенный алгоритм иерархической кластеризации.

Пусть фиксирована метрика (или полуметрика $\rho(\boldsymbol{x},\boldsymbol{y})$) на множестве векторов признаковых описаний $\boldsymbol{x}=(x_1, x_2, ..., x_n)$, x_i - действительные числа. Введем также меру расстояния между группами объектов $d(T_i, T_j)$. Примерами подобных функций являются функции (2.5) - (2.8) .

$$d_{\min}(T_i, T_j) = \min_{\mathbf{x}_\nu \in T_i, \mathbf{x}_\mu \in T_j} \rho(\mathbf{x}_\nu, \mathbf{x}_\mu), \qquad (2.5)$$

$$d_{\max}(T_i, T_j) = \max_{\mathbf{x}_\nu \in T_i, \mathbf{x}_\mu \in T_j} \rho(\mathbf{x}_\nu, \mathbf{x}_\mu), \qquad (2.6)$$

$$d_{avr}(T_i, T_j) = \frac{1}{n_i n_j} \sum_{\mathbf{x}_\nu \in T_i} \sum_{\mathbf{x}_\mu \in T_i} \rho(\mathbf{x}_\nu, \mathbf{x}_\mu), \qquad (2.7)$$

$$d_{mean}(T_i, T_j) = \rho(y_i, y_j), \text{где } y_i = \frac{1}{|T_i|} \sum_{\mathbf{x}_\nu \in T_i} \mathbf{x}_\nu . \qquad (2.8)$$

Пусть заданы функция $\rho(\mathbf{x},\mathbf{y})$ и функции $d(T_i, T_j)$ для групп объектов T_i и T_j. Рассмотрим задачу кластеризации на заданное число кластеров. Пусть к шагу № k, $k=1,2,\ldots$, получены кластеры $T^k_{1,}$, $T^k_{2,}$,..., $T^k_{m-k+1,}$, (при $k=1$, $T_\nu^1=\{\mathbf{x}_\nu\}, \nu=1,2,\ldots,m$). От данных $(m-k+1)$ кластеров осуществляется переход к $(m-k)$ кластерам путем объединения в один той пары T^k_ν, T^k_μ, для которой $d(T_\nu^k,T_\mu^k)=\min\limits_{\sigma,\tau} d(T_\sigma^i,T_\tau^i)$. Для нахождения разбиения исходной выборки на l кластеров требуется выполнение $m-l$ шагов. Полученные кластеры являются разбиением исходной выборки, причем каждый из кластеров имеет структурное представление в терминах расстояний объектов и кластеров.

2.4. Решение задачи кластерного анализа коллективами алгоритмов

Пусть в результате применения разнообразных методов кластеризации получено множество различных решений для одних и тех же данных. При отсутствии внешнего критерия, выбор какого-то одного решения из данного множества кластеризаций, наиболее адекватного реальности, может быть не ясен. Поэтому представляет интерес применение методов обработки полученных множеств кластеризаций с целью построения коллективных решений, более предпочтительных и обоснованных, чем полученные отдельными алгоритмами кластеризации.

В настоящем разделе рассматривается двухуровневый подход к решению задачи кластерного анализа. Первый уровень ее решения состоит в независимом применении набора различных алгоритмов для кластеризации данных и нахождения набора кластеризаций. Второй уровень состоит в построении окончательного оптимального коллективного решения в результате решения специальной дискретной оптимизационной задачи на перестановках. Интерпретация полученных результатов следующая: находится разбиение выборки объектов на такие подмножества, многие элементы каждого их которых принадлежат “значительному” числу одних и тех же кластеров, полученных исходными алгоритмами. Таким образом, кластеры коллективного решения образуют объекты, «близкие» друг другу по нескольким подходам одновременно.

В работах [51,52] разработана теория комитетного синтеза оптимальных решений задачи кластерного анализа коллективами алгоритмов кластеризации. Опишем кратко основные идеи и алгоритм данного подхода.

Пусть некоторым алгоритмом решена задача кластеризации для выборки объектов $X=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_m\}$ в виде разбиения или покрытия данной выборки l множествами-кластерами. Обозначим эти кластеры через $T_1,\ldots,T_l$. Результат кластеризации можно

записать в виде информационной матрицы $I = \|\alpha_{ij}\|_{m\times l}$, где $\alpha_{ij} \in \{0,1\}$, $\alpha_{ij} = 1$, если $S_i \in T_j$, $\alpha_{ij} = 0$, если $S_i \notin T_j$. Наличие нескольких единиц в одной строке соответствует принадлежности объекта сразу нескольким кластерам. Нулевая строка означает отказ от кластеризации соответствующего объекта.

Определение 6: Информационные матрицы $I = \|\alpha_{ij}\|_{m\times l}$ и $I' = \|\alpha'_{ij}\|_{m\times l}$ называются эквивалентными, если они равны с точностью до перестановки столбцов.

Произвольная матрица $\|\alpha_{ij}\|_{m\times l}$ определяет класс $K\left(\|\alpha_{ij}\|_{m\times l}\right)$ всех эквивалентных ей матриц.

Определение 7: Алгоритмом кластеризации A^c называется алгоритм, переводящий описания объектов $\mathbf{x}_1,...,\mathbf{x}_m$ в класс эквивалентности $K\left(\|\alpha_{ij}\|_{m\times l}\right)$, то есть:

$$A^c(\mathbf{x}_1,...,\mathbf{x}_m) = K\left(\|\alpha_{ij}\|_{m\times l}\right).$$

Данное определение отражает факт свободы в обозначении полученных алгоритмом кластеров.

Пусть дано n кластеризаций выборки алгоритмами $A_1^c,...,A_n^c$: $A_\nu^c(\mathbf{x}_1,...,\mathbf{x}_m) = K\left(\|\alpha_{ij}^\nu\|_{m\times l}\right)$ на l кластеров. Задача построения оптимальной коллективной кластеризации состоит в вычислении по множеству из n решений $K\left(\|\alpha_{ij}^1\|_{m\times l}\right), K\left(\|\alpha_{ij}^2\|_{m\times l}\right),...,K\left(\|\alpha_{ij}^n\|_{m\times l}\right)$ некоторого нового $K\left(\|c_{ij}\|_{m\times l}\right)$ (где $c_{ij} \in \{0,1\}$), обоснованного с содержательной стороны и оптимального с формальной.

Пусть $I_\nu \in K\left(\|\alpha_{ij}^\nu\|_{m\times l}\right)$ (для простоты будем считать, что $I_\nu = \|\alpha_{ij}^\nu\|_{m\times l}$. Оператор $\mathbf{B}(I_1,...,I_n) = B = \|b_{ij}\|_{m\times l}$ называется сумматором, если $b_{ij} = \sum_{\nu=1}^{n} \alpha_{ij}^\nu$. Очевидно, что $0 \le b_{ij} \le n$.

Матрицу, полученную в результате применения сумматора к некоторому набору информационных матриц, будем называть матрицей оценок. Оператор r называется решающим правилом, если

$$r(B) = \|c_{ij}\|_{m\times l}, \text{ где } c_{ij} \in \{0,1\},\ c_{ij} = \begin{cases} 1,\ b_{ij} \ge b_{ik},\ k \ne j, k = 1,...,l \\ 0,\ \text{иначе.} \end{cases}.$$

Определение 8. Комитетным синтезом информационной матрицы C на элементе $\tilde{I} = (I_1,...,I_n)$ называют ее вычисление: $C = r(\mathbf{B}(\tilde{I}))$, при условии **B** - сумматор, r - решающее правило.

Таким образом, матрицы $I_\nu, \nu = 1,2,...,n$, поэлементно суммируются. Полученной числовой матрице ставится в соответствие бинарная матрица C. Тогда коллективным решением можно считать $K(C)$. Данное решение определяется конкретным выбором матриц $I_\nu, \nu = 1,2,...,n$. Для оценивания коллективного решения $K(C)$ вводится понятие контрастных матриц оценок $\|b_{ij}\|_{m\times l}$, под которыми понимаются произвольные числовые матрицы со свойством $b_{ij} \in \{0,n\}$. Контрастные матрицы соответствуют тому гипотетическому случаю, когда все исходные решения задач классификации оказались равными (вместе с обозначениями классов). Случаи контрастных матриц считаются наилучшими по множеству всевозможных наборов бинарных матриц $\tilde{I} = (I_1,...,I_n)$ и «потенциально наилучшими» по множеству допустимых наборов $\tilde{I} = (I_1,...,I_n)$, $I_\nu \in K\left(\|\alpha_{ij}^\nu\|_{m\times l}\right)$. Тогда качество некоторой матрицы оценок B определяется как ее расстояние до множества всех контрастных матриц той же размерности: $\Phi(B) = \min\limits_{B^*} \sum\limits_{i=1}^{m}\sum\limits_{j=1}^{l} |b_{ij} - b_{ij}^*|$, $\mathrm{B}^* = \{\|b_{ij}^*\|\}$ - множество всех контрастных матриц размерности $m\times l$. Задача построения оптимальной коллективной классификации сводится к минимизации $\Phi(B)$, т.е. нахождению такого конкретного набора матриц $I_\nu^/, \nu = 1,2,...,n$, $I_\nu^/ \in K\left(\|\alpha_{ij}^\nu\|_{m\times l}\right)$, по которому вычисляется матрица оценок, «максимально похожая» на одну из контрастных матриц. Для решения данной дискретной оптимизационной задачи на перестановках разработан эффективный метод /50/.

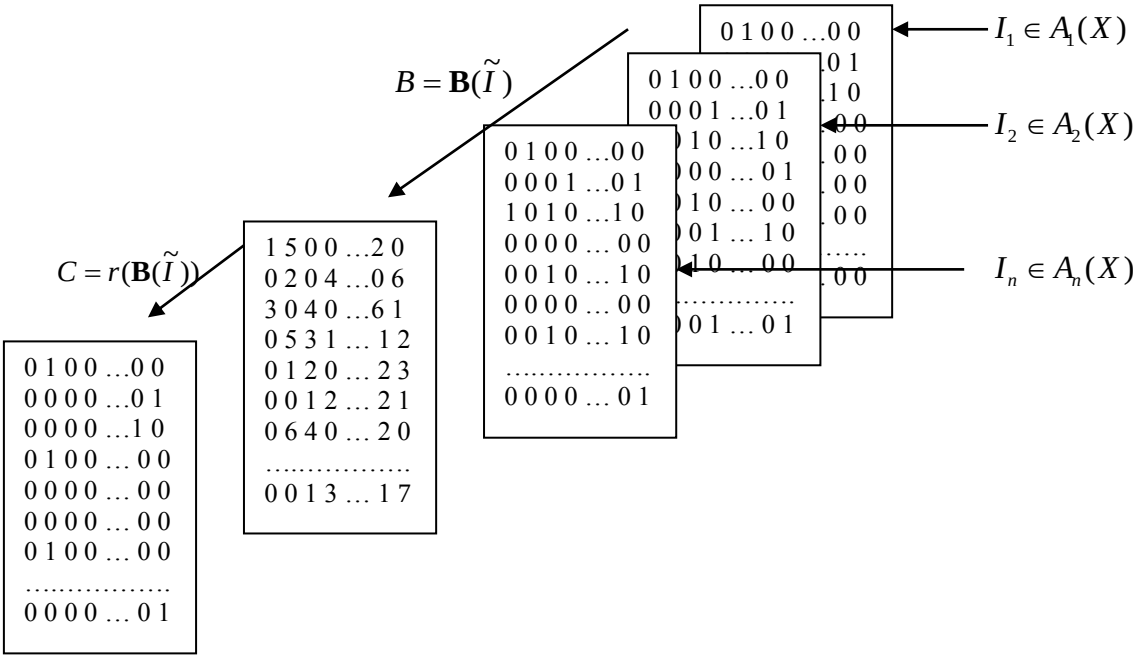

Рис.18. Общая схема комитетного синтеза коллективных кластеризаций

Вопрос выбора исходного набора алгоритмов кластеризации является в значительной мере «открытым» и здесь естественны различные подходы.

В качестве подобного «базиса» может использоваться произвольный набор имеющихся кластеризаций. Полученное коллективное решение (набор кластеров) уже интерпретируется в терминах тех исходных кластеров, пересечению которых соответствуют кластеры коллективного решения.

Другой подход к выбору базисного набора коллективных кластеризаций состоит в использовании набора различных кластеризаций, полученных в рамках одного подхода. Например, это могут быть кластеризации, соответствующие различным локально-оптимальным разбиениям по некоторому критерию качества разбиений (например, сумме квадратов дисперсий).

Представляет интерес использование и «человеко-машинных» подходов. Действительно, кластеризация с помощью автоматических формальных процедур очень сильно зависит от используемых метрик, критериев качества, зашумленности самих данных, множества параметров алгоритмов и других причин. В то же время, человеческая способность визуального выявления закономерностей на плоских конфигурациях превосходит формальные подходы. Способность человека «точного» решения «плоских» задач кластерного анализа и возможность синтеза оптимальных коллективных кластеризаций были положены в основу видео-логического метода для решения задачи кластерного анализа.

В данном подходе задача кластерного анализа решается в два этапа. Сначала пользователь просматривает проекции выборки на различные плоскости пар признаков и выделяет кластеры объектов на тех проекциях, где просматриваются некоторые сгущения или группировки. Далее, полученный набор плоских решений используется в качестве исходного множества кластеризаций для построения коллективного решения. Таким образом, решение задачи кластерного анализа находится в результате комитетного синтеза набора точных кластеризаций, полученных по различным частям обрабатываемой выборки /79/.

Для применения метода синтеза оптимальных коллективных решений в случаях различного числа кластеров полученных исходными алгоритмами, каждое исходное решение преобразуется к решению на l кластеров, например, с помощью объединения «ближайших» кластеров в один, или «дублирования» некоторых классов (создания равных столбцов в соответствующих матрицах I_ν).

Глава 3. Алгоритмы распознавания и интеллектуального анализа данных в системе РАСПРОЗНАВАНИЕ

В настоящем разделе приведены краткие описания реализованных в Системе РАСПОЗНАВАНИЕ математических методов для решения задач распознавания, классификации, прогноза и интеллектуального анализа данных, а также методы решения таких смежных задач, как визуализации данных и оценивания вероятности правильной классификации. Реализованные в Системе алгоритмы представляют все основные подходы, изложенные в главах 1, 2. Описания алгоритмов, уже представленных ранее, дополняются с существенным акцентом на их практическую реализацию и применение. При этом авторы старались избежать излишних повторений изложенного ранее материала.

3.1. Алгоритмы вычисления оценок

Оптимальные значения параметров алгоритмов распознавания, основанных на вычислении оценок, определяются из решения задачи оптимизации данной модели распознавания - находятся такие значения параметров, при которых точность распознавания на обучающей выборке является максимальной.

Для вычисления оценок используются формулы (1.16) или (1.17). Значения числовых параметров $(\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n)$ задают пороги близости соответствующих признаков и вычисляются как средний модуль разности значений признака по обучающей выборке:

$$\varepsilon_v = \frac{2}{m(m-1)} \sum_{i,j=1,i>j}^{m} | x_v(S_i) - x_v(S_j) | .$$

Для классификации применяется общее линейное решающее правило (1.14), неизвестные значения параметров которого находятся в результате решения задачи оптимизации модели. В данном случае решается задача поиска максимальной совместной подсистемы системы линейных неравенств с помощью релаксационного метода /46/(см. раздел 3.5).

3.2. Голосование по тупиковым тестам

В Системе РАСПОЗНАВАНИЕ реализован один стохастический вариант тестового алгоритма. Из таблицы обучения выбираются случайно N подтаблиц, каждая из которых состоит из 3 строк таблицы обучения, N подтаблиц, состоящих из 4 строк таблицы обучения, и т.д., N подтаблиц, состоящих из k строк таблицы обучения (здесь N и k – управляющие параметры программы). Каждая подтаблица не обязана содержать эталоны из каждого класса, т.е. допускаются подтаблицы с числом строк меньшим числа классов. Каждому тесту выбранной подтаблицы сопоставляется вес (качество), оцененный уже по

полной обучающей выборке. Для каждой подтаблицы находятся все тупиковые тесты либо один минимальный тест в зависимости от выбранного алгоритма поиска. В последнем случае для таблицы обучения находится не более $N\times(k-2)$ минимальных тестов случайных подтаблиц.

Обозначим множество всех найденных тупиковых тестов для подтаблиц, как и ранее, через $\{T\}$. Пусть $M_1 = \{S_i, S_j\}$ множество пар строк таблицы обучения, принадлежащих равным классам, а M_2 - множество пар строк из разных классов. Число элементов множеств M_1 и M_2 обозначим, соответственно, через n_1 и n_2. Антиблизость объектов по опорному множеству $T \in \{T\}$ определяется как $D_T(S_\nu, S_\mu) = 1 - B_T(S_\nu, S_\mu)$.

Определим «вес» опорного множества (в нашем случае теста T) согласно выражению (3.1)

$$Q_T = \frac{1}{n_2} \sum_{(S_i, S_j) \in M_2} D_T(S_i, S_j) - \frac{1}{n_1} \sum_{(S_i, S_j) \in M_1} D_T(S_i, S_j), \qquad (3.1)$$

а через $w_T = \dfrac{Q_T}{\sum_{T \in \{T\}} Q_T}$ – его удельный вес. Данные величины показывают, как часто бывают близки эталонные объекты одного класса и далеки объекты разных классов по выбранному опорному множеству.

Окончательно, оценки распознаваемого объекта за классы K_j, $j=1,2,\ldots,l$, вычисляются согласно следующей формуле:

$$\Gamma_j(S) = \frac{1}{2} \sum_{T \in \{T\}} w_T \left(|K_j|^{-1} \sum_{S_i \in K_j} B_T(S, S_i) + \left(m - |K_j|\right)^{-1} \sum_{S_i \notin K_j} D_T(S, S_i) \right).$$

Классификация осуществляется с помощью простейшего решающего правила.

В случаях практических задач с плохой отделимостью классов тупиковые тесты будут иметь большое число столбцов или могут вообще отсутствовать. Для «управления отделимостью классов» введен управляющий параметр программы (делитель ε - порогов), позволяющий увеличивать-уменьшать близость объектов. Для таблиц обучения с небольшим числом признаков возможно вычисление всех тупиковых тестов и, соответственно, голосование по всем тупиковым тестам. Для реализации данного варианта в Системе предусмотрена кнопка «переборный алгоритм».

3.3. Алгоритмы голосования по логическим закономерностям классов

Основой данного метода является поиск логических закономерностей в данных. Под логическими закономерностями класса K_j в данном случае понимаются предикаты вида

$$P(S) = (a_1 \le x_1(S) \le b_1) \,\&\, (a_2 \le x_2(S) \le b_2) \,\&\, \ldots \,\&\, (a_n \le x_n(S) \le b_n) \in \{0,1\} \qquad (3.2)$$

(или конъюнкции (3.2), соответствующие некоторому подмножеству признаков) такие, что:

1) хотя бы для одного объекта обучающей выборки $S_i \in K_j$ выполнено $P(S_i) = 1$;

2) для любого объекта S_i обучающей выборки $S_i \notin K_j$ выполнено $P(S_i) = 0$;

3) $P(S)$ доставляет экстремум некоторому критерию качества $\varphi(P) = \underset{P' \in \mathrm{P}}{extr}\, \varphi(P')$, где P - множество всевозможных предикатов (3.2), удовлетворяющих условиям 1), 2) /71, 76/.

В системе РАСПОЗНАВАНИЕ рассматривается стандартный критерий качества: $\varphi(P) =$ «число эталонов S_i из класса K_j : $P(S_i) = 1$»/$\left|K_j\right|$.

Логическая закономерность класса K_j называется частичной, если выполнены пункты 1), 3), а требование 2) заменяется более слабым 2′:

$\dfrac{\left|\{S_i \notin K_j \mid P(S_i) = 1\}\right|}{\left|\{P(S_i) = 1\}\right|} \le \varsigma$ (доля объектов «чужих» классов, для которых выполнено $P(S_i) = 1$, не превышает заданный порог).

Поскольку задача оптимизации $\varphi(P)$ обычно многоэкстремальна, логическими закономерностями класса считаются все предикаты $P(S)$, доставляющие локальный экстремум критерию $\varphi(P)$.

В случае веществennозначных признаков, логической закономерности (3.2) соответствует простая геометрическая интерпретация: в некотором признаковом подпространстве имеется гиперпараллелепипед, содержащий максимальное число объектов обучения из класса K_j и только класса K_j. Логические закономерности являются аналогом представительных наборов для случаев бинарных и *k*-значных признаков /3, 10, 22/. Другие близкие понятия рассматривались в /35/ и в многочисленных публикациям по решающим деревьям (например, /16, 17/).

Алгоритм поиска множества логических закономерностей класса состоит в решении последовательности однотипных «отмеченных» задач. Число данных задач

определяется автоматически согласно предполагаемому существованию $P(S)$, для которого стандартный критерий качества $\varphi(P) \geq h$ (h – параметр программы, именуемый как «минимальная доля объектов»). Опишем подобную «отмеченную» задачу.

Пусть $S_i \in K_j$ - случайно выбранный объект таблицы обучения (будем называть его «опорный» эталон). В работе /77/ описан метод поиска множества логических закономерностей $P(S)$ класса K_j таких, что $P(S_i) = 1$. Поиск оптимального предиката $P(S)$ для опорного эталона S_i (т.е. значений параметров $a_1, a_2, ..., a_n, b_1, b_2, ..., b_n$) осуществляется сначала на некоторой неравномерной сетке пространства R^n, которая задается числом интервалов разбиения значений каждого признака (для некоторых признаков, например *k*-значных, в реальности число интервалов будет меньше заданного). После нахождения оптимального предиката $P(S)$ на заданной сетке, происходит поиск оптимального предиката $P(S)$ на более мелкой сетке, в окрестности ранее найденного $P(S)$, и т.д. Процесс оптимизации заканчивается и задача поиска множества логических закономерностей, связанных с заданным опорным объектом считается решенной, если при переходе к более мелкой сетке не удается найти предикат $P(S)$ с более высоким значением критерия качества $\varphi(P)$.

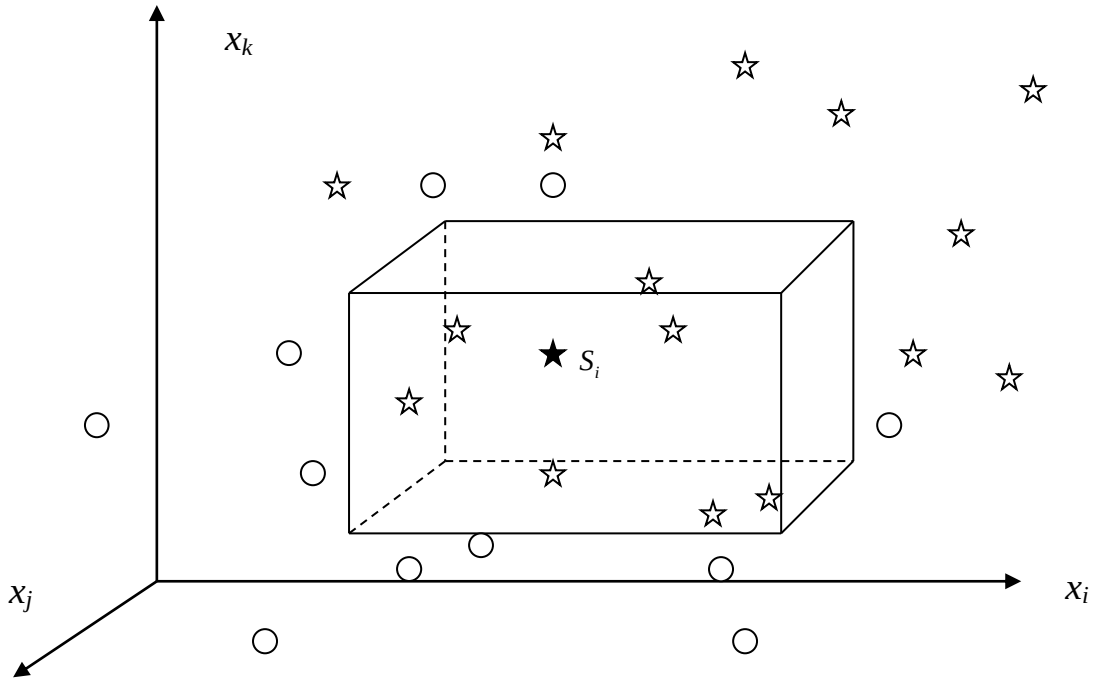

Рис.19. Геометрическая интерпретация логической закономерности класса. Символами «звездочка» отмечены объекты класса, для которого вычисляются логические закономерностями, символами «круг» - эталонные объекты остальных классов

Задача поиска оптимального $P(S)$ на каждой сетке состоит в поиске максимальной совместной подсистемы некоторой системы неравенств, при линейных ограничениях

относительно бинарных переменных, и некоторого ее решения. Последняя задача сводится к решению аналогичной задачи относительно вещественных переменных, которая решается с помощью релаксационного метода /24, 46/. В конечном итоге задача поиска оптимального предиката $P(S)$ для опорного эталона S_i заканчивается вычислением множества локально оптимальных предикатов $P(S)$ со свойством $P(S_i)=1$, причем конъюнкции (3.2) являются несократимыми (из них нельзя удалить какой-либо сомножитель).

Далее все вычисления повторяются для k случайно выбранных «опорных» эталонов класса K_j, а все найденные логические закономерности объединяются в одно множество $\mathbf{P_j}$. Значение параметра k определяется из соотношения

$$(1-h)^k \le (1-g), \qquad (3.3)$$

где g – параметр «уровень значимости» (0<g<1).

Результат работы алгоритма поиска логических закономерностей класса K_j формулируется следующим образом: «В вычисленном множестве логических закономерностей $\mathbf{P_j}=\{P(S)\}$ класса K_j с вероятностью не менее g имеется логическая закономерность $P(S)$, для которой $\varphi(P)\ge h$». Естественно, данный результат может быть получен при условии самого существования $P(S)$, для которого $\varphi(P)\ge h$.

Значение параметра k является важным фактором быстродействия программы. Например, при g=0.9 и h=0.1 из (3.3) следует $k\ge 22$. Значение $k=22$ вполне приемлемо для задач большой размерности.

ПРИМЕЧАНИЕ. Предположение $\varphi(P)\ge h$ служит лишь для автоматической оценки и выбора параметра k. При работе программы здесь возможны три результата: вычисленное множество $\mathbf{P_j}=\{P(S)\}$ содержит $P(S)$, для которого $\varphi(P)\ge h$, вычисленное множество $\mathbf{P_j}=\{P(S)\}$ не содержит $P(S)$, для которого $\varphi(P)\ge h$, множество $\mathbf{P_j}=\{P(S)\}$ оказывается пусто. Последний вариант соответствует ситуации, когда логических закономерностей нет ни для одного из выбранных опорных объектов. В данном случае следует выбрать большее число интервалов (более мелкую сетку) и повторить запуск программы, или искать частичные закономерности.

Найденные логические закономерности класса K_j могут быть «статистически значимыми» или нет. Для этого используется «перестановочный тест». Выполняется

серия из следующих t однотипных расчетов (t – параметр «количество случайных перестановок»). Осуществляется случайная перестановка строк таблицы обучения, при этом, как и ранее, первые m_1 строк новой таблицы $\tilde{T}_{nml}$ считаются эталонами первого класса, следующие по порядку $(m_2 - m_1)$ строк - эталонами второго класса, и т.д. (т.е. проводится случайное изменение номеров классов эталонных объектов с сохранением общего числа эталонов класса). Для таблиц $\tilde{T}_{nml}$ находятся наилучшие закономерности $P_i, i = 1,2,...,t,$ с соответствующими оценками качества $\varphi(P_i)$. Тогда логическая закономерность P_q из множества $\mathbf{P_j} = \{ P(S) \}$ считается статистически значимой, если из неравенств $\varphi(P_q) \geq \varphi(P_i)$, $i=1,2,...,t,$ выполнено не менее чем $100*g\%$.

При обработке неполных данных возможно использование двух методов обработки прочерков (неизвестных значений признаков): «жадный подход» и «осторожный подход». При жадном подходе считается, что все пары значений признаков сравниваемых объектов, одно или оба из которых неизвестны, близки (если объекты одного класса) и далеки (если объекты из разных классов), в то время как при осторожном, соответственно, – далеки и близки. Такой «прямой» способ позволяет автоматически обрабатывать пропущенные значения и получать хорошие решения, не прибегая к дополнительной обработке и преобразованиям прочерков.

Найденные множества логических закономерностей используются для вычисления информационных весов признаков и синтеза логических описаний классов.

Информационным весом признака считается величина $p_i = N_i / N$, где N_i - общее число логических закономерностей, в которые входит признак x_i, N – общее число логических закономерностей.

Оценки распознаваемых объектов за классы вычисляются следующим образом.

Пусть $P_i(S) = \underset{v \in \Omega}{\&} (a_v \leq x_v(S) \leq b_v), \Omega \subseteq \{1,2,...,n\}$ - логическая закономерность класса K_j. Считается, что логическая закономерность выполняется на объекте S' и объект получает «голос» за класс K_j, если предикат $P_i(S) = \underset{v \in \Omega}{\&} (\min(a_v, x_v(S')) \leq x_v(S) \leq \max(a_v, x_v(S')))$ удовлетворяет условию 2) (условию 2′ при работе с частичными закономерностями). Оценка S' за класс K_j вычисляется как сумма числа голосов по всем закономерностям данного класса, деленная на общее число закономерностей класса. Классификация объекта проводится с помощью простейшего решающего правила.

Множества логических закономерностей задают логические описания классов K_j, в качестве которых рассматриваются функции $D_j(S) = \vee P_i(S)$, где дизъюнкции берутся по множествам $\mathbf{P_j}$={ $P(S)$ }. Данные функции принимают значение 1 только на эталонах «своего» класса и 0 на всех эталонах «чужих» классов. Они могут рассматриваться как приближения характеристических функций классов K_j.

Кратчайшим логическим описанием класса K_j назовем логическую сумму $D_j^s(S) = \vee P_t(S)$, суммирование в которой проводится по подмножеству множества $\mathrm{P_j}$, содержащему минимальное число конъюнкций $P_t(S)$, и совпадающей с функцией $D_j(S)$ на эталонных объектах.

Минимальным логическим описанием класса K_j назовем логическую сумму $D_j^m(S) = \vee P_t(S)$, суммирование в которой проводится по подмножеству множества $\mathrm{P_j}$, содержащую минимальное общее число символов $x_1(S),x_2(S),...,x_n(S)$ в своей записи, и совпадающей с функцией $D_j(S)$ на эталонных объектах.

Логические (кратчайшие, минимальные) описания классов являются аналогами представлений частичных булевых функций в виде сокращенных дизъюнктивных нормальных форм (кратчайших, минимальных), а геометрические образы логических закономерностей классов - аналогами максимальных интервалов /25, 59/.

Если найдено множество $\mathrm{P_j}$ логических закономерностей класса K_j, то кратчайшее и минимальное логические описания класса находятся как решения задач поиска покрытий множества эталонов класса соответствующими предикатам $P_t(S)$ гиперпараллелепипедами.

Пусть $\mathrm{P_j}$={ $P^1(S), P^2(S),..., P^N(S)$ }, причем для каждого эталона класса K_j существует хотя бы один предикат из $\mathrm{P_j}$, выполняющийся на данном эталоне.

Рассмотрим задачу:

$$\sum_{t=1}^{N} a_t y_t \to \cdot \min, \qquad (3.4)$$

$$\sum_{t=1}^{N} P_t(S_i) y_t \geq 1, \cdots i : \cdot S_i \in K_j, \cdot y_t \in \{0,1\}. \qquad (3.5)$$

Тогда, при $a_t = 1$ единичные компоненты решения задачи (3.4-3.5) определяют предикаты кратчайшего логического описания $D_j^s(S)$ класса K_j, а при a_t, равных числу переменных в $P_t(S)$, - предикаты минимального логического описания.

Исходные множества $\mathrm{P_j}$ могут содержать равные или близкие элементы, мощность $\mathrm{P_j}$ может быть весьма велика (что однако является благоприятным в процедурах распознавания). Данные свойства множеств $\mathrm{P_j}$ существенно зависят от длины обучающей выборки и самого алгоритма их поиска. В то же время, кратчайшие и минимальные логические описания классов образуют уже неизбыточные подмножества $\mathrm{P_j}$, выражающие как основные свойства данных множеств, так и свойства самих классов.

Поэтому вычисление $D_j^s(S)$ и $D_j^m(S)$ может рассматриваться как один из подходов к проблеме сортировки логических закономерностей классов, а входящие в $D_j^s(S)$ и $D_j^m(S)$ предикаты - как наиболее компактные представления о классах, включающие как наиболее представительные знания (предикаты, покрывающие большое число эталонов), так и уникальные или редкие (предикаты, покрывающие малое число эталонов или отдельные из них).

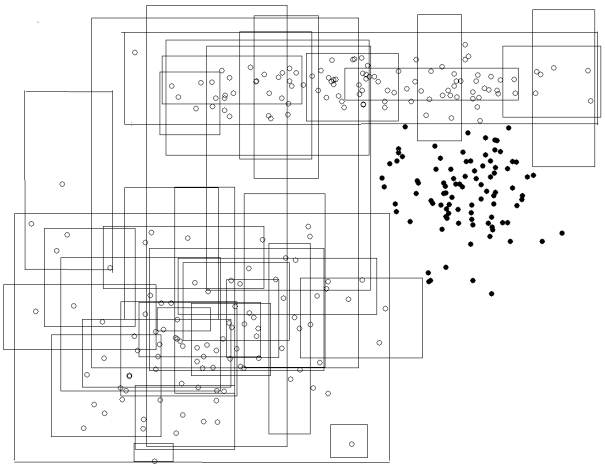

Рис. 20. Визуализация множества P_j логических закономерностей класса K_j («белый кружок»)

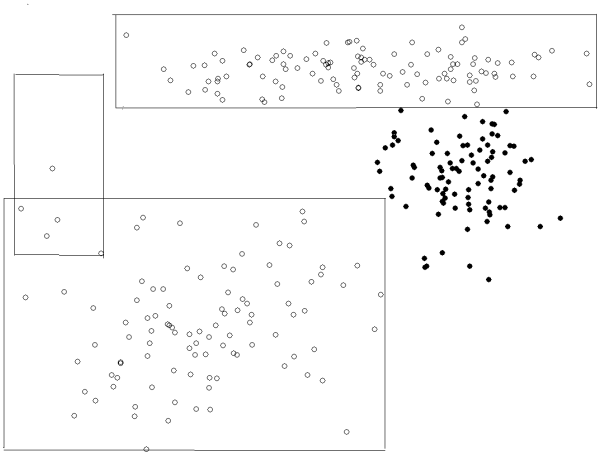

Рис. 21. Визуализация множества логических закономерностей кратчайшего описания класса K_j

3.4. Метод статистически взвешенных синдромов

Метод статистически взвешенных синдромов (СВС) основан на использовании процедуры взвешенного голосования по системам так называемых “синдромов” /37, 70, 78/. Под синдромом мы в данном случае понимаем подобласть пространства прогностических признаков принадлежащую разбиению этого пространства, обладающему разделяющей способностью. Мы будем говорить, что разбиение признакового пространства обладает разделяющей способностью, если долевые содержания объектов одного из классов в различных его областях значительно отличаются друг от друга. В методе СВС используются только одномерные и двумерные синдромы. Под одномерным синдромом, задаваемым признаком X_i, мы понимаем подобласть признакового пространства, для точек которой признак X_i удовлетворяет неравенствам $b_i' \geq X_i > b_i''$, где b_i', b_i'' вещественные граничные точки. Соответственно под двумерным синдромом, задаваемым признаками $X_{i'}$ и $X_{i''}$, понимается подобласть векторов признакового пространства, для которой признак $X_{i'}$ удовлетворяет неравенствам $b_{i'}' \geq X_{i'} > b_{i'}''$, а признак $X_{i''}$ удовлетворяет неравенствам $b_{i''}' \geq X_{i''} > b_{i''}''$.

Одномерные синдромы в методе СВС ищутся для каждого из классов по исходной обучающей выборке с помощью стабильных одномерных разбиений областей допустимых значений каждого из признаков. Соответствующие двумерные синдромы задаются как всевозможные пересечения одномерных синдромов.

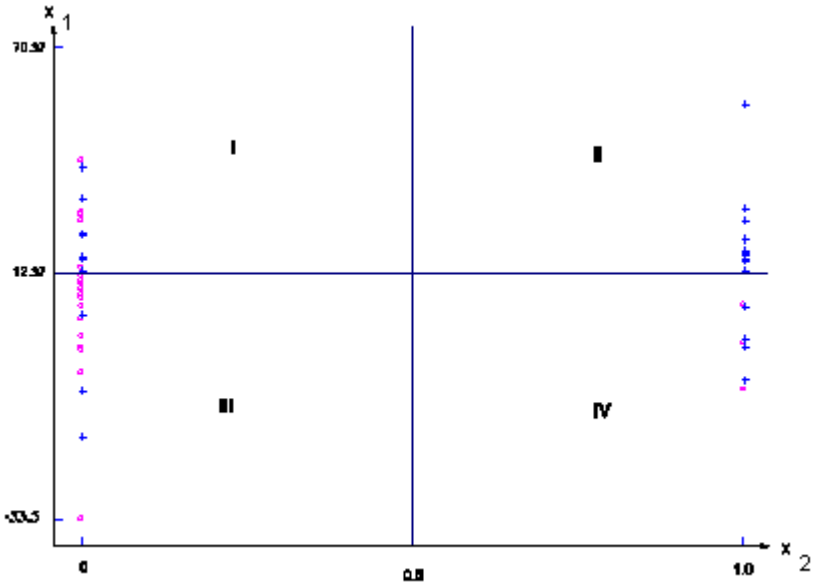

Рис.22. Пример двумерных синдромов. Символами «крестик» отмечены объекты класса, для которого вычисляются синдромы, символами «нолик» - объекты остальных классов. Видно, что доля крестиков в синдроме II значительно превышает долю крестиков в синдромах I , II и III. В то время как доля «крестиков» в синдроме III значительно ниже доли «крестиков» в синдромах I , II и IV. Синдромы I , II III и IV являются множеством всевозможных пересечений пары одномерных синдромов (I $\cup$ II) и (III $\cup$ IV), которые строятся по признаку x_1 и пары одномерных синдромов (I $\cup$ III) и (II $\cup$ IV), которые строятся по признаку x_2

Для построения одномерных синдромов используются модели I и II, различного уровня сложности. При этом модель I включает в себя все разбиения области допустимых значений признака с помощью одной граничной точки на две подобласти, а модель II включает в себя все разбиения области допустимых значений признака с помощью двух граничных точек на три подобласти. Пример двумерных синдромов дан на рисунке 22.

Оптимальное разбиение внутри фиксированной модели ищется путем максимизации функционала прогностической силы F_p.

Предположим, что мы ищем множество одномерных синдромов для класса K_j. Пусть R разбиение области допустимых значений некоторого признака X на L подобластей $q_1^j,\ldots,q_L^j$, ν_0^j - доля объектов класса K_j во всей обучающей выборке $\tilde{S}_0$, m_l^j - число объектов $\tilde{S}_0$, у которых значение X принадлежит подобласти q_l^j, ν_l^j - доля класса K_j в подмножестве объектов $\tilde{S}_0$, у которых значение X принадлежит подобласти q_l^j.

Тогда значение функционала $F_p(R,\tilde{S}_0,K_j)$ для разбиения R вычисляется по обучающей выборке $\tilde{S}_0$ в виде суммы: $F_p(R,\tilde{S}_0,K_j)=\frac{1}{\nu_0^j(1-\nu_0^j)}\sum_{l=1}^{L}(\nu_l^j-\nu_0^j)^2 m_l^j$.

В качестве оптимального для класса K_j разбиения R_o^j области допустимых значений признака X выбирается разбиение, доставляющее максимум функционалу $F_p(R,\tilde{S}_0,K_j)$. Для оценки качества распознавания наряду с функционалом прогностической силы используется также индекс нестабильности, задаваемый как отношение рассчитанной в режиме скользящего контроля средней вариации границ разбиений к выборочной дисперсии признака X. Пусть $b_1^0,\ldots,b_{L-1}^0$- граничные точки, разделяющие соседние подобласти разбиения R_o^j. Пусть $b_1^k,\ldots,b_{L-1}^k$ граничные точки разделяющие соседние подобласти разбиения R_k^j, рассчитанные по обучающей выборке $\tilde{S}_0$ без k-го объекта, D - выборочная дисперсия признака X. Индекс нестабильности границ $F_s(\tilde{S}_0,K_j,L)$ для модели с L подобластями рассчитывается как отношение:

$$F_s(\tilde{S}_0,K_j,L)=\frac{1}{D(L-1)}[\sum_{k=1}^{n}\sum_{l=1}^{L-1}(b_l^k-b_l^0)^2].$$

Выбор из моделей разбиений I или II в методе СВС регулируется с помощью порога δ_p для функционала прогностической силы F_p и порога δ_s для индекса нестабильности F_s. Каждый раз выбирается модель с большим значением функционала F_p, если ее индекс нестабильности не превышает порог δ_s. Если ни для одной модели не выполняются условия $F_p>\delta_p$ и $F_s<\delta_s$ то признак X исключается из рассмотрения.

Пусть $\tilde{Q}_j^0$ - множество синдромов для класса K_j, построенных на этапе обучения. Предположим, что для некоторого объекта s^* вектор описывающих его переменных $\mathbf{x}^*$ принадлежит пересечению синдромов $q_1^j,\ldots,q_r^j$ из системы $\tilde{Q}_j^0$. Тогда оценка объекта s^* за класс K_j вычисляется с помощью процедуры статистически взвешенного голосования.

$\Gamma_j(s^*)=\frac{\sum_{i=1}^{r} wei_i^j\nu_i^j}{\sum_{i=1}^{r} wei_i^j}$, где wei_i^j - вес i-ого синдрома, вычисляемый по формуле

$wei_i^j = \frac{m_i^j}{m_i^j + 1}\frac{1}{\hat{d}_i^j}$, $\hat{d}_i^j = (1-\nu_i^j)\nu_i^j + \frac{1}{n_i}(1-\nu_0^j)\nu_0^j$ - оценка дисперсии индикаторной функции класса K_j на синдроме q_i^j (слагаемое $\frac{1}{m_i^j}(1-\nu_0^j)\nu_0^j$ вводится для избежания обнуления $\hat{d}_i^j$ в случае, если в синдроме q_i^j содержатся только объекты из $K_j \cap \tilde{S}_0$ или только объекты из $CK_j \cap \tilde{S}_0$).

Метод СВС наряду с селекцией признаков по критерию прогностической силы и индексу нестабильности включает также дополнительный пошаговый отбор, максимизирующий рассчитываемый в режиме скользящего контроля коэффициент эффективности распознавания κ_j для класса K_j. Пусть у нас имеется некоторая контрольная выборка $\tilde{S} = \{s_1, \ldots, s_m\}$, для объектов которой вычислены оценки $\Gamma_j(s_1), \ldots, \Gamma_j(s_m)$ за класс K_j. Коэффициент κ_j вычисляется как коэффициент корреляции между оценками и индикаторной функцией класса:

$$\kappa_j = \frac{\sum_{i=1}^{m}[\Gamma(s_i) - \bar{\Gamma}(\tilde{S})][\alpha(s_i) - \bar{\alpha}(\tilde{S})]}{\sqrt{\sum_{i=1}^{m}[\Gamma(s_i) - \bar{\Gamma}(\tilde{S})]^2}\sqrt{\sum_{i=1}^{m}[\alpha(s_i) - \bar{\alpha}(\tilde{S})]^2}},$$ где $\alpha(s_i) = 1$ при $s_i \in K_j$ $\alpha(s_i) = 0$ при $s_i \in CK_j$, $\bar{\alpha}(\tilde{S})$ и $\bar{\Gamma}(\tilde{S})$ являются средними значениями индикаторной функции и функции оценок на выборке $\tilde{S}$. Отображая взаимосвязь между оценками и индикаторной функцией, коэффициент эффективного распознавания κ_j является хорошей мерой эффективности распознающего оператора, основанного на взвешенном голосовании согласно формуле (1.20). Нашей целью является поиска набора признаков, доставляющий максимум коэффициенту κ_j, рассчитанному в режиме скользящего контроля по обучающей выборке $\tilde{S}_0$. Для поиска такого набора в методе СВС используется пошаговая процедура. Пусть $\tilde{X}_{pr}^j$ - предварительный набор признаков для распознавания класса K_j, отобранных из исходного набора $\tilde{X}_{in}$ с помощью пороговых критериев прогностической силы и нестабильности границ. На первом шаге из набора $\tilde{X}_{pr}^j$ в информативный набор выбирается признак, для которого коэффициент эффективного распознавания κ_j имеет максимальное значение. На каждом последующем шаге к информативному набору добавляется признак, максимально увеличивающий

коэффициент κ_j. Процедура завершается, когда ни один из признаков, остающихся в наборе $\tilde{X}^j_{pr}$ не дает увеличения κ_j.

3.5. Линейная машина

Алгоритм «линейная машина» является известным общим подходом к разделению классов гиперплоскостями при числе классов большем двух. С каждым классом связывается линейная функция $f_j(\mathbf{x}) = \mathbf{w}_j\mathbf{x} + w_{0j}, j = 1,2,...,l$. Решающее правило имеет вид:

$$\mathbf{x} \in K_j, \text{ если } f_j(\mathbf{x}) > f_k(\mathbf{x}), k = 1,2,...,l, k \neq j$$

Нахождение неизвестных *(n+1)l* значений параметров $w_j, w_{0j}, j = 1,2,...,l$, по обучающим $T_{nml} = \|a_{ij}\|_{m\times n}$ и контрольным T'_{nql} данным осуществляется с помощью максимизации стандартного функционала качества распознавания – доли правильно распознанных объектов контрольной выборки.

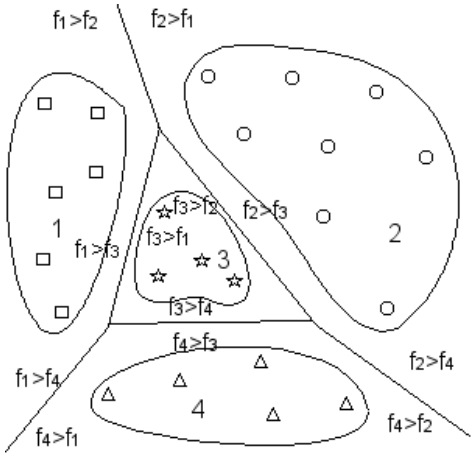

Рис. 23. Линейная машина для задачи из четырех классов. Границы областей решений, полученные с помощью линейной машины

Условием правильного распознавания объекта $S'_i \in K_j$ будет выполнение системы Z_i из *l-1* линейных неравенств

$Z_i: f_j(S_i') > f_k(S_i'), k = 1,2,...,l, k \neq j$,

а условием правильного распознавания всех контрольных объектов - выполнение системы $Z = \bigcup_{i=1}^{q} Z_i$ из *m(l-1)* неравенств (3.6).

$$A\mathbf{y} \geq 0, \mathbf{y} = \{\mathbf{w}_1 w_{01} \mathbf{w}_2 w_{02} ... \mathbf{w}_l w_{0l}\}, \qquad (3.6)$$

где A - матрица коэффициентов объединенной системы Z, а $\mathbf{y} = \{\mathbf{w}_1 w_{01} \mathbf{w}_2 w_{02} ... \mathbf{w}_l w_{0l}\}$ - вектор-столбец неизвестных параметров линейной машины.

В системе РАСПОЗНАВАНИЕ вместо системы (3.6) осуществляется решение системы

$$A\mathbf{y} \geq \varepsilon,\ \mathbf{y} = \{\mathbf{w}_1 w_{01} \mathbf{w}_2 w_{02} ... \mathbf{w}_l w_{0l}\} \qquad (3.7)$$

где управляющий параметр ε>0 - «компоненты правой части» задает ширину зоны, разделяющей классы и используется лишь с целью построения более устойчивой линейной машины: при построении линейной машины находятся такие ее параметры, которые разделяют контрольные объекты с некоторым «запасом».

Поскольку системы неравенств (3.7) как правило несовместны, в качестве «обобщенного» решения несовместной системы (3.7) принимается произвольное решение некоторой ее совместной подсистемы, состоящей из максимального числа систем Z_i. Здесь используется следующий релаксационный алгоритм приближенного поиска максимальной совместной подсистемы системы линейных неравенств /24, 46/.

Пусть дана система линейных неравенств

$$\sum_j a_{ij} x_j \geq b_i,\ i = 1...m,\ j = 1...n, \qquad (3.8)$$

Система (3.8) заменяется ей эквивалентной путем деления i- го неравенства на величину $\left|\mathbf{a}^i\right| = \sqrt{\sum_{j=1}^{n} a_{ij}^{\ 2}}$.

Считается, что смежные неравенства последовательно объединены в группы по n_1 штук. Требуется найти решение, удовлетворяющее максимальному количеству групп неравенств.

Построим следующую релаксационную последовательность $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(i)}, ...$ для поиска решения (3.8):

$$\mathbf{x}^{(i+1)} = \mathbf{x}^{(i)} + k^i \mathbf{w}^{(i)},$$

$$\text{где } \mathbf{w}^{(i)} = \sum_{j \in J^{(i)}} \mathbf{a}^j, \qquad k^i = \frac{\sum_{t \in J^{(i)}} (b_t - \sum_j a_{tj} x_j^{(i)})}{\sum_j w_j^{(i)2}}, \qquad (3.9)$$

$$J^{(i)} = \{t : \sum_j a_{tj} x_j \leq b_t\},\ t = 1...m.$$

Шаг k смещения вычисляется по формуле (3.9), либо выбирается равным на протяжении всех итераций.

Процесс поиска решения строится следующим образом. Задается произвольная начальная точка $\mathbf{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, ..., x_n^{(0)})$. В начале каждой итерации подсчитывается число выполненных групп неравенств. Если оно максимально относительно всех предыдущих итераций, то текущая точка $\mathbf{x}^{(i)}$ запоминается как самое лучшее на данный момент решение. После выполнения одного из критериев остановки счета, лучшее на данный момент решение становится окончательным. В качестве критериев остановки счета используются следующие:

1) выполнено заданное управляющим параметром максимально разрешенное число итераций;

2) сумма смещений за заданное число шагов не превышает некоторого минимального значения;

3) множество невыполненных неравенств J пусто.

Для приближенного поиска максимальной совместной подсистемы используется правило отбраковки групп неравенств, которые являются «наиболее нарушенными» при выполнении последовательности итераций. На протяжении заданного числа итераций для каждой группы накапливается сумма величин $b_i - \sum_j a_{ij} x_j$, соответствующих неравенствам из группы (при невыполнении неравенств). По прошествии данного числа итерации часть групп (образующих необходимый «процент исключаемых объектов»), имеющих самую большую сумму, помечаются как отбракованные и более не включаются в множества J.

3.6. Линейный дискриминант Фишера

Программа «линейный дискриминант Фишера» (ЛДФ) /63/ является программной реализацией классического метода построения разделяющей гиперплоскости (см. 1.2.1.), существенно расширяющей область его применения.

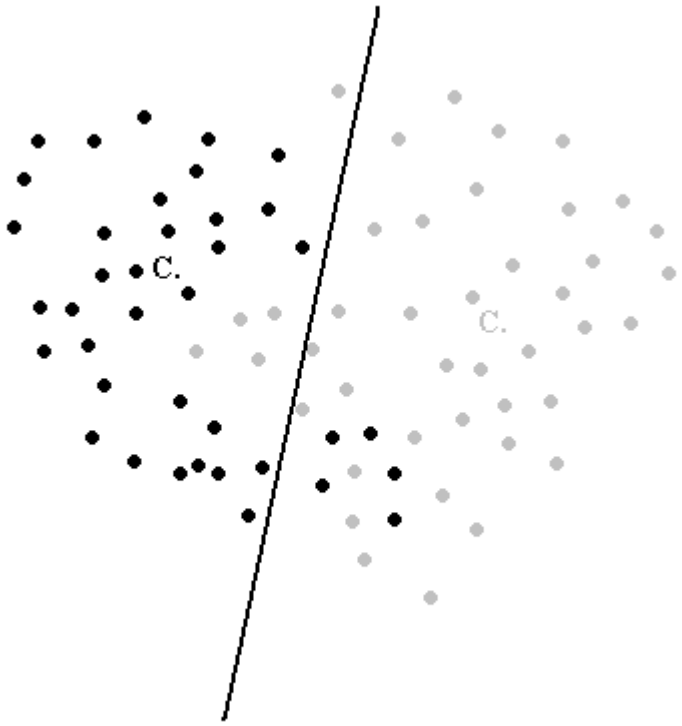

Рис. 24. Разделение классов с помощью гиперплоскости

На Рис. 24 представлен пример разделения классов с помощью линейного дискриминанта Фишера. Заметим, что существенным требованием, предъявляемым к практической задаче, является невырожденность матрицы внутриклассового разброса. В противном случае, классический метод построения ЛДФ неприменим. Для устранения этого ограничения (а также «почти вырожденных» матриц), используется ридж-оценивание матрицы разброса

$$\hat{D}_w = D_w + \beta I,$$

где I – единичная матрица. Такая процедура называется регуляризацией матрицы внутриклассового разброса. Она позволяет устранить вырожденность матрицы, если та была порождена неоднородностью выборки. Кроме того, регуляризация матрицы может улучшить качество распознавания при малых обучающих выборках. При вычислении матриц разброса особое внимание следует обратить на внедиагональные элементы. Последние существенно зависят от коэффициентов корреляции между признаками. В частности, если признаки независимы, то их коэффициенты корреляции равны нулю. В этом случае соответствующие внедиагональные элементы также должны быть нулевыми. Однако, в связи с конечным объемом выборки, вычислить истинные значения корреляций не представляется возможным. Вместо этого вычисляются их оценки, которые часто оказываются отличными от нуля даже для независимых признаков. Из-за этого

получившиеся матрицы разброса могут отражать ложные зависимости между признаками, что, в свою очередь, приводит к перенастройке алгоритма на обучающую выборку и ухудшению качества распознавания. Для устранения данных негативных эффектов используется порог значимости коэффициента корреляции, который можно устанавливать в пределах от нуля до единицы. Если оценка корреляции оказывается по модулю ниже порога, то она считается ложной и игнорируется. Внедиагональные элементы, отвечающие незначимым коэффициентам корреляции, обнуляются. Обычно, порог устанавливают близким к нулю (порядка 0.1 – 0.2), но в некоторых задачах исследователи рекомендуют использовать матрицы разброса только с диагональными элементами. Это соответствует установке порога значимости коэффициента корреляции близкого к единице.

При наличии более чем двух классов будем использовать обобщение ЛДФ. Для каждой пары классов построим свою линейную функцию, а затем проведем голосование. Каждая линейная функция определяет, к какому из двух классов отнести объект, который получает один голос за соответствующий класс. При этом объект будет отнесен к тому классу, за который он получит больше голосов. Очевидно, что при такой схеме может возникнуть ситуация, когда за несколько классов объект наберет одинаковое количество голосов. Если голос будет зависеть от расстояния до гиперплоскости, определяемой вектором $\boldsymbol{w}$ (т.е. чем дальше объект от гиперплоскости, тем больше степень уверенности в принадлежности объекта классу), то ситуация с равенством голосов станет практически невероятной. Такой режим называется «мягкой» классификацией. Его использование также позволяет использовать метод ЛДФ для построения коллективных решений.

3.7. Метод Q ближайших соседей

Данный метод распознавания образов основан на использовании функций расстояния /8/ и кратко описан в разделе 1.2.1. Выбор функции расстояния является естественным инструментом для введения меры сходства (близости) векторных описаний объектов, интерпретируемых нами как точки в евклидовом пространстве. Этот метод классификации оказывается весьма эффективным при решении таких задач, в которых классы характеризуются значительной степенью зашумленности, когда разделяющая поверхность сложна, или классы пересекаются («почти пересекаются»).

Рассмотрим выборку объектов с известной классификацией $\{S_1, S_2, ..., S_m\}$, причем предполагается, что каждый объект выборки входит в один из классов $K_1, K_2, ..., K_l$. Можно определить правило классификации, основанное на принципе *ближайшего соседа*

(БС - правило). Это правило относит классифицируемый объект S к классу, к которому принадлежит его ближайший сосед. Объект $S_i \in \{S_1, S_2, ..., S_m\}$ называется ближайшим соседом объекта S, если

$$D(S_i, S) = \min_j \{D(S_j, S)\},\ j = 1,2,...,m,$$

где D - любое расстояние, определение которого допустимо на пространстве признаковых описаний объектов.

Эту процедуру классификации можно назвать 1 – БС – правилом, так как при ее применении учитывается принадлежность некоторому классу только одного ближайшего соседа объекта S. Аналогично можно ввести Q – БС – правило, предусматривающее определение Q ближайших к S объектов и зачисление его в тот класс, к которому относится *наибольшее число* объектов, входящих в эту группу.

На рис. 25 представлена разделяющая граница для случая двух классов при применении правила одного ближайшего соседа. Можно заметить, что граница, разделяющая классы, является кусочно-линейной. Участки ее границ представляют собой геометрические места точек, равноудаленных от прямых, соединяющих объекты различных классов, находящихся на границе.

Можно показать, что в случае, если все расстояния, разделяющие объекты одного класса, меньше всех расстояний между объектами, принадлежащими различным классам, то 1 – БС – правило работает лучше, чем Q – БС – правило ($Q \neq 1$). Последнее обстоятельство приводит к практической целесообразности использования одного ближайшего соседа взамен нескольких за исключением, быть может, отдельных случаев, когда исследователь обладает определенными знаниями о специфическом характере выборки.

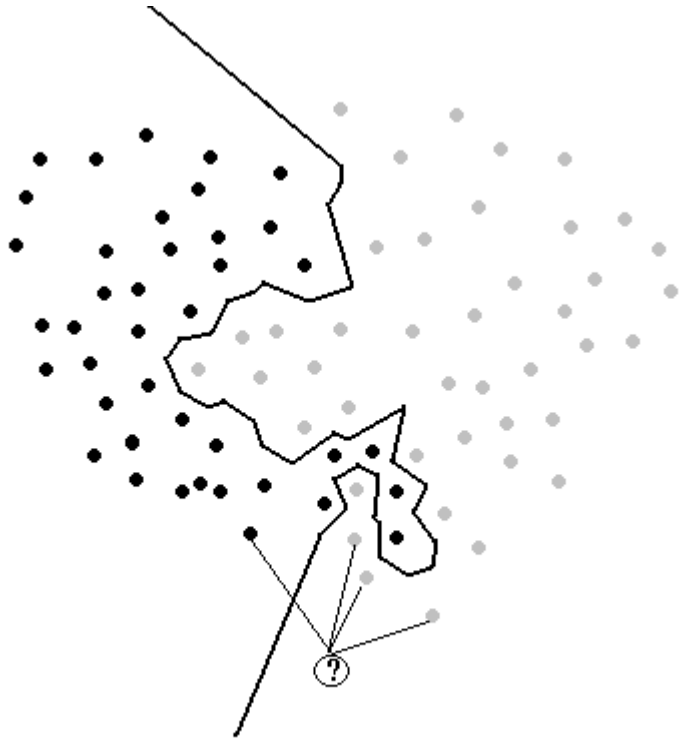

Рис. 25. Граница, разделяющая два класса для случая одного ближайшего соседа

Также можно показать, что в случае выборок большого объема $(m \to \infty)$ и при выполнении некоторых благоприятных условий вероятность ошибки 1 – БС – правила заключена в следующих пределах:

$$p_B \le p_{e_1} \le p_B(2 - \frac{l}{l-1} p_B),$$

где p_B - байесовская вероятность ошибки, т.е. наименьшая вероятность, достижимая в среднем. Данный результат, а также высокая скорость работы метода, позволяют использовать последний в случае выборок большого и сверхбольшого объема.

В том случае, если представительность классов существенно отличается между собой, то возможно использовать модификацию Q - БС – правила таким образом, чтобы близость к объектам малых классов учитывалась в большей степени, чем близость к объектам более представительных классов. Иными словами, вес объекта увеличивается, если он представляет малый класс и уменьшается в противном случае. Данную модификацию следует использовать в тех случаях, когда важность правильного распознавания малых классов является существенной.

3.8. Метод опорных векторов

Современный метод опорных векторов /62/ является развитием метода «обобщенный портрет» /11/ на случаи линейно неразделимых классов и позволяет строить оптимальные линейные или нелинейные разделяющие поверхности.

Использование метода опорных векторов для обучения распознаванию объектов двух классов, представленных обучающей выборкой $\{x_j \in \mathbf{R}^n, g_j = \pm 1, j = 1,\ldots,N\}$ (g_j – «индикатор» класса), предполагает поиск такого направления в пространстве признаков, выражаемого вектором $a \in \mathbf{R}^n$, вообще говоря, произвольной нормы $\| a \|$, чтобы «зазор» между проекциями на него объектов первого и второго классов был максимальным. Если при этом выборка является линейно неразделимой, то «мешающие» объекты сдвигаются к гиперплоскости, причем сумма необходимых сдвигов должна быть минимальной. Предполагается, что граница $b \in \mathbf{R}$ о суждении в пользу первого $a^T x + b > 0$ или второго класса $a^T x + b < 0$ должна проходить ровно посередине зазора между выборками классов, чтобы обеспечить равное и, следовательно, максимальное удаление крайних точек обеих выборок от разделяющей гиперплоскости $a^T x + b = 0$, которая в этом случае называется оптимальной. Такая концепция приводит к задаче квадратичного программирования

$$\begin{aligned} & a^T a + C\sum_{j=1}^{N} \delta_j \to \min, \\ & g_j(a^T x_j + b) \ge 1 - \delta_j, \delta_j \ge 0, \; j = 1,\ldots,N \end{aligned} \tag{3.10}$$

Здесь C - коэффициент штрафа за неправильную классификацию «мешающих» объектов. Задачу (3.10) удобно решать в двойственной форме, используя множители Лагранжа при каждом из ограничений, то есть при каждом объекте обучающей выборки.

$$\begin{cases} W(\lambda_1,\ldots\lambda_N) = \sum_{j=1}^{N} \lambda_j - \sum_{j=1}^{N}\sum_{k=1}^{N} (g_j g_k x_j^T x_k)\lambda_j \lambda_k \to \max \\ \sum_{j=1}^{N} g_j \lambda_j = 0, \; \lambda_j \ge 0, \; j = 1,\ldots,N \end{cases} \tag{3.11}$$

Те из оптимальных значений множителей Лагранжа, которые оказываются положительными, непосредственно определяют направляющий вектор оптимальной

разделяющей гиперплоскости как линейную комбинацию векторов обучающей выборки. Данные вектора называются опорными: $a = \sum_{j:\lambda_j>0} \lambda_j g_j x_j$.

Заметим, что в задаче обучения (3.11) объекты обучающей выборки x_i входят только в форме скалярных произведений $x_i^T x_j$. Это обстоятельство позволяет перенести метод на случай нелинейной разделяющей функции. Предположим, что мы сначала преобразовали данные в некоторое (возможно бесконечномерное) евклидово пространство при помощи функции $\hat{O}: \mathbf{R}^n \to H$. Теперь алгоритм обучения зависит только от скалярных произведений в пространстве H в форме $\hat{O}(x_i) \cdot \hat{O}(x_j)$. Предположим, что в исходном пространстве имеется некоторая «ядровая функция» $K: K(x_i, x_j) = \hat{O}(x_i) \cdot \hat{O}(x_j)$. Тогда оказывается достаточным использовать функцию K в алгоритме обучения, даже не задумываясь о виде функции преобразования пространства $\hat{O}$.

Таким образом, оказывается возможным получение нелинейных разделяющих поверхностей с хорошими дискриминационными свойствами. На Рис. 26 представлен вид такой поверхности для случая двух классов. Квадратами отмечены опорные вектора.

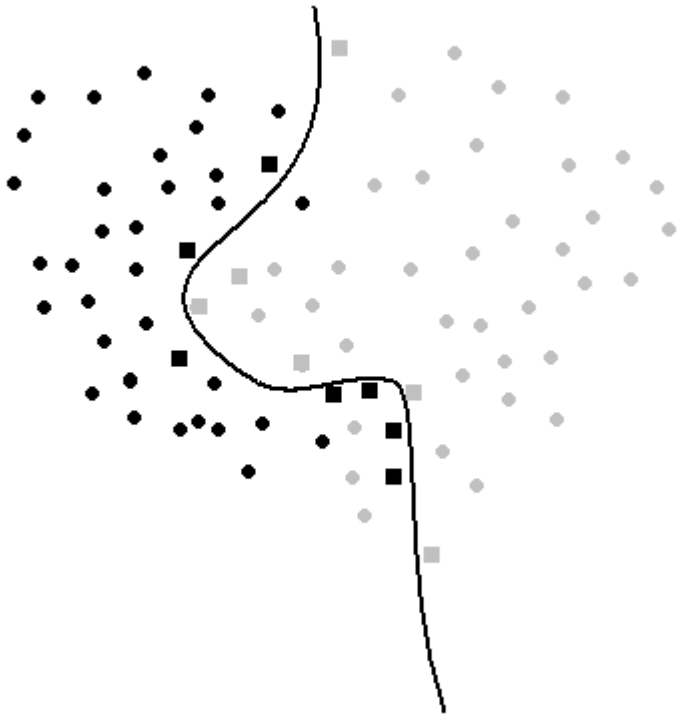

Рис. 26. Разделяющая поверхность для случая двух классов.

В качестве ядровых функций, используемых при решении задачи распознавания образов, выступают:

1. Полином скалярного произведения $K(x, y) = (x \cdot y + 1)^p$
2. Гауссиана $K(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}$
3. Тангенс гиперболический $K(x, y) = \tanh(x \cdot y - \delta)$.

Для случая полинома скалярного произведения параметр p принимает положительные целочисленные значения, начиная с единицы. Для исследования в большинстве задач распознавания образов достаточно ограничиваться двумя-тремя первыми значениями. Для гауссианы параметр σ принимает положительные действительные значения и определяет степень «размытости» потенциального поля, создаваемого опорным объектом. Использование гиперболического тангенса с параметром сдвига δ эмулирует двухслойную нейронную сеть с сигмоидальной функцией активации.

За счет выбора коэффициента штрафа C, оказывается возможным контролировать процесс обучения алгоритма. Так при больших значениях коэффициента, метод пытается сделать как можно меньше ошибок на обучающей выборке, максимально деформируя разделяющую поверхность. Во многих случаях это приводит к перенастройке на обучающую выборку, т.е. к неадекватному виду разделяющей поверхности. Качество распознавания произвольных объектов может оказаться значительно ниже. Если это происходит, то необходимо снизить значение коэффициента C. Таким образом, можно эффективно бороться с перенастройкой алгоритма.

В настоящее время метод опорных векторов является одним из наиболее широко используемых в мире. Выбор нелинейной ядровой функции обеспечивает возможность решения сложных практических задач с плохо-отделимыми в исходном пространстве классами. Наиболее эффективен метод на средних выборках (100 – 1000 объектов). При больших объемах обучающей выборки время обучения метода может оказаться слишком большим, а при малых выборках он может быть подвержен перенастройке.

3.9. Многослойный перцептрон

Многослойный перцептрон /57/ является нейронной сетью с несколькими слоями нейронов: входным, возможно несколькими промежуточными (скрытыми) и выходным слоями. Каждый нейрон промежуточного слоя соединён синапсами со всеми нейронами

предшествующего и последующего слоев (рис. 10). Сеть в системе РАСПОЗНАВАНИЕ может содержать один, два или три скрытых слоев. Также допускается отсутствие скрытых слоев, в этом случае каждый выходной нейрон соединен непосредственно с каждым элементом входного слоя (слоем нейронов-рецепторов).

Количество рецепторов равняется размерности признакового пространства, и на каждый из рецепторов подается (в настоящей реализации) нормализованная величина соответствующего признака классифицируемых объектов.

Во всех оставшихся нейронах (скрытых и выходных) осуществляется взвешенное суммирование входных сигналов, после чего результат обрабатывается активационной функцией и выдается на все выходы (единственный выход для последнего слоя).

Число выходных нейронов совпадает с количеством классов. Величины сигналов выходных нейронов конкурируют в том смысле, что объект относится в класс, сопоставленный тому нейрону, выход которого максимален. Сами величины сигналов выходного слоя можно рассматривать как оценки, даваемые сетью за тот или иной класс.

Перед началом обучения веса синапсов устанавливаются случайным образом, поэтому повторное обучение нейронной сети при тех же параметрах может дать другой результат.

В качестве активационных функций используются гиперболический тангенс или нелинейная функция с насыщением - логистическая функция или сигмоид

$$f(x) = \frac{1}{1+e^{-\alpha x}}.$$

Обучение нейронной сети состоит в поиске наилучших значений весовых коэффициентов и осуществляется с помощью алгоритма обратного распространения. Для каждого эталонного входа сеть порождает выход, который сравнивается с ожидаемым выходом. Величина сигнала выходного нейрона истинного класса должна быть максимальна, а сигналы всех прочих нейронов выходного слоя - минимальны. По результатам сравнения строится функция ошибки от весов синапсов выходного слоя, которая должна быть минимизирована методом наименьших квадратов. Обратным алгоритм называется вследствие итерационной процедуры пересчета весовых коэффициентов. На каждой отдельной итерации пересчет весов осуществляется «справа - налево»: коррекция весов выходного слоя определяется непосредственно через градиент функции ошибок, коррекция коэффициентов предпоследнего уровня вычисляется через величины коррекции коэффициентов последнего уровня, и т.д. до коррекции весов первого уровня. При этом коррекция осуществляется по формулам общего вида

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t),$$

где $\Delta w_{ij}^{(n)}(t) = -\eta \cdot (\mu \cdot \Delta w_{ij}^{(n)}(t-1) + (1-\mu) \cdot \delta_j^{(n)} \cdot y_i^{(n-1)})$. (3.12)

В выражении (3.12) величины $\delta_j^{(n)}$ являются вспомогательными коэффициентами, которые вычисляются при «обратном распространении ошибки», t – порядковый номер итерации, η - параметр программы, именуемый «скорость обучения», μ - параметр обучения, именуемый «коэффициент инерционности». Скорость обучения (которая определяет величину изменения весов) уменьшается при отсутствии сходимости последовательности значений функции ошибки: если нет улучшения значения функции ошибки за u итераций, тогда скорость обучения уменьшается в v раз (u , v - управляющие параметры программы) /54/.

3.10. Методы решения задач распознавания коллективами алгоритмов

Для решения задач распознавания существуют разнообразные подходы. Наиболее известные и широко апробированные на практике представлены в Системе РАСПОЗНАВАНИЕ. Основанные на различных принципах, идеях и моделях, они, естественно, дают вообще говоря различные результаты распознавания/прогноза при решении какой-либо задачи. При решении одной задачи наиболее точным оказывается некоторый один (или несколько) метод (методов). При решении другой задачи ситуация может оказаться «обратной», наиболее точным оказывается неудачный на предыдущей задаче метод. При этом угадать заранее метод-фаворит для новой практической задачи (без проведения предварительных экспериментов) обычно проблематично.

Альтернативой выявлению и практическому использованию одного метода является решение задачи распознавания коллективом распознающих алгоритмов, когда задача решается в два этапа. Сначала задача решается независимо друг от друга всеми или частью из имеющихся алгоритмов. Далее, по полученным решениям вычисляется окончательное «коллективное» решение. Данный подход позволяет надеяться, что при синтезе коллективного решения ошибки отдельных алгоритмов будут компенсироваться правильными ответами других алгоритмов. Действительно, данная гипотеза практически подтверждается, и коллективные решения обычно оказываются наилучшими, или близкими к наилучшим по отдельным методам. Кроме того, данная двухэтапная схема позволяет эффективно решать задачи в автоматическом режиме, что делает доступным применение Системы неквалифицированным пользователем. Основы теории решения задач распознавания коллективами алгоритмов были впервые предложены и разработаны Ю.И.Журавлевым, и кратко описаны в главе 1 /25, 26/). В настоящее время в системе РАСПОЗНАВАНИЕ реализована часть алгоритмов синтеза коллективных решений -

выпуклый стабилизатор (см. 1.4.4.), байесовский корректор и некоторые эвристические методы /67+1/.

3.10.1. Комитетные методы

Одной из наиболее простых и естественных концепций построения коллективного решения является «объединение» результатов распознавания несколькими алгоритмами в «комитетных» конструкциях. В зависимости от того, как именно производится это объединение, различают несколько методов построения комитетов алгоритмов. Вообще говоря, большинство комитетных методов использует оценки апостериорных вероятностей принадлежности объекта к классу, полученные с помощью исходных алгоритмов. Исключением является метод большинства, относящий объект к тому классу, к которому он был присвоен относительным большинством алгоритмов.

Из остальных методов наиболее употребительными являются методы усреднения, взятия минимума, взятия максимума и произведения оценок. Обозначив оценку принадлежности объекта *S* к *k*-ому классу вычисленную *j*-ым алгоритмом как $\Gamma_j^k(S)$, получим следующие формулы подсчета итоговых оценок с помощью комитетных методов.

Пусть имеется *p* исходных обученных алгоритмов распознавания. При усреднении оценок, оценка принадлежности получается как среднее арифметическое оценок за данный класс разных алгоритмов

$$\Gamma_{avr}^k(S) = \frac{1}{p}\sum_{j=1}^{p}\Gamma_j^k(S).$$

При использовании взятия минимума, оценка принадлежности за класс вычисляется как минимум всех оценок за данный класс полученных разными алгоритмами

$$\Gamma_{\min}^k(S) = \min_j \Gamma_j^k(S).$$

При использовании взятия максимума, оценка принадлежности за класс вычисляется как максимум всех оценок за данный класс полученных разными алгоритмами

$$\Gamma_{\max}^k(S) = \max_j \Gamma_j^k(S).$$

Еще одним употребительным способом построения комитетного решения является произведение оценок. В этом случае итоговые оценки принадлежности за класс могут быть получены в виде

$$\Gamma^k_{pro}(S) = \prod_{j=1}^{p} \Gamma^k_j(S).$$

Использование комитетных решений позволяет быстро проводить объединение результатов работы разных алгоритмов распознавания. К их достоинствам относится отсутствие процедуры обучения, что позволяет сразу переходить к распознаванию объектов комитетом обученных алгоритмов. Кроме того, если каждый алгоритм совершает ошибки независимо от других алгоритмов, и вероятность правильного распознавания объекта каждым алгоритмом выше 0.5, то комитетное решение может значительно увеличить качество распознавания. Наиболее эффективным при этом оказывается усреднение оценок. В то же время, если ошибки нескольких алгоритмов коррелируют друг с другом, то качество распознавания может не только не увеличиться, но даже снизиться. В этом случае следует предпочесть другие формы получения коллективных решений.

3.10.2. Метод Байеса

Одним из наиболее распространенных и хорошо зарекомендовавших себя на практике методов получения коллективных решений является метод Байеса. В данном случае для построения коллективного решения предполагается использовать статистические свойства выборки. Допустим, что отдельные алгоритмы комитета являются попарно-независимыми. Пусть имеется p алгоритмов - классификаторов A_j и l классов K_i. Для каждого классификатора A_j вычисляется матрица «отметок» LM^j размерности $l \times l$ путем применения A_j к обучающей выборке. Элементы матрицы представляют собой оценки условных вероятностей:

$$P(K_t \mid A_j(S) = K_u), t, u = 1,2,...,l .$$

Далее, в предположении о независимости классификаторов, оценка $\mu_i(S)$ апостериорной вероятности принадлежности к классу K_i вычисляется как произведение условных вероятностей возникновения i-ого класса при условии, что каждый классификатор отнес текущий объект S к некоторому своему классу K_j, т.е.

$$\mu_i(S) = \prod_{j=1}^{p} P(K_i \mid A_j(S) = K_j),\ i = 1,2,...,l .$$

Метод Байеса обладает высокой скоростью работы и, как следствие, может быть использован в случае большого количества алгоритмов, составляющих комитет.

Ограничения на его применение накладывает требование независимости отдельных классификаторов. Также следует обратить внимание на достаточный объем обучающей выборки для получения адекватных оценок условных вероятностей возникновения классов.

3.10.3. Динамический метод Вудса и области компетенции.

Основной идеей этой группы методов является нахождение для распознаваемого объекта наилучшего в некотором смысле алгоритма из заданного коллектива. Предполагается, что алгоритм может работать по-разному в разных точках пространства. В одних областях алгоритм практически не совершает ошибок, в других показывает посредственные результаты работы. Если удается для каждого объекта определить алгоритм, являющийся наилучшим в окрестности данного объекта, то получившийся алгоритм распознавания будет, по крайней мере, не хуже наилучшего из исходных классификаторов. Для определения наилучшего алгоритма необходимо провести процедуру обучения алгоритма синтеза коллективного решения. Для этого используется контрольная выборка, которая, в частном случае, может совпадать с обучающей. Введем отображения $D: R^n \to \{1,..., f\}$, ставящее в соответствие каждой точке пространства объектов номер соответствующей подобласти, $F: \{1,..., f\} \to \{1,..., p\}$, по которому для каждой подобласти осуществляется выбор соответствующего классификатора, и $E: R^n \to \{1,..., p\}$, которое каждой точке пространства ставит в соответствие номер классификатора. Тогда в общем виде можно записать схему работы полученного алгоритма следующим образом

$$A(S) = A_{E(S)}(S)$$

Агрегирующие алгоритмы этой группы различаются способами выбора отображений *E(.)*. Рассмотрим некоторые из них. Если удается представить отображение *E(.)* в виде декомпозиции $E(S) = F(D(S))$, то говорят о статическом выборе областей компетенции. В противном случае происходит динамический выбор алгоритма распознавания. Среди первой группы методов обычно функция *F(.)* имеет вид $F(j) = \arg\max_{1 \le i \le p} \nu_i(D_j)$, где D_j - соответствующая подобласть, а $\nu_i(D)$ - некоторый показатель эффективности работы (например, доля правильных ответов) *i*-ого алгоритма в области *D*. В качестве отображения *D(.)* обычно используются те или иные методы кластеризации /19/. Соответствующие подобласти называются областями компетенции. В методе областей компетенции, реализованном в программном комплексе

РАСПОЗНАВАНИЕ, используется кластеризация исходного признакового пространства методом *k*-средних. Работа метода существенно зависит от количества областей компетенции, заданного пользователем. При *k=1* имеет место классификация всех объектов наилучшим из исходных алгоритмом. При слишком большом числе областей компетенции возможны многочисленные неоправданные переключения с метода на метод, приводящие к неустойчивой классификации и деградации коллективного решения. Метод областей компетенции отличается высокой скоростью распознавания и относительно небольшим временем обучения /47/.

Одним из подходов при динамическом выборе алгоритма является использование некоторого алгоритма распознавания в качестве *E(.)*. В этом случае получается следующая двухуровневая задача распознавания образов. На первом этапе каждый объект относится к одному из классов исходными алгоритмами, а на втором – объект ассоциируется с одним из этих алгоритмов (который и будет использоваться для окончательного распознавания) с помощью другого алгоритма распознавания.

Другим подходом является определение меры компетенции каждого алгоритма в окрестности заданного объекта, например следующим образом

$$E(S) = \arg\max_{1 \le i \le p} \nu_i(U_\delta(S))$$

где $U_\delta(S)$ - дельта-окрестность объекта *S*. Таким образом, учитываются локальные свойства алгоритмов. Одним из вариантов такого подхода является метод Вудса. Мера локальной компетенции алгоритма в точке подсчитывается следующим образом. Для каждого алгоритма определяется номер класса, к которому он относит рассматриваемый объект. Затем производится подсчет доли правильно распознанных объектов этого класса ближайших к данному объекту. Количество ближайших объектов класса, используемых для оценки компетенции, задается пользователем. Создатели метода рекомендуют использовать для этого порядка 10 объектов.

3.10.4. Шаблоны принятия решений

Данный метод коллективных решений использует подход так называемого слияния алгоритмов распознавания. В рамках данного подхода отдельные методы коллектива рассматриваются не как дополняющие друг друга в различных областях пространства образов, а как конкурирующие между собой. Мы можем рассматривать результаты работы алгоритмов коллектива как входную информацию некоторого классификатора второго уровня в промежуточном признаковом пространстве. Пусть имеется *p*

классификаторов и l классов. Тогда выходы классификаторов первого уровня можно представить в виде матрицы профиля принятия решений:

$$DP(S) = \begin{pmatrix} d_{1,1}(S) \cdots d_{1,j}(S) \cdots d_{1,l}(S) \\ \vdots \qquad\qquad \vdots \\ d_{p,1}(S) \cdots d_{p,j}(S) \cdots d_{p,l}(S) \end{pmatrix}$$

Здесь $d_{i,j}(x)$ - оценка i-ого классификатора объекта S за j-ый класс. Таким образом, исходное признаковое пространство $\mathbf{R}^n$ с n признаками, переходит в новое промежуточное пространство с $p \times l$ признаками.

Метод шаблонов принятия решений предполагает вычисление шаблонов для каждого класса. Шаблон для класса K_i, который обозначим через DT_i, представляет собой центр i-ого класса в промежуточном признаковом пространстве. DT_i можно рассматривать как ожидаемое значение профиля $DP(S)$ для класса K_i. Оценка $\mu_i(S)$ за класс K_i, получаемая по p классификаторам, определяется как мера сходства DT_i и $DP(S)$. В качестве меры сходства используется евклидова метрика. Если выходы классификаторов первого уровня рассматривать как оценки апостериорных вероятностей $P(K_j \mid S), j = 1,2,...,l,$, шаблон принятия решений DT_i представляет собой несмещенную оценку математического ожидания переменной $DP(x)$ размерности $p \times l$ при условии, что правильным классом является K_i.

«Шаблоны принятия решений» считается одним из наилучших методов коллективных решений и обладает устойчивыми характеристиками в большом числе экспериментов.

3.11. Средства контроля качества распознавания.

Проблема оценки качества распознавания является весьма важной с прикладной точки зрения, т.к. при решении реальных задач необходимо иметь объективную оценку вероятности правильного распознавания произвольного объекта. Типичным способом получения такой оценки является предъявление независимой тестовой выборки и подсчет доли правильно распознанных объектов этой выборки. Заметим, что такой способ нельзя применять при малом числе ошибок (т.е. при высоком качестве распознавания), т.к. полученная таким образом оценка будет являться завышенной. В частности, при использовании алгоритма, правильно распознающего все объекты тестовой выборки

(например, корректного) оценка вероятности ошибки распознавания будет нулевой, хотя из факта отсутствия ошибок на конкретной выборке вовсе не следует абсолютная точность алгоритма (т.е. нулевая вероятность неправильного распознавания). В то же время вопрос объективной оценки вероятности правильного распознавания как раз представляет особый интерес в случае хорошего качества работы классификаторов, т.к. позволяет использовать ее для оценки рисков и потерь, связанных с неправильной классификацией.

Разумным средством оценки вероятности некоторого события является построение доверительных интервалов (доверительное оценивание). Заметим, что факт правильного распознавания объектов можно рассматривать как серию испытаний Бернулли. В этом случае для оценки неизвестной величины вероятности успеха по относительной частоте успехов, необходимо решить следующие уравнения относительно p_+ и p_-

$$\sum_{k=0}^{\nu m} C_m^k p_+^k (1-p_+)^{m-k} = 1-\alpha,$$

$$\sum_{k=\nu m}^{m} C_m^k p_-^k (1-p_-)^{m-k} = 1-\alpha,$$

где ν - доля правильно распознанных объектов тестовой выборки, m – объем тестовой выборки, а α - уровень значимости, с которым строится доверительный интервал $[p_-, p_+]$. Последний показывает вероятность того, что искомая вероятность правильного распознавания произвольного объекта будет лежать внутри доверительного интервала. Чем выше уровень значимости, тем шире получается интервал. Так, при $\alpha = 1$ в качестве доверительного интервала будет взята вся область возможных значений вероятности, т.е. отрезок [0,1], поэтому выбор уровня значимости представляет собой компромисс между шириной интервала (т.е. ценностью данной информации) и частотой попадания в него искомой вероятности (т.е. точностью данной информации).

Выписанные выше формулы позволяют получить точные значения границ доверительного интервала в рамках принятой статистической модели. К сожалению, прямой подсчет этих вероятностей наталкивается на большие вычислительные сложности (ошибки округления, большое время работы и т.д.) и при больших объемах выборки практически невозможен. В теории вероятностей получены приближенные выражения для границ доверительного интервала при различных условиях. При достаточно больших m и

ν, существенно отличных от нуля и единицы, можно использовать теорему Муавра-Лапласа, позволяющую приблизить биномиальное распределение нормальным. Для последнего существуют явные формулы, позволяющие определить границы доверительного интервала

$$p_- = \frac{m}{m+z_\alpha^2}\left[\nu + \frac{z_\alpha^2}{m} - z_\alpha\sqrt{\frac{\nu(1-\nu)}{m}+\left(\frac{z_\alpha}{2m}\right)^2}\right]$$

$$p_+ = \frac{m}{m+z_\alpha^2}\left[\nu + \frac{z_\alpha^2}{m} + z_\alpha\sqrt{\frac{\nu(1-\nu)}{m}+\left(\frac{z_\alpha}{2m}\right)^2}\right]$$

где z_α является соответствующей квантилью нормального распределения, которая может быть найдена из уравнения $\Phi_0(z_\alpha) = \frac{\alpha}{2}$ при помощи таблиц.

При значениях частоты ошибок близкой к нулю или единице использовать нормальное приближение нельзя. В этом случае можно воспользоваться малой предельной теоремой (теорема Пуассона), которая позволяет приблизить биномиальное распределение пуассоновским. В этом случае для определения границ доверительного интервала достаточно решить уравнения относительно λ_- и λ_+

$$\sum_{k=0}^{\nu m} e^{-\lambda_+}\frac{(\lambda_+)^k}{k!} = 1-\alpha,$$

$$\sum_{k=\nu m}^{m} e^{-\lambda_-}\frac{(\lambda_-)^k}{k!} = 1-\alpha.$$

Границы доверительного интервала в этом случае равны соответственно λ_-/m и λ_+/m. В программном комплексе РАСПОЗНАВАНИЕ реализованы все упомянутые выше подходы с автоматическим выбором наиболее подходящего способа подсчета границ доверительного интервала для данной задачи. Пользователь может выбрать тот или иной уровень значимости, варьируя ширину доверительного интервала.

3.12. Минимизация признакового пространства в задачах распознавания

Стандартная постановка задачи распознавания предполагает, что начальная информация о классах (обучающая информация) задается выборкой векторных признаковых описаний объектов, представляющих все классы. Во многих случаях система признаков формируется "стихийно". В ее состав включаются все показатели, влияющие на

классификацию (хотя бы чисто гипотетически), и которые можно вычислить или измерить. Независимо от числа имеющихся признаков, исходная система признаков, как правило, избыточна и включает признаки, не влияющие на классификацию или дублирующие друг друга. В некоторых практических задачах распознавания затраты на вычисление части признаков могут быть значительными и конкурировать со стоимостью потерь при распознавании. Решение задач обучения при меньшем числе признаков также является более точным, а полученные решения более устойчивыми. Таким образом, решение задач минимизации признаковых пространств является важным во многих отношениях.

Рассмотрим задачу минимизации признакового пространства в следующей постановке. Пусть имеется модель М алгоритмов распознавания, признаковое пространство $X_1, X_2, \ldots, X_N$ и критерий качества $f(A)$ есть доля правильно распознанных объектов контрольной выборки алгоритмом A. Требуется найти такое признаковое подпространство $X_{i_1}, X_{i_2}, \ldots, X_{i_n}$, с минимальным n, для которого $f(A) \geq f_0$, где f_0 - некоторый минимально допустимый порог точности алгоритма распознавания A, построенного по данным обучения для данного подпространства.

В силу своего комбинаторного характера, методы перебора значительного числа различных признаковых подпространств практически нереализуемы, поэтому обычно используются процедуры последовательного выбора из системы k признаков подсистемы из $k-1$ признака. Здесь используются различные общие подходы (последовательный отброс наименее информативных признаков, использование кластеризации признаков, и т.п.). Специальные подходы отбора и преобразования признаков имеются в статистической теории распознавания. Многие модели распознавания включают и свои специфические способы оценки и отбора признаков.

В настоящем разделе рассматривается метод минимизации признакового пространства, ориентированный на модели частичной прецедентности /25, 26/ и основанный на кластеризации признаков с учетом их информативности и коррелированности. Рассмотрим один подход, основанный на использовании алгоритмов голосования по системам логических закономерностей /71/.

Пусть P - некоторое множество логических закономерностей, найденное по данным обучения.

Определение 9. Величина $wei(i) = N(i)/|P|$ называется мерой информативности признака (информационным весом признака) X_i, если $N(i)$ - число логических закономерностей множества P, содержащих признак X_i.

Пусть *N(i,j)* - число одновременных вхождений признаков X_i, X_j в одну закономерность по множеству P. Величину $Lcorr(i,j) = 1 - \frac{N(i,j)}{\min(N(i), N(j))}$ назовем логической корреляцией признаков X_i и X_j . Данная величина равна нулю, когда во всех закономерностях, куда входит признак X_i, присутствует X_j (и наоборот), т.е. признаки "дополняют друг друга". Корреляция равна единице, если ни в одну закономерность с признаком X_i не входит X_j. Отметим, что при выборе критериев $\phi(P_t)$ согласно /76/ равным признакам будет соответствовать единичная корреляция. В случаях равных или пропорциональных признаков (столбцов таблицы обучения), в силу свойств логических закономерностей N_{ij}=0 (что непосредственно следует из алгоритма их поиска) и, следовательно, $Lcorr(i,j) = 1$.

Если ***min***(N_i ,N_j)=*0*, полагаем $Lcorr(i,j)$ =0 (данный случай возникает , например, если x_i(S)≡*const*).

Рассмотрим задачу нахождения таких кластеров признаков, для которых входящие в них признаки обладают близкими корреляционными свойствами. В качестве меры корреляционной близости признаков рассмотрим более "тонкий" критерий чем $1 - Lcorr(i,j)$, а именно, основанный на полуметрике

$$r(i,j) = \sum_{l=1, l\neq i,j}^{N} \left| Lcorr(i,l) - Lcorr(j,l) \right| + (N-2) \times (1 - Lcorr(i,j)) .$$

Первое слагаемое показывает насколько близки признаки по отношению к другим признакам, а второе – насколько они «схожи» между собой. Множитель *N-2* добавлен для того, чтобы слагаемые были соразмерны и вносили примерно одинаковый вклад в определение близости между признаками.

В качестве алгоритма кластеризации для заданной полуметрики $r(i,j)$ и фиксированного числа кластеров использовалась иерархическая группировка /19/, в которой расстояние между кластерами определялось согласно функции

$$r(K_p, K_q) = \max_{i \in K_p, j \in K_q} (r(i,j)) .$$

После нахождения *n* кластеров в сокращенную подсистему признаков включаются наиболее информативные признаки (по одному из каждого кластера).

Таким образом задача минимизации признакового пространства решается следующим образом. Предполагается, что для исходного признакового пространства выполнено $f(A) \geq f_0$. Вычисляется матрица значений $r(i,j)$.

Пусть на некотором шаге *k=0,1,2,...* получено с помощью кластеризации *N-k* группировок признаков а из каждой группировки выбран признак с максимальным весом. В результате будет получено признаковое подпространство $X^{N-k} = \{X_{i_1}, X_{i_2}, ..., X_{i_{N-k}}\}$. Пусть A_{N-k} - построенный в данном подпространстве алгоритм распознавания. В качестве решения задачи минимизации признакового пространства принимается $X^{N-k} = \{X_{i_1}, X_{i_2}, ..., X_{i_{N-k}}\}$, соответствующее максимальному *k* при ограничении $f(A_{N-k}) \geq f_0$.

На Рис.27 приведены графики изменения точности распознавания в модели голосования по системам логических закономерностей при двух подходах к минимизации признакового пространства на примере задачи распознавания состояния ионосферы /83/. Исходное признаковое пространство включало 34 признака, задача распознавания решалась относительно двух классов, обучающая выборка имела длину 181 объектов, контрольная - 170. Черная линия соответствует последовательному отсеву менее информативных признаков, серая - минимизации признакового пространства согласно изложенному выше алгоритму. Видно, что серая линия лежит, как правило, ниже черной. "Волнистость" графиков $f(A)$ является естественным следствием набора факторов (неидеальность процедур вычисления предикатов $P_t(S)$, малая длина выборок, "частичная несогласованность" выборок, когда информативность признака на обучающей таблице и контрольной имеет некоторое различие). Из рисунка следуют естественные качественные выводы о данной задаче распознавания. Удаление первой трети малоинформативных признаков мало влияет на точность распознавания и не зависит от используемого метода их сокращения - оставшиеся 20 признаков вполне компенсируют отсутствие остальных 14. При удаление последующих 10 потери при кластеризационной минимизации растут меньше, чем при частотной.

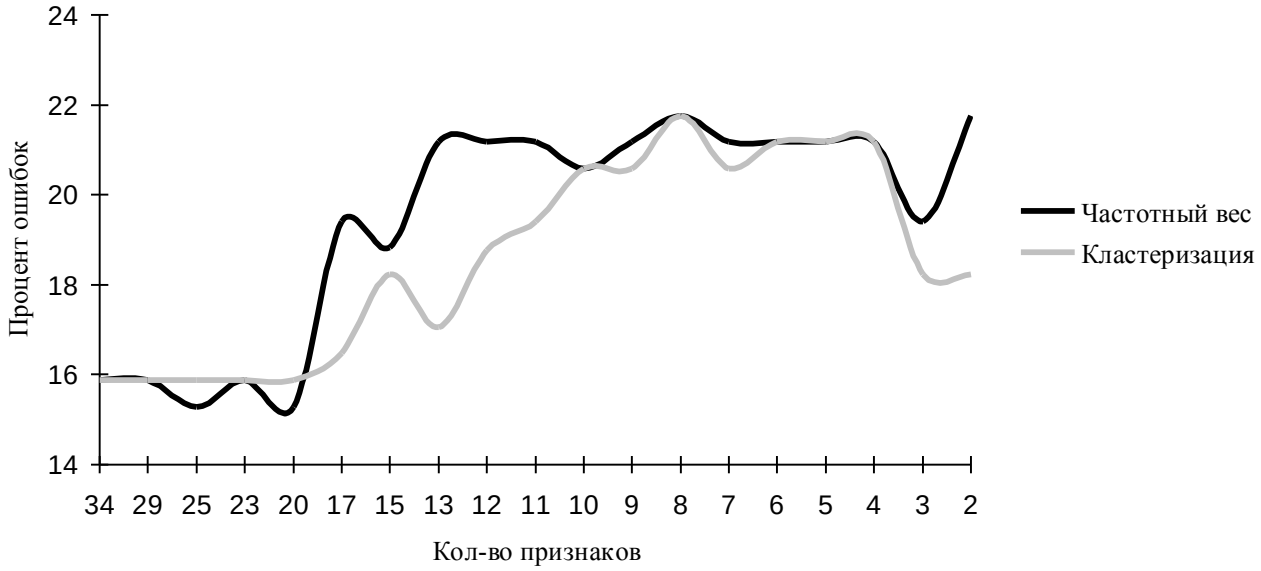

Рис.27. Минимизация признакового пространства на примере задачи распознавания состояния ионосферы

3.13. Алгоритмы кластерного анализа

В настоящем разделе приводятся дополнительный материалы к описаниям реализованных в системе РАСПОЗНАВАНИЕ алгоритмов кластерного анализа, изложенным в главе 2. Коллективное решение задачи кластерного анализа находится согласно подходу, описанному в 2.4.

Алгоритм иерархической группировки реализует стандартную схему иерархической кластеризации на заданное число кластеров. В качестве функций $d(T_i, T_j)$ расстояния между группировками объектов реализованы два «противоположных» подхода:

1. $d_{\min}(T_i, T_j) = \min_{S_\nu \in T_i, S_\mu \in T_j} \rho(S_\nu, S_\mu)$;

2. $d_{\max}(T_i, T_j) = \max_{S_\nu \in T_i, S_\mu \in T_j} \rho(S_\nu, S_\mu)$.

При использовании первого подхода ближайшим соседом каждого объекта будет объект из того же кластера. Это главное свойство данного подхода. При выполнении данного свойства, тем не менее, в пределах одного кластера могут быть весьма далекие объекты, менее близкие, чем некоторые объекты из различных кластеров.

Наоборот, при использовании второго подхода центральным условием алгоритма является создание таких кластеров, в которых не будет «очень непохожих» объектов. Здесь близость объектов разных кластеров отходит на второй план, главное – построение кластеров с минимальными диаметрами.

Когда размеры кластеров существенно больше расстояния между ними, алгоритм «ближайший сосед» формирует кластеры из малого числа объектов или вообще

состоящие из единичных точек. Данный алгоритм удобен для поиска «выбросов» в данных.

Алгоритм *k* внутригрупповых средних является одним из наиболее известных методов кластеризации. Алгоритм находит такие кластеры, для объектов которых центр «своего кластера» будет ближе центра любого «чужого кластера». Алгоритм состоит из следующих шагов.

Шаг 1. Выбираются k исходных центров кластеров $\mathbf{y}_1(1), \mathbf{y}_2(1), ..., \mathbf{y}_k(1)$. Этот выбор производиться произвольно, например, первые k результатов выборки из заданного множества объектов.

Шаг l. На l-м шаге итерации заданное множество объектов $X = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m\}$ распределяется по k группировкам по следующему правилу:

$$\mathbf{x} \in T_j(l), \text{ если } \left\|\mathbf{x} - \mathbf{y}_j(l)\right\| < \left\|\mathbf{x} - \mathbf{y}_i(l)\right\|$$

для всех $i = 1,2,...,k$, $i \neq j$, где $T_j(l)$ - множество образов, входящих в кластер с центром $\mathbf{y}_j(l)$. В случае равенства решение принимается произвольным образом.

Шаг l+1. На основе результатов шага l определяются новые центры группировок $\mathbf{y}_j(l+1)$, $j = 1,2,...,k$, исходя из условия, что сумма квадратов расстояний между всеми объектами, принадлежащими множеству $T_j(l)$, и новым центром данной группировки должна быть минимальной.

Центр $\mathbf{y}_j(l+1)$, обеспечивающий минимизацию $J_j = \sum_{\mathbf{x} \in T_j(l)} \left\|\mathbf{x} - \mathbf{y}_j(l+1)\right\|^2$, $j = 1,2,...,k,$ является выборочным средним, определенным по множеству $T_j(l)$. Следовательно, новые центры кластеров определяются как

$$\mathbf{y}_j(l+1) = \frac{1}{N_j} \sum_{x \in T_j(l)} x, \; j = 1,2,...,k,$$

где N_j - число выборочных объектов, входящих в множество $T_j(l)$. Очевидно, что название алгоритма «*k* внутригрупповых средних» определяется способом, принятым для последовательной коррекции назначения центров кластеров.

Равенство $\mathbf{y}_j(l+1) = \mathbf{y}_j(l)$ при $j = 1,2,...,k$ является условием сходимости алгоритма, и при его достижении выполнение алгоритма заканчивается. Полученные множества $T_j(l)$, $j = 1,2,...,k$, и образуют искомые кластеры. В противном случае последний шаг повторяется.

Качество работы алгоритмов, основанных на вычислении k внутригрупповых средних, зависит от числа выбираемых центров кластеров, от последовательности осмотра объектов и, естественно, от геометрических особенностей данных.

Алгоритмы «итеративная оптимизация» и «метод локальной оптимизации» являются близкими методами нахождения локально-оптимальных разбиений данных на заданное число кластеров в результате минимизации критерия «сумма внутриклассовых дисперсий, или сумма квадратов ошибок» (см. раздел 2.1.).

В методе локальной оптимизации в качестве функций расстояния используются эвклидова метрика, «максимальное отклонение признака» и «сумма отклонений признаков». Строится последовательность кластеризаций, каждая кластеризация получается из предыдущей путем переноса некоторого объекта из одного класса в другой так, чтобы критерий качества монотонно уменьшался. Кроме того, в данном методе имеется возможность нахождения оптимального числа кластеров.

Рассмотрим данный алгоритм с критерием качества «сумма квадратов отклонений» при заданном фиксированном числе кластеров k .

a) Сначала проводится предварительное разбиение выборки объектов на группы. Выбираются k наиболее удаленных точек и объекты распределяются в группы, как множества объектов, для которых будет ближайшей одна из выделенных точек. Функция близости вычисляется по указанной пользователем метрике.

b) Затем проводится итеративная оптимизация функционала штрафа — суммы внутриклассовых разбросов $J = \sum_{p=1}^{k} J_p$, $J_p = \frac{1}{|T_p|} \sum_{\mathbf{x}_i \in T_p} \rho(\mathbf{x}_i, \mathbf{y}_p)^2$, где $\boldsymbol{y}_p$ — центр p-й группы T_p, к которой отнесен объект $\mathbf{x}_i$. J_p равен среднему квадрату расстояния от объектов, отнесенных в p-ю группировку, до его центра (внутриклассовому разбросу). На каждой итерации выбирается группировка T_p, объект $\mathbf{x}_i$ и группировка T_q такие, что при переносе объекта $\mathbf{x}_i$ из T_p в T_q функционал J уменьшится на максимальную величину. Процесс завершается, когда никакой последующий перенос не уменьшает функционал (получен локальный минимум) или достигнуто указанное пользователем максимальное число итераций. Полученные группировки объектов T_p , $p=1,2,...,k,$ и считаются искомыми кластерами.

Алгоритм нахождения оптимального числа кластеров в заданном диапазоне от a до b состоит в следующем. Вначале, меняя число кластеров k в цикле от a–1 до b+1 производим кластеризацию и сохраняем полученное значение критерия J^k и соответствующее распределение объектов по k кластерам. Затем строим оценки

«качества» данных кластеризаций с учетом числа кластеров. Эти оценки основаны на эвристическом критерии поиска точки, в которой наиболее сильно падает скорость уменьшения функционала *J*.

Обозначим $\Delta_i = J^{i+1} - J^i$. Теперь введем следующие понятия:

«взвешенная левая производная»

$$l_k = \sum_{i=a}^{k-1} \frac{\Delta_i}{2^{k-i}} + \frac{\Delta_{a-1}}{2^{k-a}},$$

«взвешенная правая производная»

$$r_k = \sum_{i=k}^{b-1} \frac{\Delta_i}{2^{i-k+1}} + \frac{\Delta_b}{2^{b-k}},$$

в которых берутся сумма значений дискретной производной Δ_i слева (справа) от рассматриваемой точки k с экспоненциально убывающими весами при удалении от точки k так, чтобы сумма этих весов с обеих сторон была равна 1. Назовем функционалом качества разбиения на k кластеров величину относительного изменения взвешенных производных $\frac{l_k}{r_k}$ и будем выбирать то разбиение, для которого эта величина максимальна.

3.14. Визуализация многомерных данных

При решении задач распознавания, классификации и анализа данных важное значение имеет наличие средств визуализации многомерных данных, позволяющих наглядно получать представление о конфигурации классов, кластеров и расположении отдельных объектов. Данные средства необходимы прежде всего в случае задач с большим числом признаков, когда отдельные проекции в 2-3-х мерных подпространствах признаков содержат мало информации относительно n-мерных описаний, или в случаях бинарных и к-значных признаков. Данная задача рассматривалась в следующей широко известной постановке /19/.

Пусть в n - мерном евклидовом пространстве задан набор из m элементов $\mathbf{x}_i \in R^n, \quad i = 1 \ldots m.$ Требуется найти отображение этого набора точек на плоскость R^2 так, чтобы метрические соотношения между образами точек на плоскости максимально соответствовали бы метрическим соотношениям между ними в исходном n-мерном признаковом пространстве: «близкие» («далекие») n- мерные точки, остались бы «близкими» («далекими») на плоскости. Данную искомую плоскость будем называть плоскостью обобщенных признаков (параметров).

Пусть $\mathbf{y}_i$ – отображение элемента $\mathbf{x}_i$ на R^2, δ_{ij} – расстояние между элементами $\mathbf{x}_i$, $\mathbf{x}_j$ в R^n, а d_{ij} – расстояние между $\mathbf{y}_i$, $\mathbf{y}_j$ в R^2. Будем искать такое отображение, для которого сумма различий расстояний между точками будет минимальна

$$J(y) = \sum_{i,j} \left(\delta_{ij} - d_{ij}\right)^2 \to \min.$$

Так как функция J содержит только расстояния между точками, она инвариантна к жесткому передвижению всей конфигурации.

Минимизация функции J проводится с помощью стандартной процедуры градиентного спуска $\mathbf{y}^{i+1} = \mathbf{y}^i - s\nabla J(\mathbf{y}^i)$, где $\mathbf{y}$ - конфигурация точек на плоскости, i - номер итерации, $\nabla J(\mathbf{y}^i)$ - значение градиента функции J, s - шаг спуска. В качестве начальной конфигурации $\mathbf{y}^0$ берется проекция точек $\mathbf{x}_i$ на некоторую плоскость. Шаг спуска s меняется согласно методу «удвоения»: шаг предыдущей итерации или последовательно умножается либо делится на 2 до тех пор, пока наблюдается уменьшение функции $J(\mathbf{y}^{i+1})$. Градиент вычисляется по формуле:

$$\nabla J_i(\mathbf{y}) = -\sum_j \left(\delta_{ij} - d_{ij}\right)\frac{\mathbf{y}_i - \mathbf{y}_j}{d_{ij}}.$$

При больших m затраты машинного времени могут быть практически неприемлемы, при этом может не существовать адекватного отображения исходной конфигурации на плоскость, поэтому количество исходных элементов случайным образом уменьшается до некоторого числа $n_1 = \frac{C}{m}$, где C - подобранная экспериментально константа. На рис. 28, 29 приведены проекция некоторой обучающей выборки с k – значными признаками и ее визуализация на плоскости обобщенных признаков.

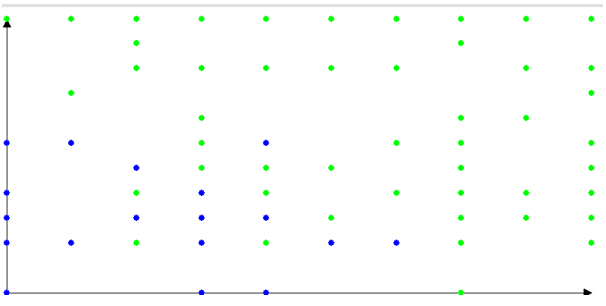

Рис. 28. Проекция данных выборки breast_learn на плоскость признаков №1, 6.

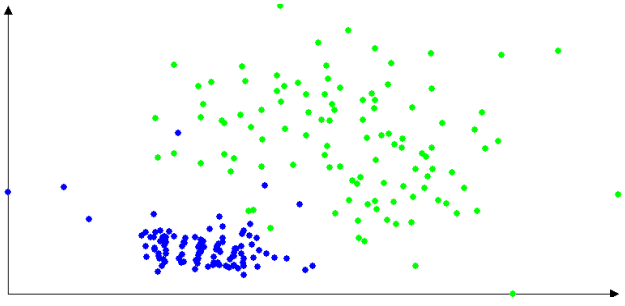

Рис. 29. Проекция данных выборки breast_learn на плоскость обобщенных признаков.

3.15. Использование методов распознавания при прогнозировании временных рядов.

В различных областях практической деятельности нередко возникает задача предсказания значения переменной Z в момент времени $t+1$ по величине этой переменной в предшествующие моменты времени $t, t-1, t-2, \ldots$. Данная задача является частным случаем более общей задачи предсказания значения некоторой переменной Z в момент времени $t+1$ по значениям переменных (k признаков) из множества $\tilde{\mathbf{Z}}$ в предшествующие моменты времени $t, t-1, t-2, \ldots$. Причем множество $\tilde{\mathbf{Z}}$ может содержать саму переменную Z. Для решения данной задачи разработан достаточно широкий спектр моделей и методов, включая модели выделения основных трендов, скользящего среднего, авторегрессий и др. Однако стремление повысить точность необходимого во многих областях краткосрочного прогноза заставляет разрабатывать новые математические средства решения этой задачи. Одним из возможных подходов здесь также является применение распознавание образов.

Для многих практических задач точный прогноз величины Z невозможен, однако цели прогнозирования оказались бы частично достигнутыми, если бы удалось указать направление изменения Z между моментами t и $t+1$. В качестве примера можно указать задачу прогноза динамики курсовой стоимости акций на фондовом рынке. Выбор направления изменения фактически является задачей отнесения ситуации, сложившейся к моменту времени t, к двум классам: K_1- последующий рост величины Z в момент времени $t+1$, K_2- последующее снижение величины Z в момент времени $t+1$. В качестве прогностических переменных (признаков) в данном случае выступают величины переменных из множества $\tilde{\mathbf{Z}}$ в моменты $t, \ldots, t-l$, где l - длина временного интервала, используемого для прогноза. Иными словами ситуация может быть описана с помощью вектора-описания $S = [Z_1(t-l), \ldots, Z_1(t), \ldots, Z_k(t-l), \ldots, Z_k(t)]$.

Имея в своем распоряжении результаты наблюдений за изменениями переменных на некотором временном отрезке $[1,2,\ldots,T]$, где $T >> l$, мы можем построить выборку прецедентов $S_1, S_2, ..., S_m, y(S_1), y(S_2), ..., y(S_m)$, $m = T - l - 1$ - полное число вхождений временных отрезков длины l в отрезок $[1,2,\ldots,T-1]$. Описание объекта S_i задается вектором $[Z_1(t_i - l), \ldots, Z_1(t_i), \ldots, Z_k(t_i - l), \ldots, Z_k(t_i)]$, где $t_i = T - i$, $i = 1,2,...T-l-1$. Величина $y(S_i)$ считается равной 1 если $Z(t_i + 1) > Z(t_i)$, и $y(S_i)$ считается равной 2 если $Z(t_i + 1) \le Z(t_i)$. Данная выборка может рассматриваться в качестве обучающей выборки для построения распознающего алгоритма, относящего вектор

$[Z_1(t-l),\ldots,Z_1(t),\ldots,Z_k(t-l),\ldots,Z_k(t)]$ при произвольном моменте времени t к классу K_1, что соответствует прогнозу роста Z к моменту времени $t+1$, или к классу K_2, что соответствует прогнозу снижения Z к моменту времени $t+1$.

Естественным образом данная постановка обобщается на случай прогнозирования степени увеличения (уменьшения) прогнозируемой величины как задача распознавания с числом классов большим двух.

Глава 4. Практические применения

В настоящей главе приводятся примеры практического применения математических методов теории распознавания и интеллектуального анализа данных в различных предметных областях. Рассматриваемые прикладные задачи были исследованы с различной степенью глубины. Многие работы выполнялись в рамках долгосрочных договоров с соответствующими организациями и при их активном участии. В данных случаях были получены результаты, максимально адекватные тому объему знаний, который в принципе доступен к извлечению из выборок прецедентов. В основном, данные результаты практических применений опубликованы в научной печати и доложены на конференциях. Значительная часть данных была взята из открытых источников (публикации, Интернет) или была предоставлена авторам коллегами. Полученные в данных случаях результаты «разовых расчетов» являются, как правило, «поверхностными», точность прогноза для ряда задач была невысокой. Как правило, подобные результаты можно существенно улучшить, уточнить и доработать при более детальном ознакомлении с предметной областью или (тем более) совместном решении данных задач с их постановщиками. Тем не менее, авторы посчитали целесообразным привести результаты и подобных «микроисследований» с иллюстративной целью максимального охвата области практических применений и демонстрации возможностей обработки «сырого» материала.

Следует отметить очевидную истину: точность распознавания и прогноза, выявленные закономерности непосредственно зависят от практической постановки задачи, качества и количества имеющихся данных. Если не существует в действительности детерминированной или статистической связи между имеющейся системой признаков и распознаваемым свойством (параметром, характеристикой, объектом, ситуацией, и т.п.), то наивно рассчитывать найти то, чего не существует. Если обучающие данные не представительны (т.е. к распознаванию могут предъявляться в значительном количестве объекты, существенно отличающиеся от наблюдаемых ранее), то более правильным ответом в данных случаях будут отказы от распознавания в виде «распознаваемый объект является незнакомым наблюдением» чем необоснованная их классификация.

4.1. Приложения в области бизнеса, экономики и финансов

Для данной области приложений характерно быстрое появление новых проблем, которые отсутствовали в недалеком прошлом, но решение которых может быть непосредственно связано с достижением существенного финансового или экономического

эффекта. Данные практические задачи в силу своей новизны, как правило, еще не имеют точных математических моделей для их решения. К числу подобных примеров можно отнести задачу прогноза курса акций предприятий, оценки надежности клиента при кредитовании, анализа продаж товаров в супермаркетах, и многие другие. Приведем примеры данных приложений.

4.1.1. Оценка стоимости квартир.

Задача состоит в автоматической оценке стоимости квартир по ее внутренним и внешним характеристикам (жилая площадь, строительный материал дома, местонахождение, этаж, удаленность от станции метро, и др.). Применение методов оценки стоимости жилья по имеющимся выборкам прецедентам (совокупность расширенных описаний некоторого множества квартир плюс цена их продаж) позволяет проводить беспристрастную, независимую и точную оценку стоимости. Под расширенным описанием квартиры понимается весь стандартный комплекс параметров, которые обычно являются ценообразующими.

Примером подобной постановки является оценка стоимости жилья в пригородах Бостона /66/. Задача автоматической оценки стоимости жилья решается как задача распознавания интервала его стоимости (очень низкая, низкая, средняя, выше средней, высокая). В качестве признаков используются 13 экологических, социальных, технических показателей: число жилых комнат, доля чернокожего населения в районе, среднее расстояние до основных супермаркетов, качество воздуха, и др. Для обучения использована выборка из 242 объектов, для контроля – выборка из 264 объектов.

Точность распознавания составила 77%, причем практически все ошибки были связаны с отнесением объекта в соседний класс, что естественно в силу искусственного разделения на классы. Количество грубых ошибок (отнесение не в соседний класс) составило менее 1%. Примером логической закономерности класса наиболее дешевого жилья является конъюнкция (6.63724<= «концентрация окислов нитратов/10000000»)&(1.1296<= «среднее расстояние до пяти центров занятости Бостона» <=2.44939) & (0.32<= «Признак B» <=13.692) & (1.73<= «Процент населения низшего статуса»), выполненная на 12 из 16 эталонах первого класса. «Признак B» определяется выражением 1000(Bk - 0.63)^2, где Bk есть доля чернокожего населения.

4.1.2. Оценка состояния предприятий кондитерской промышленности по комплексу финансовых показателей и структуре рабочего персонала

Рассматривалась задача оценивания эффективности работы предприятий. Данная задача решалась экспертами для группы из 89 предприятий на основе детального изучения финансово-экономической деятельности данных предприятий. Основная задача состояла в автоматической оценке по набору признаков остальных предприятий отрасли, используя результаты исследованных предприятий в качестве эталонной выборки. Использовалась трехбальная шкала оценки (три класса).

Для описания предприятий использовались значения по каждому кварталу следующих 9 показателей.

1. Объем продукции (работ, услуг) в действующих (отпускных, договорных) ценах предприятия.
2. Балансовая прибыль.
3. Среднегодовая стоимость промышленно-производственных основных фондов.
4. Прибыль, остающаяся в распоряжении предприятия (чистая прибыль).
5. Затраты на 1 рубль товарной продукции в действующих ценах, в копейках.
6. Среднесписочная численность промышленно-производственного персонала.
7. Сумма фондов заработной платы и материального поощрения промышленно-производственного персонала.
8. Среднесписочная численность рабочих, человек.
9. Сумма фондов заработной платы и материального поощрения рабочих.

В список наиболее информативных признаков по всем 4 кварталам попал признак «затраты на 1 рубль товарной продукции в действующих ценах», причем его значение за третий квартал оказалось самым информативным относительно всех других. Близкими к наиболее информативным признакам оказались «среднегодовая стоимость промышленно-производственных основных фондов» за третий квартал и «сумма фондов заработной платы и материального поощрения промышленно-производственного персонала» - за четвертый. Наименее важными для оценивания оказались признаки №1, 6, 7 за первый квартал.

Точность распознавания в режиме скользящего контроля составила 75%, причем не было ни одной грубой ошибки, когда успешное предприятие оценивалось отрицательно, и наоборот.

4.1.3. Подтверждение кредитных карточек

Данная информация была взята из открытой базы данных по адресу http://www.isc.uci.edu/~mlearn/MLRepository.html, источник информации - quinlan@cs.su.oz.au.

Описание кредитных карточек основывалось на 15 веществennозначных и к-значных (которых было две трети от общего количества) признаков, причем содержание каждого признака автором информации не раскрывалось. Для обучения в данной задаче распознавания с двумя классами использовано 342 эталона. Точность распознавания на 348 контрольных объектах составила 86%.

4.1.4. Кластеризация продукции автомобильного рынка

Для анализа современного автомобильного рынка была проведена кластеризация выборки 99 автомобилей с кузовом типа «седан» отечественных и импортных марок. Описание каждой модели включало значения 11 технических признаков, наиболее существенных с позиций покупателя – объем двигателя, масса, ускорение, количество цилиндров, тип коробки передач и другие. Оптимальное число кластеров для данной задачи оказалось равным четырем, причем полученные кластеры имеют естественную интерпретацию.

- Автомобили ручной сборки - Rolls Royce, Maybach, Bentley.
- Представительские - автомобили марок BMW, Lexus, Lincoln, Pontiac, S-класс марки Mercedes Benz.
- Средний класс – E-класс марки Mercedes Benz, большинство автомобилей Opel, Toyota, Subaru, Nissan, Волга.
- Малолитражные и дешевые автомобили – ВАЗ, Fiat, Seat, Skoda, Daewoo.

Настоящие результаты кластеризации позволяют прогнозировать восприятие автомобиля на рынке потенциальным покупателем по совокупности приведенных параметров, что может служить ориентиром для производителя по оптимизации отдельных характеристик автомобиля и обоснования его рыночной стоимости.

4.1.5. Оценка надежности клиента при выдаче кредита

Одной из актуальных прикладных задач предсказания в области экономики является задача оценивания целесообразности выдачи банковского кредита частному лицу. Обучающая выборка включала представителей двух классов – класса выгодных клиентов и класса невыгодных клиентов. Выдача каждого кредита характеризуется 26 признаками, такими как размер кредита, размер текущего банковского счета заемщика, назначение кредита, заработная плата заемщика, его семейное положение и другими. Обучающая выборка содержала информацию о 1000 выданных кредитов, 700 из которых были выплачены своевременно (класс №1), а 300 – с той или иной задержкой (класс №2). Точность распознавания в режиме скользящего контроля составила 75%.

Примечание. Автор постановки задачи и данных -

Professor Dr. Hans Hofmann, Institut f"ur Statistik und "Okonometrie Universit"at Hamburg, FB Wirtschaftswissenschaften, Von-Melle-Park, 5, 2000, Hamburg, 13

4.1.6. Распознавание сортов вина.

Настоящая задача является примером задачи контроля продукции пищевой индустрии. Подобные задачи могут найти широкие применения также при распознавании фальсифицированной продукции. Задача распознавания сортов вин по химическому анализу относится к хорошо поставленным задачам с легко отделимыми классами. Реальную ценность могут иметь результаты, в которых процент правильно распознанных объектов приближается к 100%. В данном примере исследуется выборка, объектами которой являются результаты химического анализа, выраженные в 13 признаках, таких как содержание алкоголя, яблочной кислоты, магния, цветовой оттенок и другие. Выборка разбита на 3 класса, соответствующих 3-м сортам вина, изготовленным из винограда, выросшего в одном и том же регионе Италии. Точность распознавания различными алгоритмами в режиме скользящего контроля составила 87.6-97.2%% правильных ответов, причем наилучшую точность показал метод «линейная машина».

Примечание. Автор постановки задачи и данных -
Forina, M. et al, PARVUS - An Extendible Package for Data Exploration, Classification and Correlation. Institute of Pharmaceutical and Food Analysis and Technologies, Via Brigata Salerno, 16147 Genoa, Italy.

4.2. Приложения в области медицины и здравохранения

4.2.1. Кластеризация возрастных распределений населения

Приведем результаты обработки некоторых демографических данных по материалам Всемирной организации здравоохранения.

Исходную информацию составили возрастные распределения населения 46 стран различных континентов (кроме Африки). Каждое распределение кодировалось строкой значений 10 числовых признаков. Признаки определяли долю населения стран из определенных возрастных групп. Основная задача состояла в автоматической классификации стран по их возрастным распределениям.

Для анализа данных использовался пакет TAXON (алгоритм Форель и методы иерархической группировки при различных метриках с последующим построением коллективного решения) /79/. Было построено коллективное решение задачи кластерного анализа, состоящее из шести кластеров. В первый, наиболее представительный кластер

(21 объект) попало большинство европейских стран, США и Австралия, имеющие достаточно ровные графики возрастных распределений на интервале от 5—14 до 55—64 лет. Во второй кластер зачислены некоторые островные государства (Ирландия, Куба и т. д.) и Израиль. Третий кластер составили в целом крупные развивающиеся страны (Таиланд, Филиппины, Венесуэла и т. д.). Для их графических представлений характерен резкий пик в юношеском возрасте. Сингапур и Гонконг образовали четвертый таксон. В отдельные таксоны выделились Япония и Норвегия. Для последней характерен весьма существенный процент населения в пожилом возрасте.

Визуальное сопоставление графиков возрастных распределений, соответствующих одному таксону, показало отчетливое качественное их совпадение. Результаты хорошо согласуются с социально-экономическими и культурными уровнями различных стран, а также особенностями их местонахождения и развития /18, 79/.

4.2.2. Прогноз результатов лечения остеогенной саркомы

Остеогенная саркома является тяжелым онкологическим заболеванием костей, часто возникающее в очень молодом возрасте. Современные методы лечения остеогенной саркомы включают в себя курс медикаментозного химиотерапевтического лечения с использованием препаратов, токсичных для опухолевых тканей. После курса химиотерапевтического лечения обычно проводится хирургическая операция по удалению из организма больного остатков опухоли.

Прогноз степени деструкции опухоли. Для успешного проведения хирургической операции при ее планировании необходима информация о степени разрушения опухоли в результате химиотерапии. Однако точно оценить степень разрушения можно только с помощью гистологического анализа уже удаленных тканей. В связи с этим в Онкологическом центре РАМН была поставлена задача разработать метод оценки степени разрушения опухоли по совокупности доступных на предоперационном этапе косвенных показателей (клинических, рентгенологических, лабораторных и др.). При разработке метода использовалась информация, содержащаяся в историях болезней 244 пациентов. Была использована двухуровневая шкала оценок степени деструкции (высокая и низкая степени).

Использование методов распознавания позволило создать прогностический алгоритм, общая точность которого составила 79% правильных прогнозов. При этом точными оказались 85% прогнозов, предсказывающих низкую степень деструкции, и 68% прогнозов, предсказывающих высокую степень деструкции.

Прогноз выживаемости по совокупности иммунологических параметров. Важнейшим фактором, определяющим защитные силы организма и влияющим на развитие заболевания, является состояние иммунной системы больного. Исследование взаимосвязи между иммунным статусом и исходом заболевания может сыграть важную роль для выработки эффективных методов лечения. В связи с этим была поставлена задача исследования возможности использования совокупности иммунологических показателей для прогноза исхода остеогенной саркомы. Исследования проводились по информации, содержащейся в историях болезней 80 пациентов, проходивших курс лечения в Онкологическом центре РАМН и не имевших метастаз к окончанию курса. Были сформированы две группы пациентов. Первая из групп включала 55 больных, у которых метастазы появились в течение одного года после окончания курса лечения. Вторая группа соответственно включала 25 больных с благоприятным исходом.

Использование методов распознавания позволило создать прогностический алгоритм с общей точностью прогноза исхода 76%. Точность прогноза оценивалась с использованием метода скользящего контроля /78/.

4.2.3. Прогноз динамики депрессивных синдромов.

Острые периоды сотрясения головного мозга, являющегося следствием черепно-мозговых травм, нередко сопровождаются депрессивными состояниями больных. Правильная оценка глубины депрессивного расстройства и тесно связанной с ней динамики заболевания в его начальном периоде существенна для выбора оптимального курс лечения. Вместе с тем, по направлению динамики могут быть достаточно четко выделены группа больных с явной положительной динамикой и группа больных, у которых не наблюдается сколь либо выраженных улучшений. Другой важной характеристикой диагностики заболевания является необходимость коллективного учета разнообразных факторов, описывающих конституциональные биологические и психологические особенности пациентов. Перечисленные обстоятельства указали на целесообразность использования для решения задачи прогноза динамики депрессивных синдромов методов распознавания.

Для обучения была использована информация, содержащаяся в историях болезней 50 пациентов 1586 окружного военного клинического госпиталя. Использование методов распознавания позволило создать прогностический алгоритм, общая точность которого составила 86% правильных прогнозов /14/.

4.2.4. Диагностика инсульта

Инсульт является одним из тяжелейших заболеваний с очень высокой летальностью. Известны два наиболее распространенных типа этого заболевания (ишемический и геморрагический), которые значительно отличаются по течению и по методам лечения. Задача диагностики типа инсульта является достаточно сложной даже для опытного врача. Вместе с тем учет типа инсульта является принципиально важным для организации правильного и эффективного лечения. В связи с этим была поставлена задача создания компьютерного метода диагностики типа инсульта по широкой совокупности клинических параметров, доступных для быстрого измерения. Для создания такого алгоритма было предложено использовать средства распознавания образов. Построение алгоритма проводилось по информации, содержащейся в историях болезней 301 пациента. В результате был построен диагностический алгоритм, имеющий точность 88% при распознавании ишемического инсульта и 71% при распознавании геморрагического инсульта, что примерно соответствует точности диагностики, достигнутой в лучших клиниках неврологического профиля.

4.2.5. Задача прогноза результатов лечения рака мочевого пузыря

Одним из наиболее успешных примеров использования иммунологических методов для лечения онкологических заболеваний является лечение рака мочевого пузыря путем введения в организм больного противотуберкулезной вакцины БЦЖ. Использование такого лечения приводит к исчезновению или значительному уменьшению опухоли примерно в 65% случаев. Важным является выявление на ранних этапах лечения тех пациентов, для которых данный метод не является эффективным, с целью проведения для них стандартного курса, основанного на химиотерапии и оперативном вмешательстве. Была поставлена задача прогноза результатов лечения по совокупности биохимических параметров. В результате был создан алгоритм, имеющий точность 85%.

4.2.6. Прогноз диабета

Рассматривалась задача предсказания наличия у пациента диабета по косвенным признакам. Замеры были взяты у 768 женщин племени Пима (Аризона, США), 268 из которых оказались больны диабетом. Для распознавания диабета использовались следующие восемь признаков: количество беременностей, концентрация глюкозы в плазме, диастолическое кровяное давление, толщина складки кожи в районе трицепса, 2-х часовой тест на содержание инсулина в крови, индекс массы тела, наследственная функция диабета, возраст. Точность построенного алгоритма распознавания составляла 77% правильных ответов. Логические закономерности достигали качества 20.5% (процент

пациентов, на которых они выполнялись). Подобным примером является следующая конъюнкция: $(x_1 \le 9{,}5)$ & $(154{,}5 \le x_2 \le 197{,}5)$ & $(69 \le x_3 \le 105)$ & $(x_4 \le 42{,}5)$ & $(x_5 \le 479)$ $(23{,}35 \le x_6 \le 51{,}15)$ & $(0{,}1365 \le x_7 \le 1{,}263)$ & $(21{,}5 \le x_8 \le 57{,}5)$.

Примечание. Автор постановки задачи и данных -

Vincent Sigillito (vgs@aplcen.apl.jhu.edu), Research Center, RMI Group Leader, Applied Physics Laboratory, The Johns Hopkins University, Johns Hopkins Road, Laurel, MD 20707.

4.2.7. Диагностика рака груди и распознавание меланомы

Задача диагностики рака груди рассматривалась по данным /72/. Обучающая выборка состояла из 344 эталонов, в том числе, 218 из класса « benign » и 126 из класса “malignant”. Для описания объектов использовались 9 признаков, принимающих целочисленные значения от 1 до 10:

1. **Clump Thickness 1 – 10;**
2. **Uniformity of Cell Size 1 – 10;**
3. **Uniformity of Cell Shape 1 – 10;**
4. **Marginal Adhesion 1 – 10;**
5. **Single Epithelial Cell Size 1 – 10;**
6. **Bare Nuclei 1 – 10;**
7. **Bland Chromatin 1 – 10;**
8. **Normal Nucleoli 1 – 10;**
9. **Mitoses 1 - 10.**

Контрольная выборка включала 355 объектов. Результаты, приведенные в таблице 8, показывают как высокую точность методов коллективного распознавания, так и являются иллюстрацией решения «простой задачи» - когда классы хорошо разделимы, тогда точность всех алгоритмов является высокой и примерно равной.

Методы распознавания	***Процент правильных ответов на контрольной выборке***
Алгоритмы вычисления оценок	95.5
Линейный дискриминант Фишера	94.4
Линейная машина	94.9
Логические закономерности	94.9
Многослойный персептрон	93.2
k – ***ближайших соседей***	95.2
Метод опорных векторов	95.2
Статистически взвешенные синдромы	94.4
Тестовый алгоритм	93.0

КОЛЛЕКТИВНЫЕ РЕШЕНИЯ	
Метод Байеса	94.6
Области компетенции	95.8
Голосование по большинству	95.8
Выпуклый стабилизатор	95.5
Шаблоны принятия решений	95.2
Динамический метод Вудса	94.6

Таблица 1. *Результаты контрольных испытаний различных алгоритмов на задаче диагностики рака груди*

Результаты, приведенные в таблице 1 показывают как высокую точность методов коллективного распознавания, так и являются иллюстрацией решения «простой задачи» - когда классы хорошо разделимы, тогда точность всех алгоритмов является высокой и примерно равной.

В таблице 2 приведены веса признаков, посчитанные с использованием метода «логические закономерности». Различия в их значениях незначительны, что свидетельствует об информативности всех признаков.

Анализ признаков	
Вес признака №1	0.825
Вес признака №2	0.907
Вес признака №3	0.812
Вес признака №4	0.799
Вес признака №5	0.719
Вес признака №6	0.600
Вес признака №7	0.800
Вес признака №8	0.916
Вес признака №9	0.862

Таблица 2. Веса признаков

Наилучшей логической закономерностью второго класса является предикат P(**x**)=(2<=X3<=9)&(5<=X7), выполняющийся на 67% эталонов второго класса.

В трудах 9-й Скандинавской конференции (Швеция, г. Уппсала, 6-9 июня 1995 г.) были приведены предварительные результаты решения задачи распознавания меланомы по 32 признакам, первые 12 из которых описывают геометрическую форму новообразования кожи, последние 21 признак - ее радиологические характеристики /65/. Исходную информацию составила выборка из числовых строк, каждая из которых является 32-признаковым описанием либо злокачественной опухоли (класс 1) - malignant lesions, либо неопасного новообразования (класс 3) - benign lesions, либо “переходного, промежуточного состояния новообразования” (класс 2) - dysplastic pigmented skin lesions. Задача распознавания мелономы состояла в автоматическом отнесении некоторой строки

из 32 чисел, являющейся описанием новообразования кожи некоторого пациента, к одному из трех вышеуказанных классов. Исходная информация была разбита на две таблицы, включающей представителей всех классов: таблицу обучения (17 объектов первого класса, 20 второго и 20 третьего) и таблицу контроля (12, 10 и 10 объектов соответствующих классов). Таблицы обучения и контроля в точности соответствуют экспериментам в /65/, там же приведены описания признаков.

Точность распознавания с помощью комбинаторно-логических методов на контрольной выборке составила 71.9%, что соответствует максимально достигнутой при сравнении с другими подходами. Следует отметить, что обучающая выборка объектов существенно непредставительна. В данной ситуации точность статистических методов существенно ниже (что подтверждается и публикацией /65/). При исключении из задачи промежуточного второго класса, т.е. ее сведения к ответу на вопрос «есть или нет злокачественное новообразование», точность диагностики превышает 90%.

4.2.8. Распознавание сужения сердечных сосудов.

Рассматривалась задача определения наличия сердечных заболеваний у людей из группы риска, с жалобами на боли в груди. Выборка содержит данные обследования 270 пациентов, у 120 из которых обнаружено более чем 50% сокращение диаметра крупных сосудов. Результаты обследования пациента выражаются в виде 13 признаков, таких как возраст, количество холестерина в сыворотке крови, кровяное давление, пульс, локация болей и другие. Точность распознавания в данной задаче с двумя классами составила более 83% в режиме скользящего контроля. Особенность информации – разнотипные признаки.

Примечание. Авторы постановки задачи и данных -

1. Hungarian Institute of Cardiology. Budapest: Andras Janosi, M.D.
2. University Hospital, Zurich, Switzerland: William Steinbrunn, M.D.
3. University Hospital, Basel, Switzerland: Matthias Pfisterer, M.D.
4. V.A. Medical Center, Long Beach and Cleveland Clinic Foundation:
   Robert Detrano, M.D., Ph.D.

4.3. Приложения в области техники

4.3.1. Диагностика состояний нефтепромыслового оборудования в отношении солеобразования

В процессе эксплуатации нефтепромыслового оборудования возникают негативные процессы, существенно осложняющие нефтедобычу. Так, обводнение нефтяных скважин

в условиях интенсивной разработки месторождений во многих случаях приводит к образованию неорганических солей (обычно карбоната кальция). Последующее их отложение в нефтепромысловом оборудовании (НПО) существенно усложняет эксплуатацию скважин, приводит к поломке погружных электроцентробежных насосов, значительно сокращает межремонтный период работы скважин.

Эффективность борьбы с отложением солей в НПО находится в прямой зависимости от времени обнаружения процесса солеобразования, непосредственное наблюдение которого практически недоступно в процессе эксплуатации скважин. Таким образом, внедрение эффективной методики диагностики состояния скважин в отношении солеобразования является необходимым условием успешной борьбы с солеотложением в НПО.

Традиционные методы диагностики основаны на экспертной оценке прироста давления на буфере скважины, расчете моделей карбонатного равновесия по данным химического анализа попутно добываемых вод или гидродинамических расчетах добываемой смеси нефть-вода в окрестности забоя скважины. Точность их не высока. Для решения данной задачи диагностики использовались модели распознавания, основанные на вычислении оценок.

Для описания явления солеобразования использовались системы признаков, связанных, по мнению экспертов, с данным явлением и которые могут быть измерены в промышленных условиях: давление, температура, химический состав попутно добываемых вод, условия эксплуатации. Объектами распознавания являлись описания скважин с помощью 15—20 признаков, замеренных в один и тот же момент времени для фиксированной скважины. Таблицы обучения формировались из выборок описаний скважин, на которых было зафиксировано отложение солей (и, следовательно, наличие солеобразования) и выборок описаний скважин, на которых солеотложений зафиксировано не было. Данные две группы описаний использовались для описания класса солеобразующих и класса соленеобразующих скважин.

Численные расчеты проводились по данным месторождений п/о Юганскнефтегаз и п/о Нижневартовскнефтегаз с использованием различных систем признаков. Точность распознавания находилась в пределах 85—95 % правильных ответов. Для решения задачи обычно было достаточно около половины исходных признаков. В состав информативных подмножеств признаков входили как признаки, связь которых с явлением солеобразования является хорошо известной (прирост давления, содержание ионов Ca^{+2}, HCO_3^-), так и признаки, целесообразность учета которых при диагностике состояний скважин в отношении солеобразования была ранее неизвестна /38-40/.

4.3.2. Контроль состояния технических устройств.

Рассматривалась задача диагностики, связанная с контролем управления космическим кораблем «Шаттл». Расположенная на корабле система радиаторов может находиться в одном из семи состояний (классов), в зависимости от показаний трех сенсорных датчиков, которые формируют значения 9-ти признаков. Особенность данной задачи состояла в том, что из 43500 измерений, которые использовались в качестве обучающей выборки, 99.6% относятся лишь к трем состояниям системы радиаторов, на остальные же 4 состояния приходилось 0.43% измерений. Точность распознавания на контрольных 14500 измерениях составила 99.7-99.8%.

Примечание. Автор постановки задачи и данных -
Jason Catlett, Basser Department of Computer Science, University of Sydney, N.S.W., Australia for providing the shuttle dataset.

4.4. Приложения в области сельского и лесного хозяйства

4.4.1. Прогноз урожайности сельскохозяйственных культур

Рассматривалась задача прогноза урожайности (ц/га) озимой пшеницы по описанию ее состояния на различных стадиях роста и основных климатических условий. Исходную информацию составили две числовые таблицы: $T^{1}_{9,53,7}$ и $T^{2}_{8,53,7}$. Строки таблиц являются описаниями состояния пшеницы на стадии колошения (для $T^{1}_{9,53,7}$) и стадии молочной спелости (для $T^{2}_{8,53,7}$), а также соответствующих климатических условий по четырем районам Ставропольского края в различные годы. Таким образом решалась задача прогноза с временем упреждения соответственно в один месяц и две недели. Разбиение исходной информации на классы проводилось по величине урожайности, которая колебалась от 7 до 36 ц/га. Класс K_i определялся как множество описаний состояния пшеницы с урожайностью в пределах полуинтервала $(\alpha_{i+1}, \alpha_i]$ (i = 1, 2, . . ., 7), где α_i (i = 1, 2, . . ., 8) принимали значения: 36; 29; 23; 18,5; 15; 12; 9; 7.

Задачи прогноза для каждой из фаз решались независимо. В признаковое пространство как для фазы колошения, так и для фазы молочной спелости вошли признаки: число колосоносных стеблей, высота растений, влагообеспеченность растений, средние температуры воздуха за определенные промежутки времени, средние дефициты влажности воздуха за те же отрезки времени. Кроме того, на стадии колошения использовались дополнительно признаки: продолжительность периода «выход в трубку —

колошение», продолжительность периода «возобновление вегетации—колошение». На стадии молочной спелости дополнительно использованы признаки: число колосков в колосе, продолжительность периода «колошение—молочная спелость».

Средняя ошибка прогноза составила 5 % от урожайности для фазы колошения и 4,2 % — для фазы молочной спелости. Максимальные ошибки прогноза были равны соответственно 8,9 и 8,2 % /53/.

В работах /4, 12/ описана модель оперативной обработки данных дистанционного зондирования в целях прогнозирования урожая в сельском хозяйстве.

4.4.2. Обработка материалов многозональной съемки с целью определения преобладающих пород

Исследуемую информацию составили «спектральные портреты» лесных массивов Рязанской области, представленных насаждениями сосны, березы, осины и ели. Площадь каждого массива была не менее 4 га. Многозональная аэросъемка выполнялась блоком из четырех камер в различных спектральных зонах. По негативам измерялись плотности в каждой зоне, на основе которых формировались описания лесных массивов в виде строк из семи признаков-плотностей по отдельным зонам, некоторых их отношений. Эталонная выборка состояла из 94 описаний. Основная задача состояла в построении алгоритма автоматического распознавания пород по спектральным данным обследуемого лесного массива.

Построенный в классе алгоритмов вычисления оценок оптимальный по функционалу качества алгоритм распознавания показал высокую точность — свыше 95 % правильных ответов. Оценка информативности признаков выявила неинформативность зоны 580—605 нм, что согласуется с результатами других подходов /5/.

4.5. Приложения в области физики, химии, биологии

4.5.1. Распознавание радиосигналов

Рассматривалась прикладная задача из области радиофизики /83/. Имеется система из 16-ти высокочастотных антенн, исследующая свойства ионосферы. Требуется отличать друг от друга 2 типа сигналов – «положительные», отраженные находящимися в ионосфере свободными электронами и несущие полезную информацию о структуре ионосферы и «отрицательные», прошедшие сквозь ионосферу без отражения. Электромагнитные сигналы характеризуются набором из 17-ти пульсаций, каждая из которых в свою очередь имеет 2 аттрибута. Таким образом, для описания сигналов использовалось 34 признака. Для численных расчетов использовались таблицы обучения и

контроля примерно равного объема, при этом каждый класс включал 150-200 объектов. Точность распознавания на контрольной выборке была в пределах 82.5 -98.7%%. Интересно, что наилучшими по точности оказались такие разнотипные алгоритмы, как «многослойный перцептрон» и «к-ближайших соседей». Точность распознавания на контроле методами построения коллективных решений составила 95.7-99.6%% , причем наилучший результат показал «выпуклый стабилизатор».

4.5.2. Прогноз свойств твердых сплавов стали

Синтез твердых сплавов стали с требуемыми свойствами требует проведения большого числа трудоемких экспериментов. При этом актуальной является задача прогнозирования результатов экспериментов до их реального проведения на основе имеющегося опыта подобных экспериментов, а также известных сплавов стали. Пусть дана выборка описаний экспериментов по синтезу твердых сплавов стали, содержащая как положительные результаты экспериментов, когда были получены сплавы с требуемыми свойствами, так и отрицательные результаты. Под описанием эксперимента понимается совокупность всех характеристик, от которых, как предполагается, зависит результат: физико-химические, структурные характеристики компонентов сплава, параметры технологии и т.д. Тогда данную выборку описаний экспериментов с их результатами можно использовать в качестве обучающей и прогнозировать результаты планируемых экспериментов.

Подобные задачи прогноза рассматривались на примере прогноза сплавов, удовлетворяющих ограничениям по пределу прочности на изгиб и твердость. В данном случае в качестве признаков использовались значения весового содержимого каждого компонента в сплаве. Точность прогноза сплавов с данными свойствами на контрольных данных составила свыше 93 процентов правильных ответов.

На базе физико-химических и патентных соображений была составлена выборка гипотетических сплавов, для которых осуществлялся прогноз. Для реального синтеза был выбран тот компонентный состав, прогноз для которого был наиболее перспективен. В итоге был создан новый сплав с высокими физико-механическими и эксплуатационными свойствами, который может быть использован взамен вольфрамосодержащих твердых сплавов типа T5KIO /13/.

4.5.3. Распознавание мест локализации протеина

Задача распознавания состояла в распознавании мест локализации протеина по 7 косвенным признакам /67/. Общее число мест локализации протеина (классов) составляло

8. Обучающая и контрольная выборки включали, соответственно, 158 и 179 элеменов. Особенностью задачи является неравномерность распределения объектов по классам. Так, распределение объектов в обучающей информации по классам было следующим: 76, 33, 24, 11, 6, 4, 2 и 2, соответственно. Точность распознавания контрольных объектов различными алгоритмами составила 75-83% правильных ответов.

Наилучшие логические закономерности первых (наиболее представительных) классов представлены в таблице 3. В столбцах таблицы приведены левые и правые границы интервалов значений признака по каждой из закономерности и процент эталонных объектов, на которых данные закономерности выполнены.

Номер класса	Признак x1	Признак x2	Признак x5	Признак x6	Признак x7
1	0.06≤x1≤0.65	x2≤0.56	x5≤0.72	x6≤0.52	x7≤0.60
2	x1≤0.61	0.33≤x2≤0.68	0.30≤x5≤0.74	0.57≤x6	0.16≤x7
3	0.58≤x1≤0.78	0.48≤x2≤0.87	0.16≤x5≤0.63	0.38≤x6≤0.57	0.18≤x7≤0.55
4	0.63≤x1≤0.84	0.25≤x2≤0.53	0.49≤x5≤0.69	0.58≤x6≤0.87	0.60≤x7≤0.88

Таблица 3. Наилучшие логические закономерности первых четырех классов

Интерпретация первых четырех классов была следующая:

- «cytoplasm»;
- «inner membrane without signal sequence»;
- «perisplasm»;
- «inner membrane, uncleavable signal sequence».

4.5.4. Прогноз свойств новых неорганических соединений

Разработка теоретических методов поиска новых неорганических веществ с заданными свойствами является одной из важнейших проблем химии и материаловедения. Один из наиболее перспективных путей решения проблемы связан с использованием нового подхода, возникшего на стыке химии и современной информатики - с компьютерным конструированием неорганических веществ и материалов /68/. Основная гипотеза, лежащая в основе этого метода: фундаментальные свойства многокомпонентных неорганических веществ при различных условиях (температуре, давлении, соотношении компонентов и т.д.) связаны периодическими зависимостями с фундаментальными свойствами химических элементов, входящих в их состав. Существование таких зависимостей является следствием Периодического закона Д.И.Менделеева. Предполагается, что многочисленные, известные к настоящему времени, неорганические вещества подчиняются этим зависимостям.

Задача поиска зависимостей в информации БД по свойствам неорганических веществ формулируется следующим образом. Пусть і-ый химический элемент определен набором М свойств (x_{i1}, x_{i2},…, x_{iM}). Тогда К-компонентное химическое соединение описывается точкой в М*К-мерном пространстве свойств компонентов. Из-за

периодичности свойств химических элементов, точки, соответствующие комбинациям близких по химической природе элементов, должны образовывать компактные классы в этом многомерном пространстве. Пусть существует некоторый набор химических соединений (в общем случае следует говорить о физико-химических системах, образованных различными элементами), для которых известна принадлежность к разным классам (*обучающая выборка*). При этом каждая физико-химическая система задается набором значений свойств образующих ее элементов и/или более простых соединений (простых галогенидов, оксидов, халькогенидов и т.д.). Необходимо построить в M*K-мерном пространстве гиперповерхности (геометрический аналог искомой закономерности), разделяющие физико-химические системы одного класса от систем других классов. Предполагается, что, вследствие периодичности свойств, полученные разделяющие поверхности можно использовать для определения статуса еще неисследованных физико-химических систем. Этот процесс *прогнозирования* требует знания только свойств химических элементов или более простых соединений, образующих неизученную физико-химическую систему. Таким образом, задача поиска веществ, подобных уже исследованным, сведена к классической задаче обучения ЭВМ классификации объектов.

Несмотря на множество алгоритмов обучения ЭВМ, поиск метода, наиболее подходящего для решения химических задач, как правило, осуществляется путем «проб и ошибок». Для того, чтобы оценить возможности различных алгоритмов при решении задач компьютерного конструирования неорганических соединений, European Office of Aerospace Research and Developments (EOARD) предложил набор тестовых задач. Система РАСПОЗНАВАНИЕ была протестирована на одной из этих задач: задаче прогнозирования двойных систем (физико-химических систем, образованных двумя химическими элементами, например, Fe-C, Ni-Al и т.д.) с образованием соединений (любого состава) при нормальных условиях (298 K и 1 атм.) и без образования соединений (твердые растворы, эвтектические системы или гетерогенные смеси).

Обучающая выборка включала 923 положительных объекта (системы с образованием соединений) и 410 отрицательных объектов (системы без образования соединений). Экзаменационная выборка состояла из 473 положительных и 219 отрицательных объектов. Для описания физико-химических систем использовались 87*2 = 174 признака (87 свойств для каждого химического элемента: атомный номер; температуры плавления и кипения; энтальпии плавления, кипения, атомизации; модуль Юнга; электроотрицательности; первые три потенциала ионизации; размерные факторы; Менделеевский номер и т.д.). Точность расчетов на контрольной выборке превысила 95%

правильных ответов, что соответствует наилучшим результатам, полученным с использованием других систем распознавания.

4.5. Приложения в области обработки и распознавания изображений

4.5.1. Распознавание изображений автомобилей

Описанные методы могут быть широко применимы для распознавания графических изображений. В качестве примера приведем задачу распознавания изображений автомобилей, снятых камерой под различными углами зрения. Требуется различить изображения четырех типов – двухэтажного автобуса. микроавтобуса Шевроле. седана Сааб 9000 и купе Опель Манта 400. Каждое изображение было получено в разрешении 128x128 при одинаковом освещении и минимизации бликов, и характеризуется 18-ю признаками: компактность (квадрат периметра/площадь), округлость (квадрат среднего расстояния от центра/площадь), эксцесс и ассиметрия относительно горизонтальной и вертикальной оси, и т. п. Каждому типу автомобиля соответствует один из четырех классов, в каждом классе содержится около 200 объектов-изображений. Точность распознавания контрольных объектов (на выборке из 100 изображений) составила около 80% правильных ответов.

Примечание. Автор постановки задачи и данных -

Drs.Pete Mowforth and Barry Shepherd. Turing Institute George House. 36 North Hanover St. Glasgow. G1 2AD

4.5.2. Распознавание рукописных цифр

Задача распознавания рукописных символов является одной из важнейших областей применения методов распознавания. В данной сфере работают исследователи многих ведущих университетов и компаний, имеются коммерческие программы распознавания оптических символов и многочисленные публикации. В работах /7, 61/ описаны результаты распознавания рукописных цифр с помощью моделей частичной прецедентности. Исходный материал составляли 1000 рукописных цифр, представление каждой цифры задавалось бинарной матрицей 16x16.

Численные эксперименты были проведены при двух способах описаний цифр: «бинарном» и «символьном». Первый вид описаний непосредственно порождался исходным бинарным представлением. Каждое описание цифры, заданное бинарной матрицей 16x16, переводилось в более грубое (но более устойчивое по элементам фиксированного класса) описание 6x6. Значениями новых признаков являются суммы элементов квадратов исходной матрицы 3x3 или прямоугольников 1x3 и 3x1. Символьные

описания формировались на языке признаков, отражающих логическую и статистическую картину распределения единиц в матрице 16x16. По матрице 16x16 вычисляются признаки, характеризующие контуры цифр, гистограммы, производные контуров, число компонент связности по горизонтальным и вертикальным линиям, и др. (всего 478). Далее использовались лишь 36-47 наиболее информативных признаков.

Для обучения и контроля использовались выборки по 500 описаний. Точность распознавания с применением моделей частичной прецедентности составила около 94% при 6x6 признаковом описании и 97% при символьном. Учитывая непрезентативность выборки (исходная информация была создана с участием 10 персон), данные результаты следует считать весьма высокими.

Глава 5. Описание программного продукта

5.1. Описание графической оболочки

5.1.1. Главное окно

Основу рабочей среды программы составляет стандартный многодокументальный графический интерфейс Windows. После запуска программы на экран выводится следующее окно (см. рис. 30).

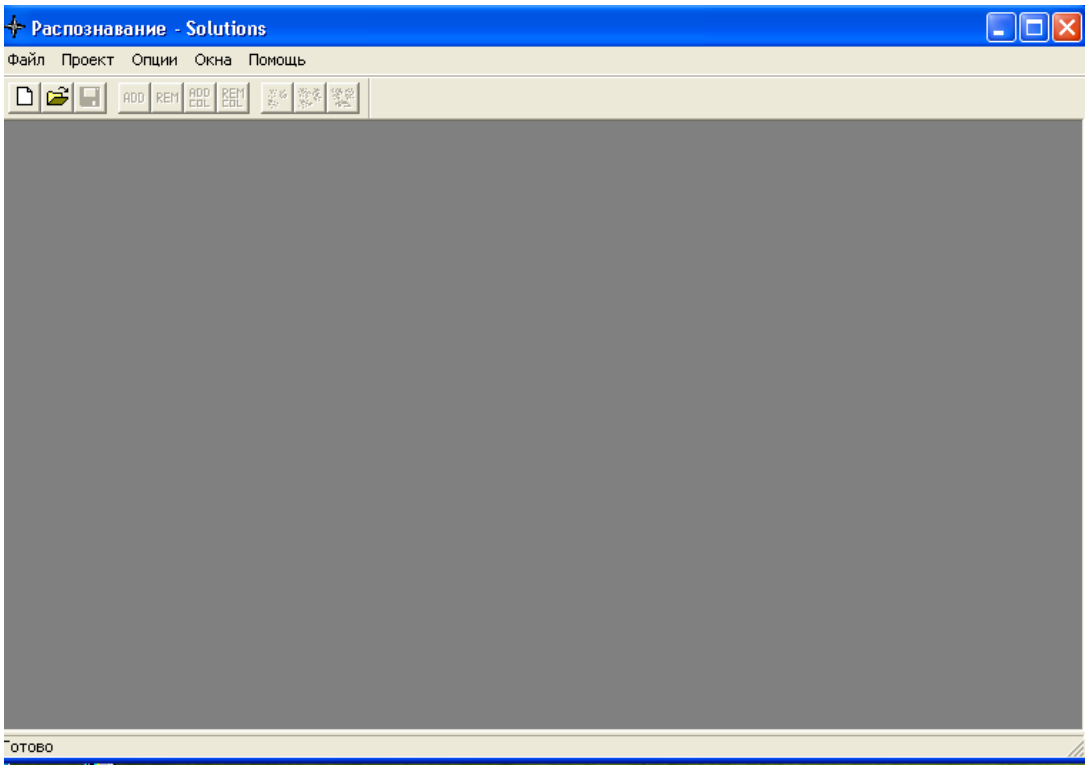

Рис. 30. Внешний вид программы

Данное окно имеет меню и панель инструментов, которые позволяют управлять программой. Все кнопки на панели инструментов дублируют некоторые пункты меню. Меню состоит из следующих пунктов: «Файл», «Проект», «Опции», «Окна», «Помощь».

На панели инструментов находятся слева направо три группы кнопок. Первая группа - кнопки управления проектами: «Создать новый проект», «Загрузить существующий проект», «Сохранить текущий проект». Вторая группа кнопок – кнопки управления методами внутри конкретного проекта: «Добавить в проект метод», «Удалить метод из проекта», «Добавить в проект метод построения коллективного решения», «Удалить из проекта метод построения коллективного решения». Третья группа кнопок – это кнопки управления загрузкой выборок с данными: «Загрузить выборку для обучения»,

«Загрузить выборку для обучения коллективных методов», «Загрузить выборку для распознавания».

В зависимости от режима работы некоторые кнопки могут быть недоступны, например, при решении задачи кластеризации не используется выборка для распознавания и, соответственно, не будет доступна кнопка загрузки распознаваемой выборки.

5.1.2. Главное меню

5.1.2.1. Пункт меню «Файл»

Посредством данного пункта меню происходит управление рабочими проектами системы РАСПОЗНАВАНИЕ. Состоит из пунктов: «Новый проект», «Загрузить проект», «Закрыть проект», «Сохранить проект», «Недавние проекты», «Выход».

Новый проект.

После выбора соответствующего пункта меню на экране появится диалог создания нового проекта (см. Рис. 31):

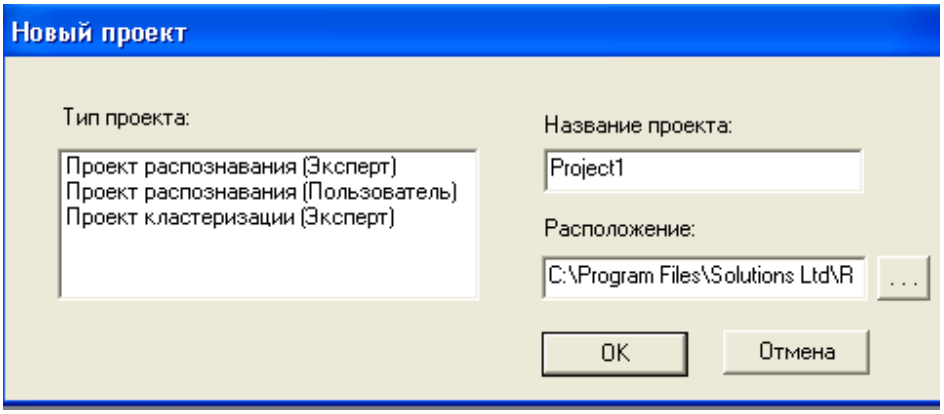

Рис. 31. Создание проекта

В левой части диалога надо выбрать один из возможных типов проектов, а в правой части - путь к директории, в которой будет храниться проект и его название. Итоговая директория, в которой будут храниться файлы проекта, будет иметь путь «Расположение»\«Название проекта». При нажатии на кнопку «...» на экране появится диалог с возможностью выбора директории проекта. Как уже говорилось выше, типы проектов подразделяются на «эксперт»/«пользователь». В первом случае пользователю будет предоставлен доступ ко всем параметрам методов и возможность самому контролировать качество обучения, во втором случае пользователю предоставляется некоторый набор готовых сценариев обучения, которые сами выбирают параметры методов и средства контроля качества обучения. В созданной директории появится файл «Название проекта.crp», который и является файлом проекта. При добавлении методов в

проект в данной директории будут появляться директории с рабочими файлами соответствующих методов.

Загрузить проект.

После выбора данного пункта меню на экран выводится стандартный диалог Windows, предлагающий загрузить файл. Расширение файлов проекта - .crp. После выбора соответствующего файла на экране появится окно ранее сохраненного проекта.

Закрыть проект.

Закрывает текущий проект. Если проект не был сохранен, то выводит на экран предупреждающее сообщение.

Сохранить проект.

Сохраняет данный проект в файл, чтобы его можно было использовать в дальнейшем. При этом все методы, которые были обучены, запоминают результаты обучения и в дальнейшем при загрузке проекта можно сразу приступать к распознаванию без предварительного обучения. Стоит заметить, что методы запоминают те значения параметров, при которых они были в последний раз обучены, а не те которые в данный момент выставлены в окнах настроек.

Недавние проекты.

Выводит на экран меню со списком последних проектов, после выбора соответствующего проекта загружает его.

Выход.

Осуществляет выход из программы, аккуратно завершая при этом все происходящие процессы. Выход может произойти не сразу, а по истечении некоторого времени, которое потребуется для завершения процессов, происходящих в программе.

5.1.2.2. Пункт меню «Проект».

При помощи команд из данного меню происходит управление составными частями проекта, такими как методы и выборки. Состоит из следующих пунктов: «Загрузить обучающую выборку», «Загрузить выборку для обучения коллективных методов», «Загрузить распознаваемую выборку», «Добавить в проект», «Удалить из проекта».

Загрузить обучающую выборку.

Позволяет загрузить выборку, которая будет использоваться для обучения методов. Сначала на экране появится диалог с возможностью выбора типа загрузчика (см. рис. 32):

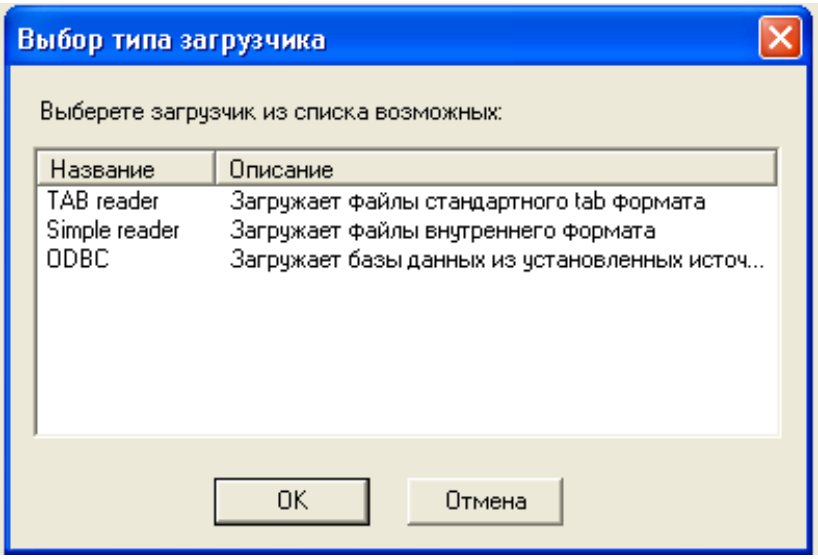

Рис. 32. Выбор загрузчика.

Tab reader загружает файлы формата программы LOREG /6/, Simple reader загружает текстовые файлы, содержащие выборки, значения которых разделены «,». ODBC загружает большинство стандартных типов файлов, в том числе Excel файлы.

В случае выбора «Tab reader» или «Simple reader» на экране появится стандартный диалог загрузки файлов, а в случае выбора ODBC - диалог, который позволяет указывать различные параметры загрузки из соответствующих файлов. Более подробно загрузчики и соответствующие форматы файлов будут описаны ниже.

Загрузить выборку для обучения коллективных методов.

Методы построения коллективных решений могут требовать отдельной выборки для обучения. В случае ее отсутствия можно просто загрузить выборку, которая использовалась для обучения методов коллектива. В остальном процесс загрузки происходит так же, как и в предыдущем пункте. Данный пункт меню доступен только в случае проектов для распознавания.

Загрузить распознаваемую выборку.

Загрузка выборки для распознавания. Процесс загрузки аналогичен описанному выше. Данный пункт меню, естественно, доступен только для распознавания.

Добавить в проект.

При выборе данного пункта на экране появится подменю, позволяющее добавить в проект либо обычный метод, либо метод построения коллективного решения. После выбора соответствующего типа метода на экране появится диалог со списком имеющихся в наличие методов соответствующего типа (см. рис. 33):

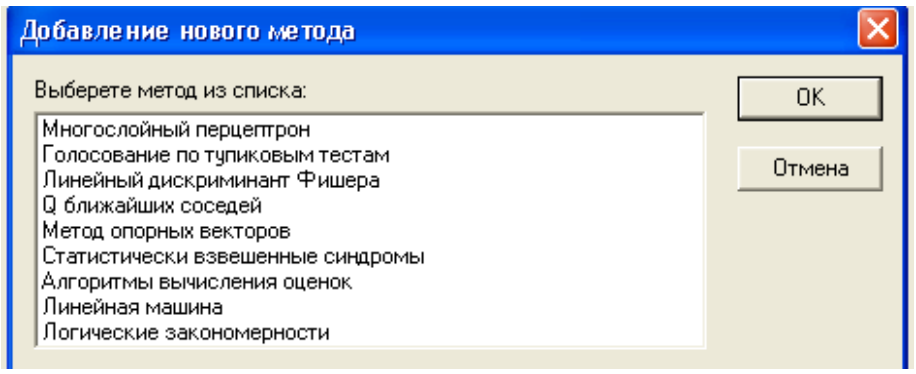

Рис. 33. Добавление метода.

Удалить из проекта.

При выборе данного пункта меню на экране появится подменю, описанное в предыдущем пункте. При выборе соответствующего типа метода на экран будет выведен диалог с содержащимися на данный момент в проекте методами (см. рис. 34):

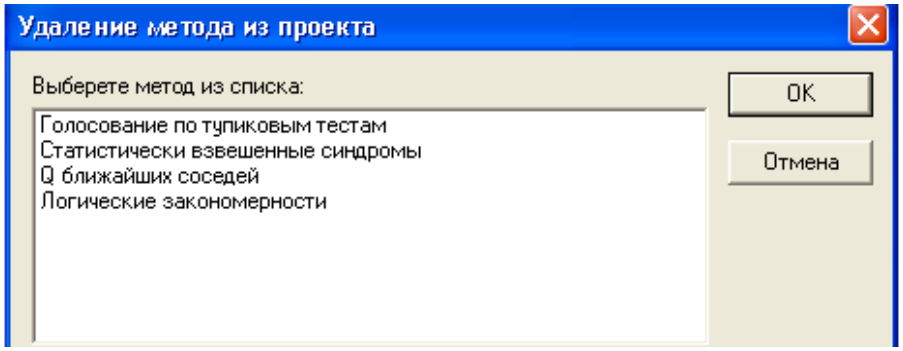

Рис. 34. Удаление метода.

5.1.2.3. Пункт меню «Опции».

Состоит из единственного пункта, который выводит на экран диалог с опциями программы (см. рис. 35):

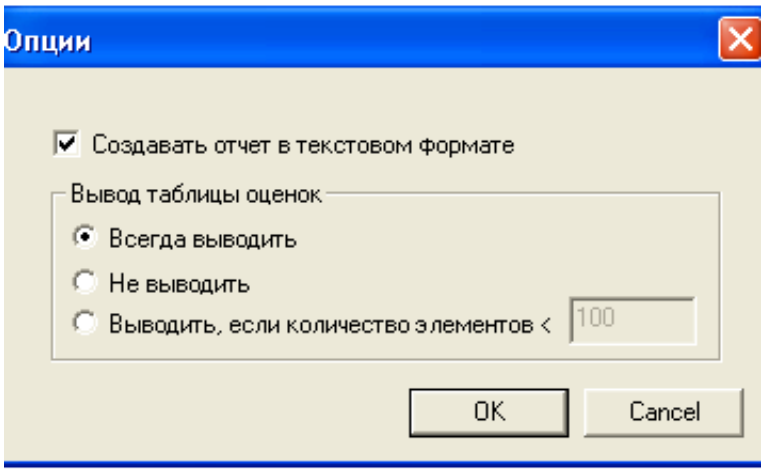

Рис. 35. Настройка отчета

Переключатель «Создавать отчет в текстовом формате» позволяет включить/отключить создание отчета в текстовом виде. При работе программы

результаты обучения и распознавания всегда представляются в виде HTML документа. Данная опция позволяет наравне с этим документом создавать такой же отчет еще и в виде текстового файла. Вывод большой таблицы оценок на экран в HTML формате может потребовать значительного времени, поэтому предоставляется возможность включения/отключения вывода таблицы оценок, или вывод таблицы оценок, если она меньше определенного размера.

5.1.2.4. Пункт меню «Окна».

Является стандартным пунктом меню многодокументального приложения Windows. Позволяет переключаться между окнами различных проектов и распределять их внутри главного окна программы различным образом.

5.1.2.5. Пункт меню «Помощь».

Состоит из двух пунктов: «Помощь», «О программе».

Помощь.

Выводит на экран окно, которое содержит в себе описание программы в виде гипертекста.

О программе.

Выводит на экран диалог с информацией о разработчиках данного программного продукта.

5.1.3. Окно проекта.

Основная обработка данных происходит в рамках проекта системы. Каждый проект содержит некоторый набор методов соответствующего типа из предоставляемых системой. В рамках проекта обрабатывается одна и та же выборка данных всеми включенными в проект методами.

5.1.3.1. Окно проекта (эксперт).

Окна проектов для распознавания и кластеризации в случае экспертного пользователя выглядят схожим образом. Отличие заключается в закладках в правой части экрана. Данные закладки управляют различными этапами обработки информации, которые у методов распознавания и кластеризации различны. Описание каждой закладки будет дано ниже.

Основное окно проекта выглядит следующим образом (см. рис. 36):

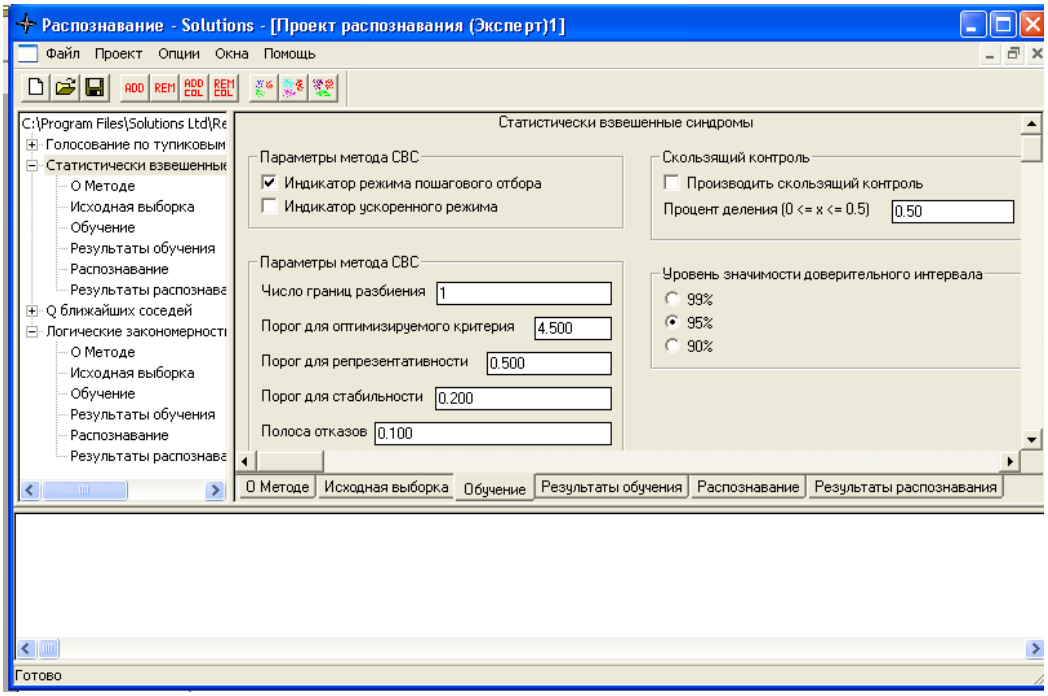

Рис. 36. Основное окно проекта

В левой части окна отображается дерево методов проекта. С его помощью можно переключаться между разными методами и между этапами работы одного метода. Дерево имеет следующую структуру. Корнем дерева является проект, который определяется своим названием. На следующем уровне находятся методы. В случае, если в проект добавлено несколько одинаковых методов (это может быть сделано с целью сравнить результаты, даваемые методом при различных параметрах), то к названию методов будут добавлены номера (1), (2),... Вершины, отходящие от каждого метода, соответствуют управляющим закладкам с таким же названием.

В правой части окна содержатся управляющие закладки, каждая из которых означает некоторый этап работы метода. Более подробно об этом написано ниже при описании управления методами.

В нижней части окна находится поле, в которое выводятся различные сообщения, возникающие при работе методов. При каждом старте процесса обучения или распознавания данное окно будет автоматически очищаться.

5.1.3.2. Окно проекта (пользователь)

В данной версии программы проект для обычного пользователя реализован только для задачи распознавания. В отличие от экспертного случая, в таком проекте пользователю не предоставляется возможность непосредственного управления методами. Вместо этого ему предлагаются несколько готовых сценариев обучения. В соответствии с выбранным сценарием будет производиться индивидуальная настройка каждого метода и

применяться различные средства контроля качества обучения. Окно проекта выглядит следующим образом (см. рис. 37):

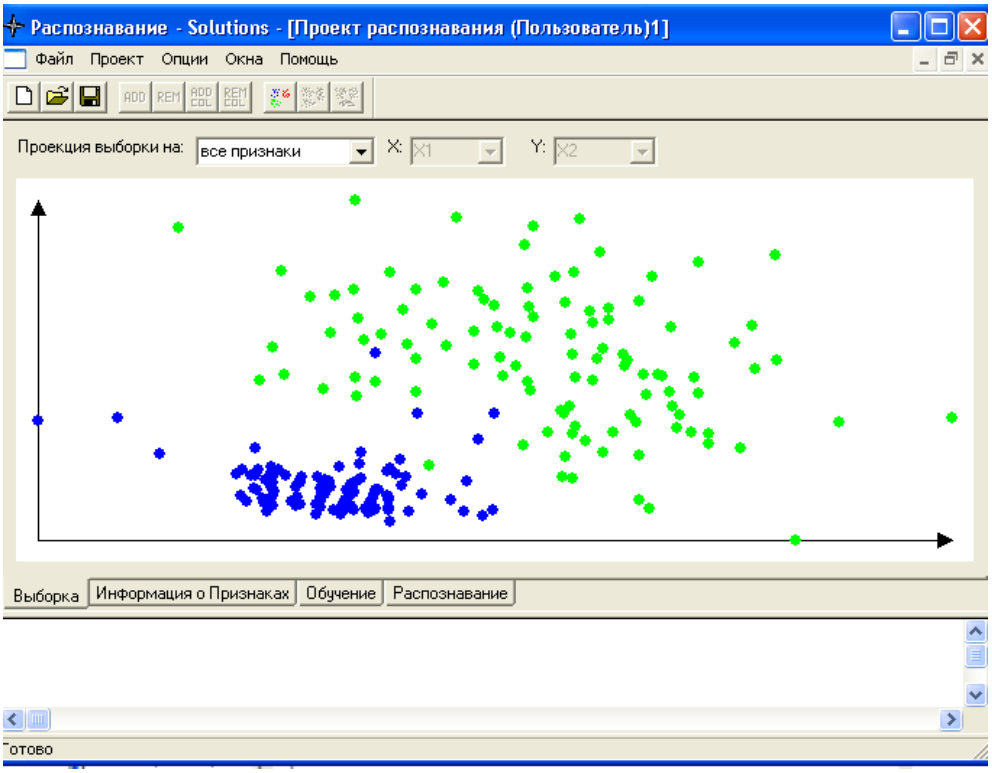

Рис. 37. Окно проекта (пользователь).

Окно состоит из двух частей. В нижнюю часть выводятся различные сообщения, которые возникают при работе методов, а в верхней находятся управляющие закладки. Для данного типа проекта можно загрузить только выборку для обучения. Распознавание в таком проекте производится для конкретного объекта, который задается вручную. Оценивание точности распознавания каждым алгоритмом осуществляется автоматически по обучающим данным. Опишем управляющие закладки данного проекта.

Закладка «Выборка».

На данной закладке отображается обучающая выборка. Система предоставляет различные возможности по отображению выборки, которые будут описаны ниже.

Закладка «Информация о признаках».

Закладка выглядит следующим образом (см. рис. 38):

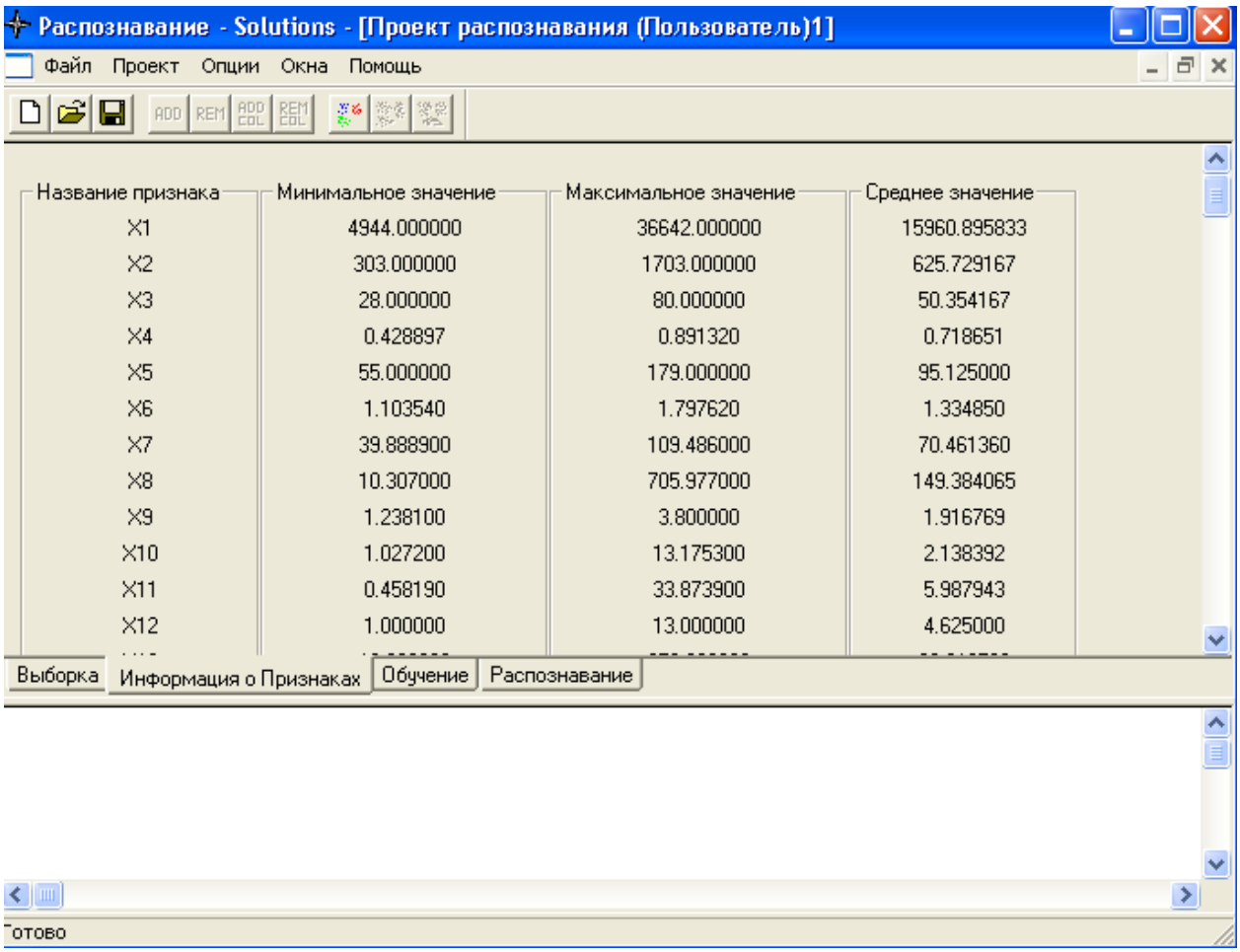

Рис. 38. Информация о признаках.

В текущей версии на закладке отображены для каждого признака его минимальное, максимальное и среднее значения.

Закладка «Обучение».

Закладка выглядит следующим образом (см. рис. 39):

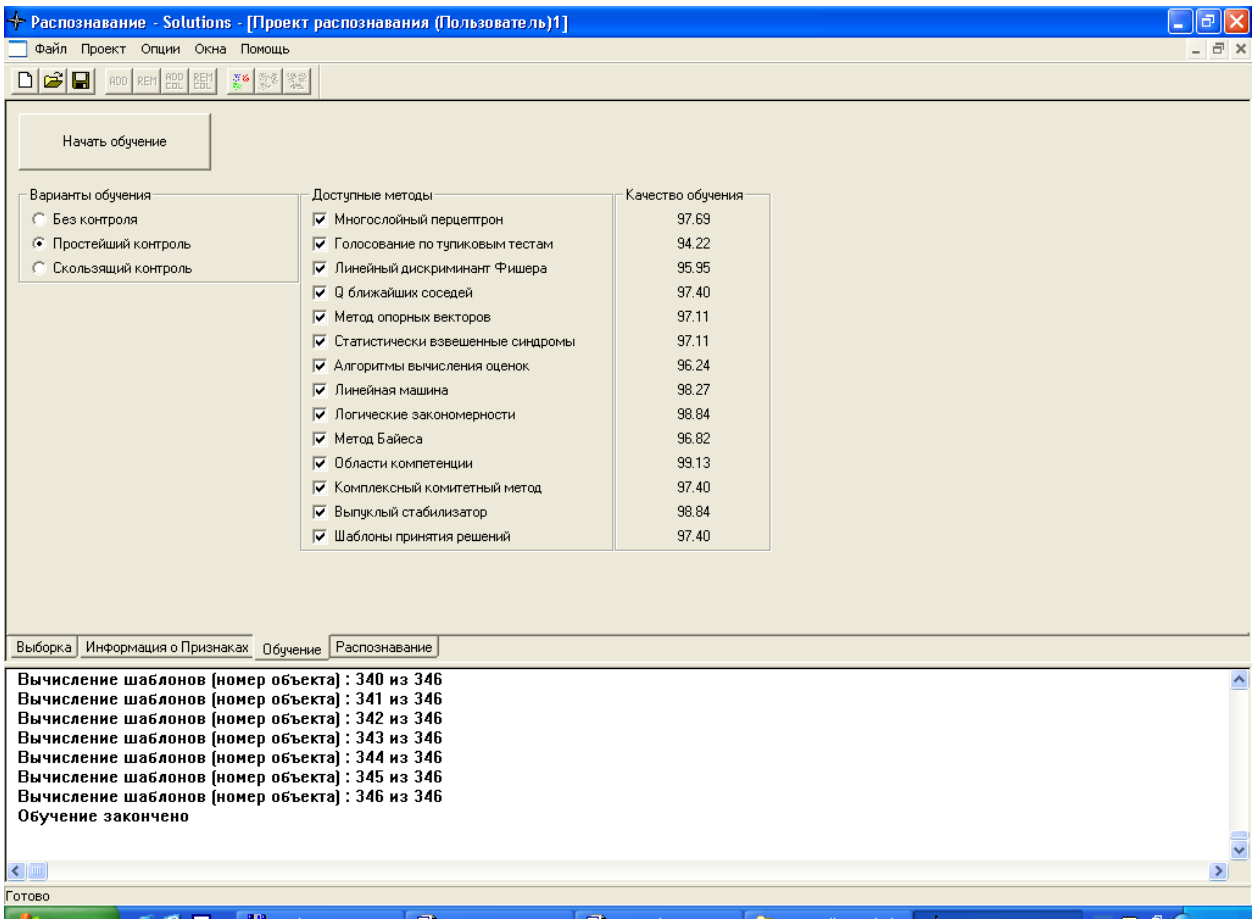

Рис. 39. Обучение

Левый столбец выводит список всех доступных сценариев обучения и средств контроля качества. Пользователь должен выбрать один сценарий из списка. В следующем столбце выведены все доступные для обучения методы, пользователь должен отметить те методы, которые он хочет использовать для обработки заданной выборки. После этого он нажимает на кнопку «Начать обучение». По мере обучения методов на экране будут отображаться получаемые оценки качества обучения. После того как все методы обучены, пользователь переходит к следующей закладке.

Закладка «Распознавание».

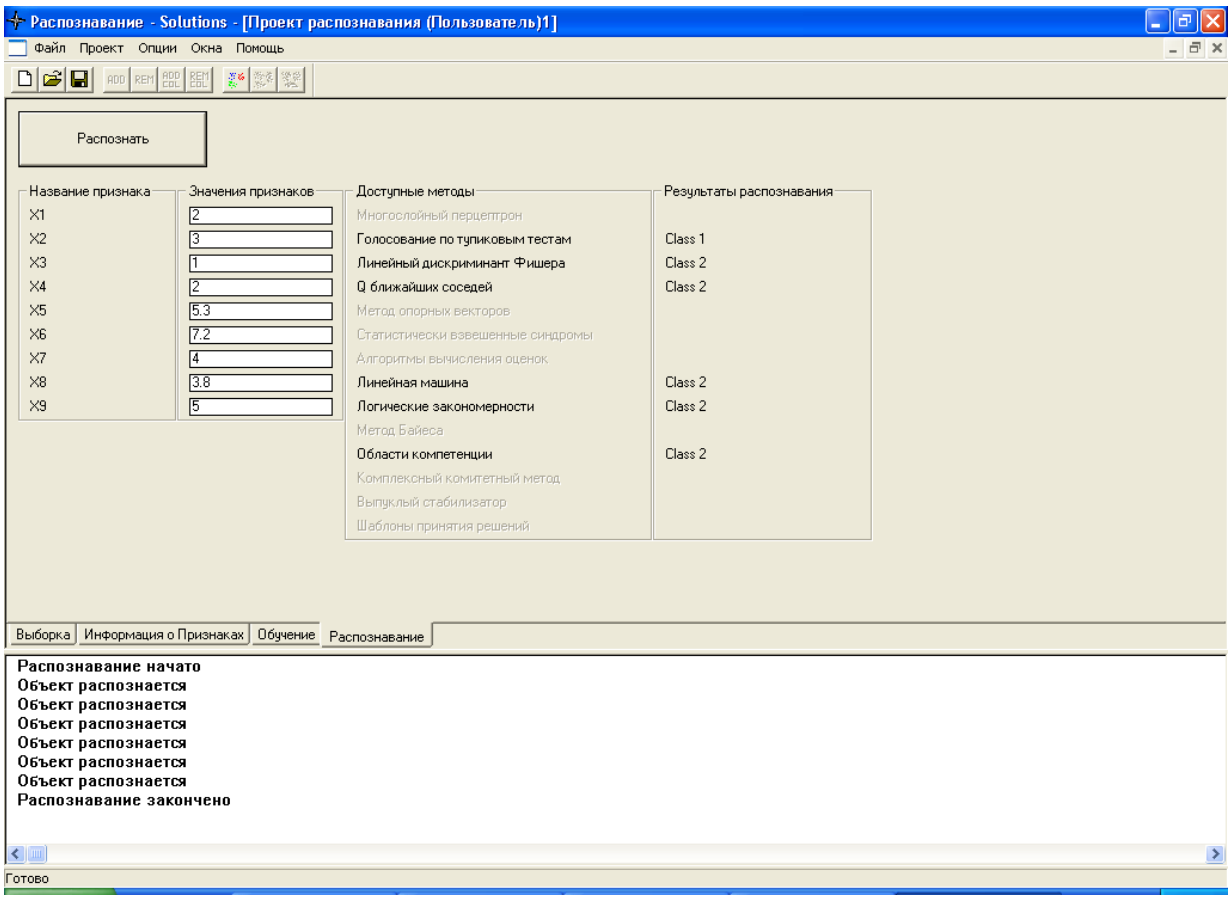

Рис. 40. Вид закладки **«Распознавание»**

На данном этапе работы пользователю предлагается задать значения признаков распознаваемого элемента. Те значения, которые не заданы, будут обрабатываться как неизвестные. После нажатия на кнопку «Распознать» произойдет распознавание заданного элемента всеми обученными ранее методами.

5.1.4. Метод распознавания (эксперт).

Опишем закладки, посредством которых управляются методы распознавания в режиме эксперт.

Закладка «О методе».

Закладка выглядит следующим образом (см. рис. 41):

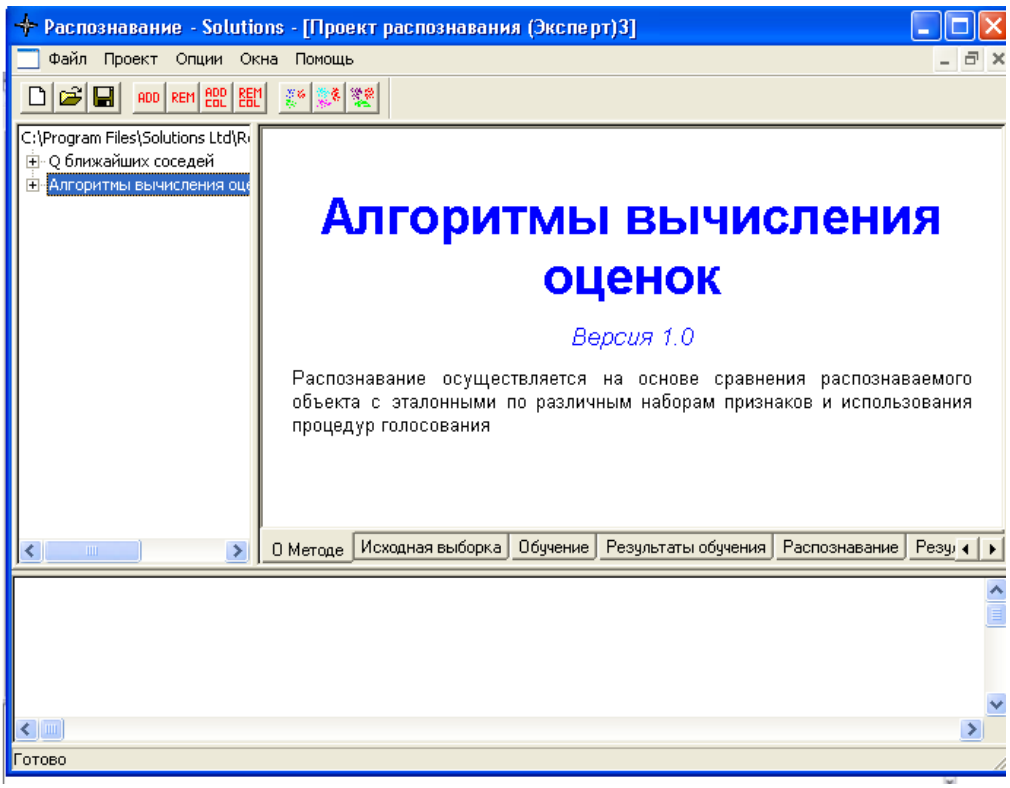

Рис. 41. Вид закладки **«О методе»**

На ней выводится краткая информация о методе: название, версия и описание метода «в одну строку».

Закладка «Исходная выборка».

На данной закладке отображается выборка, которая была загружена для обучения. Выглядит закладка следующим образом (см. рис. 42):

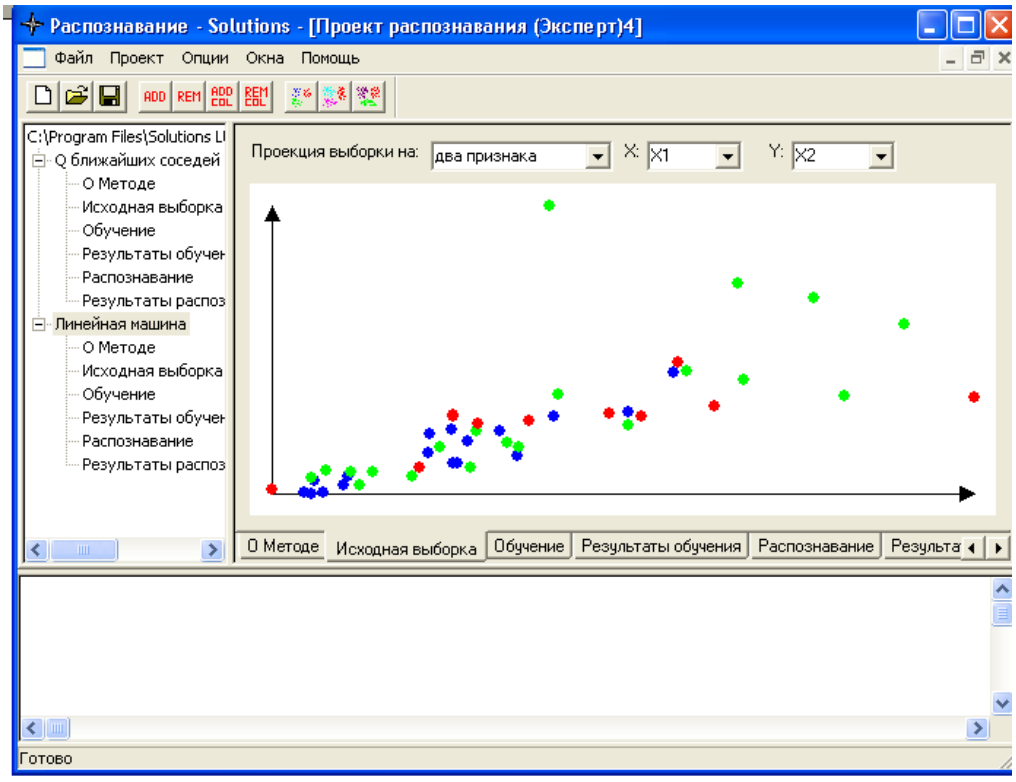

Рис. 42. Вид закладки **«Исходная выборка»**

В системе предусмотрены различные режимы отображения выборки, по умолчанию выборка проецируется на плоскость признаков №1, 2. Объекты одного цвета относятся к одному классу. Признаки, на плоскость которых отображается выборка, можно выбрать из соответствующих «выпадающих списков». В списке «Проекция выборки на» предусмотрено три режима проекций: первый – проекция на два заданных признака, второй – проекция на один выбранный признак (выводится гистограмма значений признака), третья – проекция на плоскость двух обобщенных признаков. При нажатии на правую кнопку мыши на экране появится выпадающее меню, состоящие из двух пунктов. При выборе первого из них на экране появиться следующий диалог (см. рис. 43):

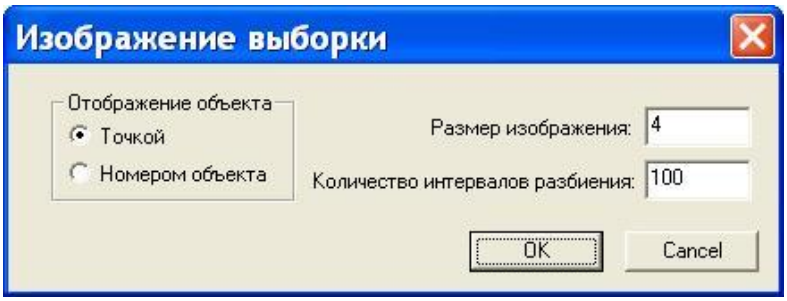

Рис. 43. Изображение выборки

В данном диалоге можно выбрать следующие параметры изображения выборки: отображение объектов точками или номерами, размер точки/номера, число интервалов разбиения значений признака. Последний параметр используется при построении проекции выборки на один признак.

При выборе второго пункта меню на экране появится следующий диалог (см. рис. 44):

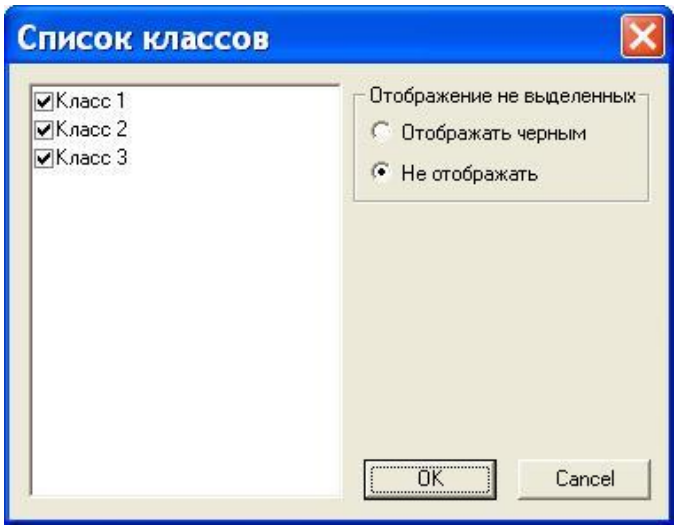

Рис. 44. Отображение классов

При помощи данного диалога можно отключить отображение некоторых классов или отображать ненужные классы черным цветом. Это может быть полезно, когда мы обрабатываем выборку с большим числом классов и классы на проекции могут быть

плохо различимы. В таком случае, можно выводить один или несколько выбранных классов отдельно друг от друга.

Закладка «Обучение».

На данной закладке изображены окна настроек конкретных методов, поэтому в зависимости от метода они будут выглядеть по-разному. На рис.45 приведен пример окна настроек метода «k-ближайших соседей».

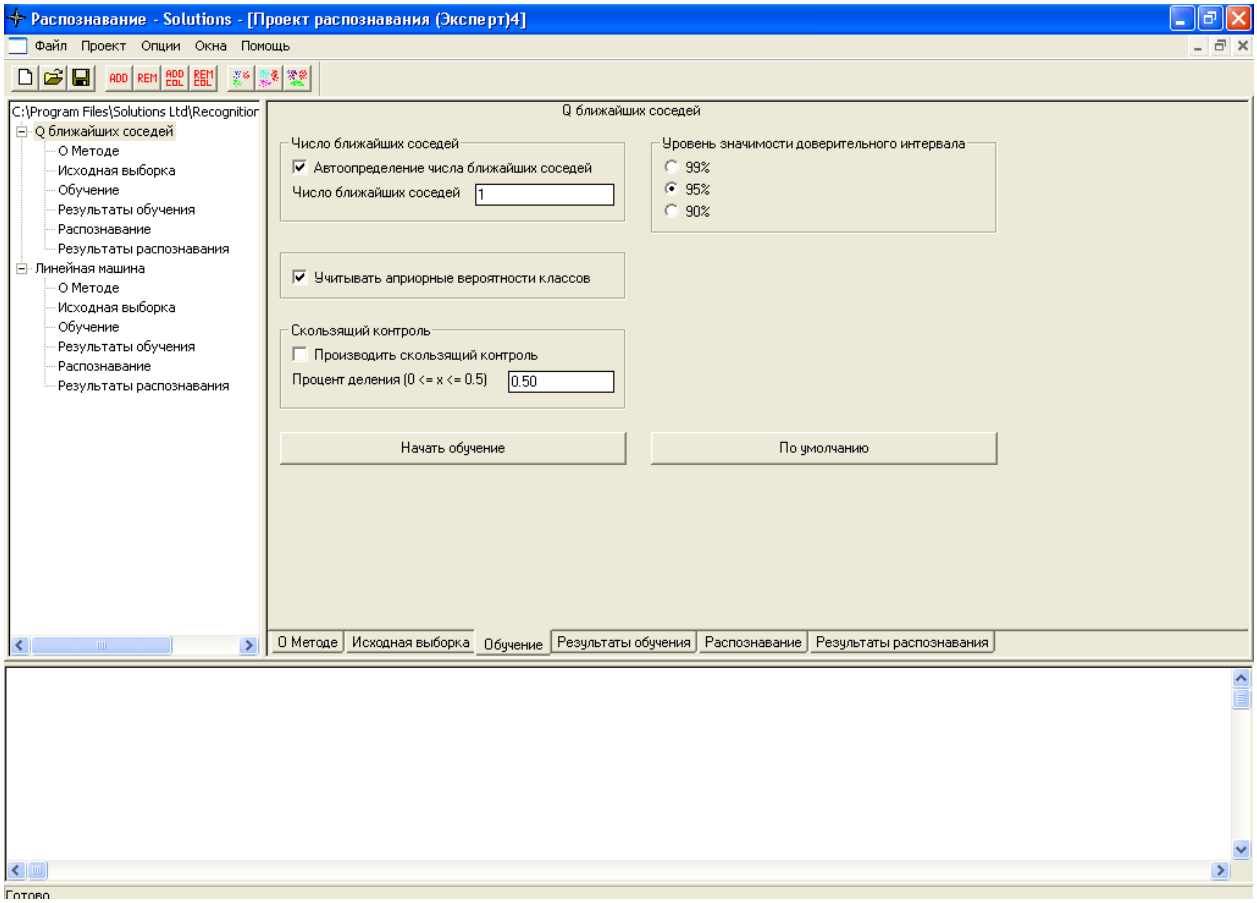

Рис. 45. Настройка метода

Для начала процесса обучения надо нажать на кнопку «Начать обучение». При нажатии на кнопку «По умолчанию», будут выставлены значения параметров по умолчанию. Для всех методов доступна процедура скользящего контроля, которая управляется при помощи групп параметров «Скользящий контроль» и «Уровень значимости доверительного интервала».

Закладки «Результаты обучения» и «Результаты распознавания».

Выглядят схожим образом, на них выводится отчет в виде HTML документа о результатах обучения и распознавания соответственно (см. рис. 46):

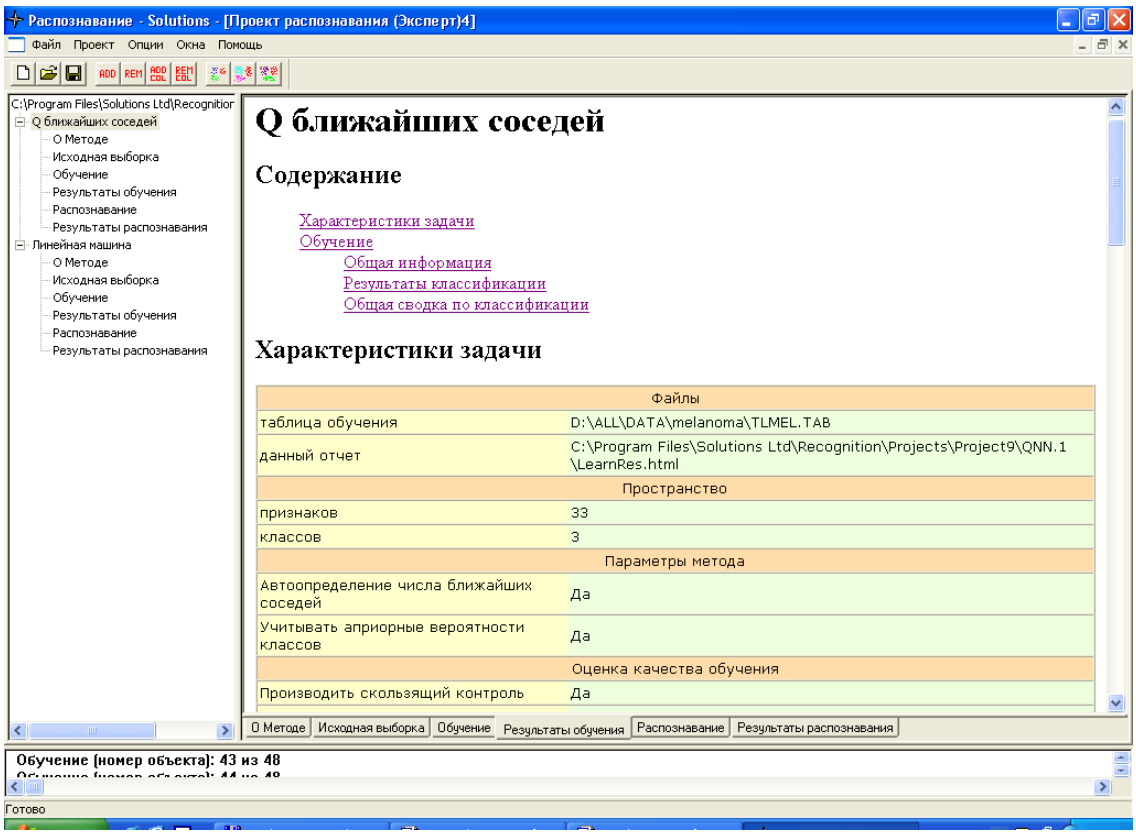

Рис. 46. Фрагмент отчета

Результаты скользящего контроля (если активизирована соответствующая опция) выводятся в виде таблицы результатов распознавания (классификации) элементов таблицы обучения (номер объекта, результат его классификации, оценки за классы, номер истинного класса объекта, см. рис. 47) и общей сводки по классификации (рис. 48).

Результаты классификации

Класс	Список объектов
1	1, 3÷9, 11÷16, 18÷20, 23÷27, 29÷31, 33÷37, 39÷48, 50÷58, 60÷71, 73÷79, 81÷84, 86÷101, 106, 112, 114, 117, 119, 123, 130, 134, 139, 150, 159, 169, 187, 196, 250, 258
2	2, 17, 21, 22, 28, 32, 38, 59, 72, 80, 85, 102÷105, 107÷111, 113, 115, 116, 118, 120÷122, 124÷129, 132, 133, 135÷137, 140÷145, 147÷149, 151÷158, 160÷168, 170, 171, 173÷180, 182÷186, 188÷195, 197÷200, 274, 301, 304, 316, 355, 356, 397
3	10, 49, 201÷249, 251÷257, 259÷273, 275÷300, 382, 391
4	131, 138, 146, 172, 181, 302, 303, 305÷315, 317÷354, 357÷381, 383÷390, 392÷396, 398÷400

Объект	Класс	За 1	За 2	За 3	За 4	Правильный класс	
1	1	**0.564**	0.494	0.484	0.458	1	успех
2	2	0.539	**0.550**	0.463	0.448	1	ошибка
3	1	**0.594**	0.490	0.486	0.430	1	успех
4	1	**0.568**	0.483	0.508	0.442	1	успех
5	1	**0.549**	0.518	0.479	0.455	1	успех
6	1	**0.525**	0.524	0.450	0.501	1	успех
7	1	**0.558**	0.506	0.484	0.452	1	успех

Рис. 47. Результаты классификации

Общая сводка по классификации

	Объектов, отнесенных в класс						
Класс	Объектов	Правильно	Ошибочно	Из класса 1	Из класса 2	Из класса 3	Из класса 4
1	104 (26.0%)	**87 (83.7%)**	17 (16.3%)	**87 (87.0%)**	15 (15.0%)	2 (2.0%)	0 (0.0%)
2	98 (24.5%)	**80 (81.6%)**	18 (18.4%)	11 (11.0%)	**80 (80.0%)**	1 (1.0%)	6 (6.0%)
3	101 (25.3%)	**97 (96.0%)**	4 (4.0%)	2 (2.0%)	0 (0.0%)	**97 (97.0%)**	2 (2.0%)
4	97 (24.3%)	**92 (94.8%)**	5 (5.2%)	0 (0.0%)	5 (5.0%)	0 (0.0%)	**92 (92.0%)**
Отказы	0 (0.0%)			0 (0.0%)	0 (0.0%)	0 (0.0%)	0 (0.0%)
	Итого						
	400 (100 %)	**356 (89.0%)**	44 (11.0%)	100 (25.0%)	100 (25.0%)	100 (25.0%)	100 (25.0%)

Рис. 48. Общая сводка по классификации

При обучении тестового алгоритма дополнительно выводится найденный список опорных множеств с оценками их информативности (рис.49).

Тест 11	0.0380085 (5 6)
Тест 12	0.0378082 (1 4)
Тест 13	0.0345126 (4 6)
Тест 14	0.0343924 (5 7)
Тест 15	0.0343244 (1 3)
Тест 16	0.0236573 (2 3 4)
Тест 17	0.0233667 (3 6 7)
Тест 18	0.0224514 (3 4 5)
Тест 19	0.0224471 (1 3 6)
Тест 20	0.0220881 (3 5 6)
Тест 21	0.0219087 (1 6 7)
Тест 22	0.0214892 (2 3 7)
Тест 23	0.0211248 (4 6 7)

Рис. 49. Опорные множества тестового алгоритма и их веса

При обучении метода «логические закономерности» файл отчета содержит дополнительно перечень логических закономерностей (статистически значимые выделены жирным шрифтом, рис.50), перечень эталонов, на которых выполняется соответствующая закономерность, и их доля от общего числа эталонов класса (рис. 51), веса признаков и перечень несократимых логических закономерностей (полученных из ранее найденных с помощью удаления части сомножителей, рис. 52).

Найденные закономерности	
Количество закономерностей	109
Закономерность №1	**(X3<=1.13147)(X4<=1.59382)(X7<=0)**
Закономерность №2	**(X1<=2.84986)(X2<=1.54519)(X3<=1.51755) (0.536456<=X4<=1.25122)(X5<=2.62045)(X6<=1.67057) (X7<=0)**
Закономерность №3	**(X1<=1.29771)(X5<=1.93238)(X6<=1.42637)**
Закономерность №4	**(X1<=1.79055)(X3<=1.81172)(X5<=1.65361)(-0.76<=X6<=0.682131)**
Закономерность №5	**(X1<=1.30158)(X2<=2.37435)(X3<=1.13147)(X4<=2.32963) (X6<=2.23654)(X7<=0)**
Закономерность №6	**(X1<=1.74539)(X3<=1.50376)(X6<=1.52133)(X7<=0)**
Закономерность №7	**(X1<=1.07452)(X5<=1.15188)(X6<=1.49614)**
Закономерность №8	(X1<=1.8486)(0.739733<=X2)(X3<=2.39092)(X4<=0.925881) (X5<=1.70303)(0<=X7<=0)
Закономерность №9	(X2<=2.57713)(X3<=1.18203)(0.536456<=X4<=1.63079)(X5<=0.89214) (X6<=0.755779)(X7<=0)
Закономерность №10	**(X1<=1.78732)(X3<=1.47561)(X5<=2.62036)(X6<=1.61242) (X7<=0)**

Рис. 50. Найденные логические закономерности

Веса закономерностей	
Вес закономерности №1	**4 5 6 7 9 12 19 26 31 34 35 36 37 41 44 45 46 55 62 63 65 68 69 71 75 78 83 92 96 97 99** : 0.31
Вес закономерности №2	**3 26 31 34 36 44 45 46 63 68 73 75 92 96** : 0.14
Вес закономерности №3	**1 2 3 4 7 13 20 23 27 30 33 39 42 48 49 51 52 54 56 57 60 63 64 68 69 70 71 74 75 76 77 80 84 87 90 94 96 97 99** : 0.39
Вес закономерности №4	**4 8 15 20 25 36 40 43 49 50 54 60 64 65 69 70 71 74 80 84 87 97** : 0.22
Вес закономерности №5	**1 5 7 9 12 18 23 34 37 45 51 53 54 55 56 62 63 64 68 69 71 74 75 84 92 93 96 97 98 99** : 0.3
Вес закономерности №6	**1 3 4 5 7 9 12 14 16 20 23 30 31 34 36 37 43 45 46 48 52 61 63 64 65 68 69 71 73 74 75 76 77 79 84 87 91 92 95 96 97 99** : 0.42
Вес закономерности №7	**1 3 7 30 39 42 48 49 52 54 56 57 61 69 70 71 76 80 84 87 94 99** : 0.22
Вес закономерности №8	**25 40 54 60 71 85** : 0.06

Рис. 51. Веса закономерностей

Анализ признаков	
Вес признака №1	0.518
Вес признака №2	0.516
Вес признака №3	0.721
Вес признака №4	0.695
Вес признака №5	0.736
Вес признака №6	0.661
Вес признака №7	0.652
Найденные закономерности	
Количество закономерностей	147
Закономерность №1, порожденная базовой №1	(X3<=1.13147)(X4<=1.59382)(X7<=0)
Закономерность №2, порожденная базовой №2	(X3<=1.51755)(0.536456<=X4<=1.25122)(X7<=0)
Закономерность №3, порожденная базовой №3	(X1<=1.29771)(X5<=1.93238)(X6<=1.42637)

Рис. 52. Веса признаков и несократимые закономерности

Отчеты о результатах распознавания аналогичны отчетам о результатах скользящего контроля.

Закладка «Распознавание».

На данной закладке отображается выборка, которая была загружена для распознавания. Управление изображением осуществляется точно таким же образом, как и в случае выборки для обучения. Отличается данная закладка только наличием кнопки «Распознать», которая запускает процесс распознавания (см. рис. 47):

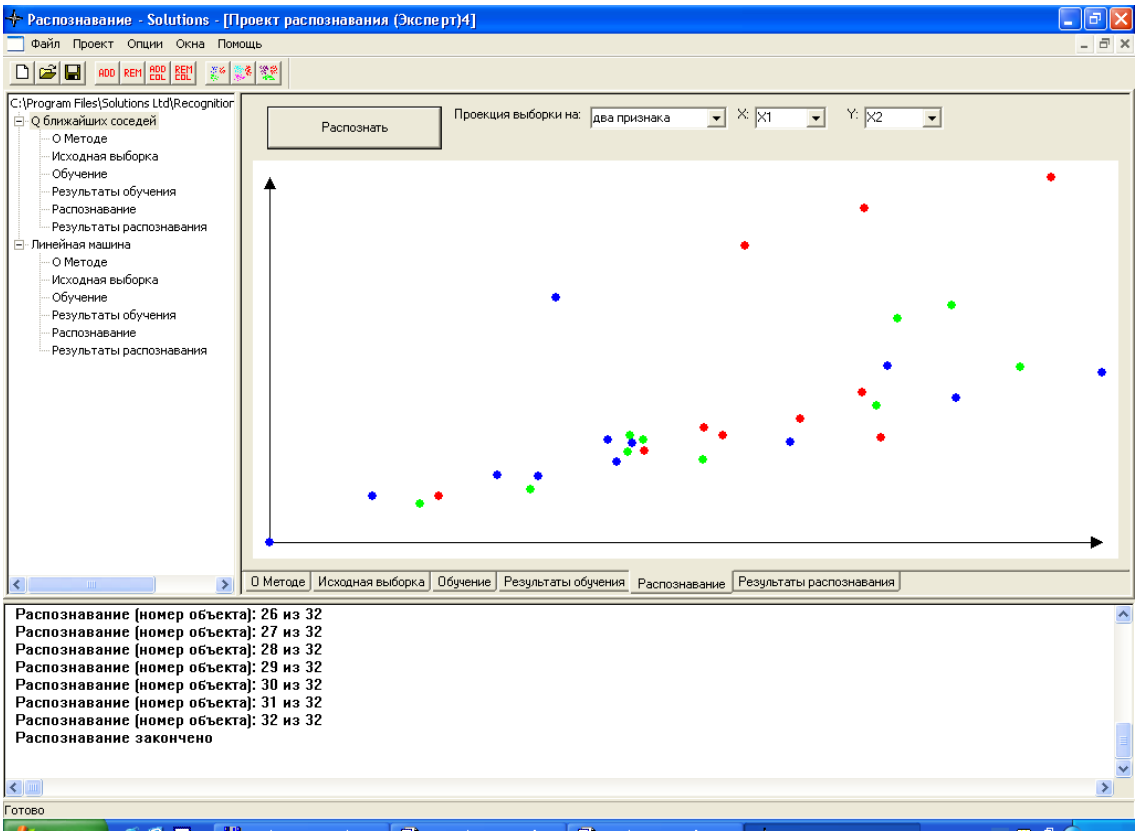

Рис. 53. Закладка «Распознавание»

5.1.5. Метод кластеризации

Управление методом кластеризации осуществляется по тому же принципу, что и методом распознавания. Отличие заключается в следующем. В случае кластеризации мы имеем не 6, а 5 управляющих закладок. Первые четыре из них аналогичны первым четырем закладкам метода распознавания. На пятой же закладке представлены результаты кластеризации в графическом виде в соответствии с результатами работы метода (см. рис. 39):

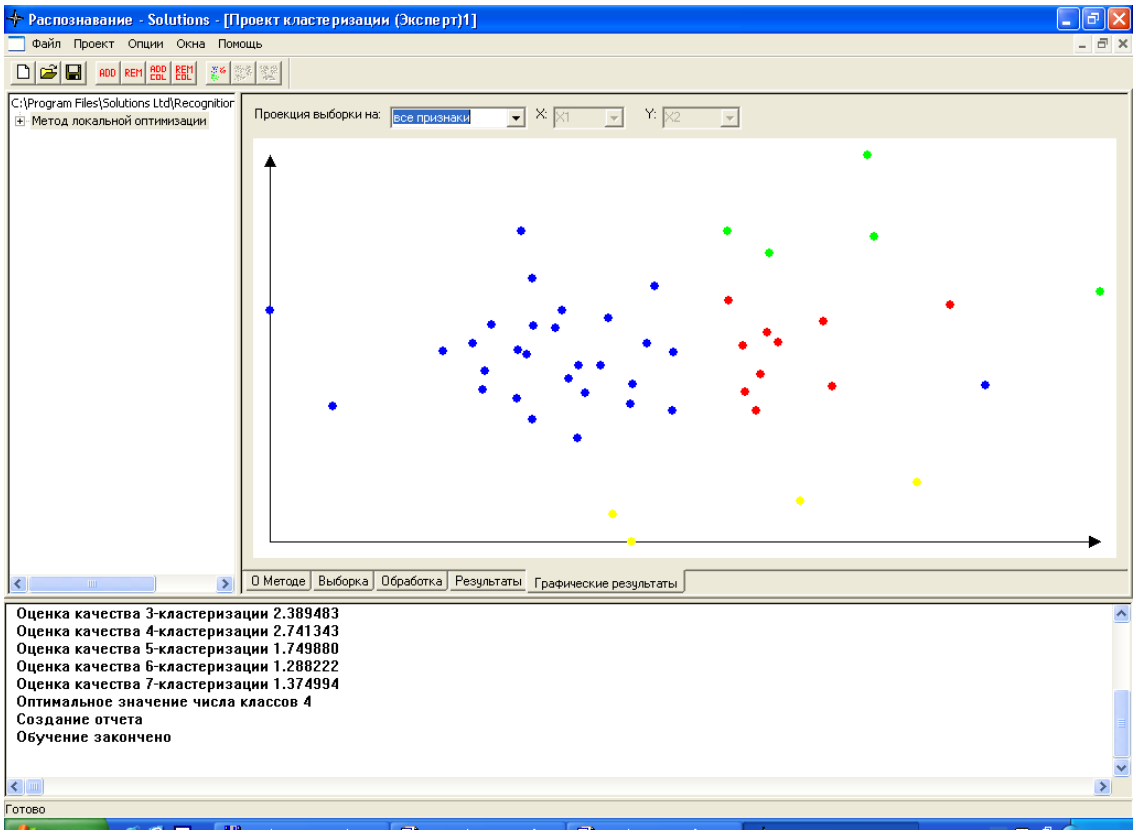

Рис. 54. Управление кластеризацией

5.2. Ввод и предобработка данных

В настоящий момент в системе реализованы два различных источника данных: текстовые файлы определенного формата и источники данных ODBC (в том числе файлы Microsoft Excel). После первичной обработки данных соответствующими модулями чтения (преобразование классообразующего признака в ODBC-reader'е), происходит дополнительная обработка данных для стандартизации работы методов.

Общий принцип любой обработки следующий: выборки подразделяются на «главные» и «зависимые». Главная выборка – это выборка, которая используется для обучения методов распознавания, зависимые – это выборки для обучения коллективных

методов и распознавания. Любые преобразования, совершаемые с зависимыми выборками, базируются на соответствующем преобразовании главной выборки.

5.2.1. Количественные признаки

В данном случае признаки по элементам главной выборки преобразуются так, чтобы в результате их выборочные средние и дисперсии были равны, соответственно, 0 и 1. Для зависимых выборок при преобразовании признаков используются значения сдвига и коэффициенты масштабирования, полученные на главной выборке.

5.2.2. Обработка номинальных признаков

Признак считается номинальным в том случае, если среди его значений встречается хотя бы одно нечисловое (символьное) значение.

В случае главной выборки последовательно перебираются все значения. Если некоторое значение не встречалось ранее, то ему присваивается очередной номер и его значение заменяется на этот номер. Если некоторое значение встречалось ранее, то используется номер, который был создан для него ранее. Нумерация начинается с 0.

В случае зависимого признака используются наборы значений и порядковые номера, которые были созданы для соответствующего признака главной выборки. Если некоторое значение не встречается среди значений главного признака, то оно считается неизвестным.

Если признак по главной выборке определен как числовой, то любое нечисловое значение обрабатывается как неизвестное значение признака.

5.2.3. Неизвестные значения

Если среди значений некоторого признака встречаются неизвестные, то они заменяются на средние значения соответствующего признака. При этом если некоторый метод поддерживает обработку неизвестных значений специальным образом, то приоритет отдается способу данного метода.

5.2.4. Замена главной выборки

Если пользователь загрузил новую главную выборку, то все зависимые от нее выборки будут обработаны в соответствии с новой главной выборкой.

5.2.5. Классообразующий признак

Классообразующий признак преобразуется также, как и обычный номинальный признак. При этом несущественно, является ли он числовым или нет.

Существенные отличия:

- Нумерация начинается с 1.
- Значения не смещаются и не масштабируются, они всегда 1,2,3,...
- Неизвестные значения заменяются на -1.

5.3. Структура программы

Вся система разбита на модули, которые реализуют разработанный и жестко зафиксированный интерфейс. Основным модулем программы является **Engine**, отвечающий за управление всеми процессами, происходящими в системе, и связь между модулями. Вторым важным модулем является **GUI** (graphical user interface), который отвечает за графический интерфейс пользователя, то есть меню, панели инструментов, графическое представление данных и результатов работы методов. Завершают список группа однотипных модулей обработки данных, реализующих различные математические методы распознавания и кластеризации, и группа модулей загрузки данных. Общая структура системы представлена на рисунке 55.

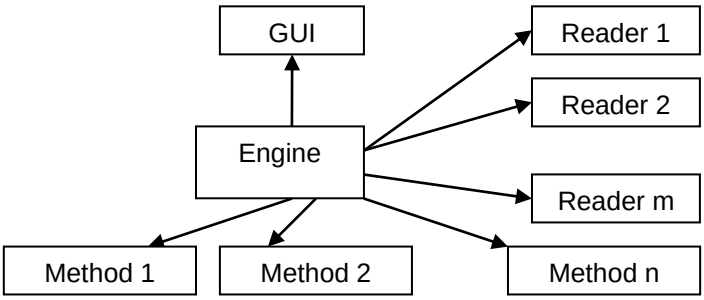

Рис. 55. Управление кластеризацией

Такое устройство позволяет, во-первых, проводить независимую модернизацию различных частей программы, и, во-вторых, свободно наращивать математическую часть и добавлять новые источники данных. Более того, фиксированный интерфейс обмена данных между модулями предоставляет конечному пользователю возможность написания

собственных методов и присоединения их к системе без участия первоначальных разработчиков.

Еще одной важной особенностью организации программы является широкое использование динамического связывания. Каждый метод реализован в виде отдельной динамической библиотеки и может модернизироваться без затрагивания других частей программы. Кроме того, не требуется производить каких-то сложных процедур установки новых методов или модулей чтения – новые модули подключаются «на лету», простым переписыванием соответствующей библиотеки в директорию методов или источников данных.

Различные математические методы и полезные утилиты собраны в отдельной статической библиотеке. Это позволяет использовать разработанные и уже отлаженные эффективные процедуры, такие как решение задачи целочисленного линейного программирования или нахождения максимальной совместной подсистемы неравенств, в новых методах распознавания.

Важно отметить так же существенную параллельность работы системы, которая выражается в том, что все процессы обучения, распознавания, загрузки и обработки данных выполняются в разных потоках. Это обстоятельство имеет важные последствия как для удобства работы с программой (пользователь получает возможность работать с графической оболочкой в то время, как могут обучаться методы), так и для скорости вычислений. При использовании многопроцессорной машины это дает существенное увеличение быстродействия программного комплекса. Эта возможность позволяет также распределять вычисления по сети в том случае, если библиотеки методов реализованы не в виде динамических библиотек, а в виде объектов DCOM.

СПИСОК ИСПОЛЬЗОВАННЫХ

1. Айвазян С.А. и др. ПРИКЛАДНАЯ СТАТИСТИКА: Классификация и снижение размерности, М. Финансы и статистика, 1989.
2. Айзерман М.А., Браверманн Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. - М.: Наука, 1970.-384 с.
3. Баскакова Л.В., Журавлев Ю.И. Модель распознающих алгоритмов с представительными наборами и системами опорных множеств //Журн. вычисл. матем. и матем. физики. 1981. Т.21, № 5. С.1264-1275.
4. Бирюков А.С., Виноградов А.П., Долгих Н.А., Рязанов И.В., Рязанов В.В., Оперативная обработка данных дистанционного зондирования в целях прогнозирования. Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001, 169-172.
5. Богачев А.В., Булавин Е.С., Рязанов В.В., Анализ материалов многозональной съемки с целью определения преобладающих пород. Экономико-математическое моделирование лесохозяйственных мероприятий. Л.:Лен.НИИ ЛХ. 1980. С.58-62.
6. Богомолов В.П., Виноградов А.П., Ворончихин В.А., Журавлев Ю.И., Катериночкина Н.Н., Ларин С.Б., Рязанов В.В., Сенько О.В. Программная система распознавания ЛОРЕГ : алгоритмы распознавания, основанные на голосовании по системам логических закономерностей, М.: ВЦ РАН , 1998. 63 с.
7. Богомолов В.П., Виноградов А.П., Ларин С.Б., Рязанов В.В., О некоторых результатах распознавания рукописных цифр с использованием моделей распознавания, основанных на принципе частичной прецедентности, Доклады 7-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Пущино, 1995, 75-76.
8. Бонгард М.М. Проблема узнавания. М.: Наука, 1967. 320 с.
9. Бушманов О.Н., Дюкова Е.В., Журавлев Ю.И., Кочетков Д.В., Рязанов В.В. Система анализа и распознавания образов, Распознавание, классификация, прогноз: Мат. методы и их применение. М.:Наука, 1988. Вып.2. С.250-273.
10. Вайнцвайг М.Н. Алгоритм обучения распознаванию образов "Кора" // Алгоритмы обучения распознаванию образов. М.: Сов.радио, 1973. С. 8-12.
11. Вапник В.Н., Червоненкис А.Я. Теория распознавания образов (статистические проблемы обучения). – М.:Наука, 1974.-415 с.
12. Виноградов А.П. , Журавлев Ю.И., Рязанов В.В., Чернявский Г.М., Разработка системы оперативного прогнозирования сельскохозяйственного урожая на территории РФ, Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001, 217-219.
13. Воликов Ю.К., Дюкова Е.В., Левашов Е.А., Рязанов В.В. Применение методов распознавания образов для прогнозирования свойств твердых сплавов группы СТИМ, Структура, свойства и технология металлических систем и керамик. Сб.научных трудов. Моск.инст.стали и сплавов. Москва, 1988.
14. Даровских И.В, Кузнецова А.В., Сенько О.В., Реброва О.Ю. Прогноз динамики депрессивных синдромов в остром периоде сотрясения головного мозга по показателям первичного обследования. “Социальная и клиническая психиатрия” N 4, 2003г., с. 18-24.
15. Дмитриев А.Н., Журавлев Ю.И., Кренделев Ф.П., О математических принципах классификации предметов и явлений. Сб. "Дискретный анализ". Вып. 7. Новосибирск, ИМ СО АН СССР. 1966. С. 3-11.
16. Донской В.И. Алгоритмы обучения, основанные на построении решающих деревьев// Журнал выч. мат. и матем. физики. 1982, т.22, №4, с. 963-974.
17. Донской В.И. , Башта А.И. Дискретные модели принятия решений при неполной информации. –Смферополь: Таврия, 1992. – 166 с.
18. Дородницына В.В. Классификация стран по возрастному распределению и ее связь с характеристиками заболеваний. –М.: ВЦ АН СССР. – 1985.-19 с.

19. Дуда Р., Харт П., Распознавание образов и анализ сцен. Издательство "Мир", Москва, 1976, 511 с.
20. Дюк В., Самойленко А., Data Mining: учебный курс – СПб: Питер, 2001. – 368 с.
21. Дюкова Е.В. Асимптотически оптимальные тестовые алгоритмы в задачах распознавания// Проблемы кибернетики. М.: Наука, 1982. Вып. 39. С. 165-199.
22. Дюкова Е.В. Алгоритмы распознавания типа “Кора”: сложность реализации и метрические свойства// Распознавание, классификация, прогноз (матем. методы и их применение). М.: Наука, 1989. Вып.2. С. 99-125.
23. Дюкова Е.В., Журавлев Ю.И., Рязанов В.В. Диалоговая система анализа и распознавания образов. М.: ВЦАН СССР, 1988, 40 с.
24. Еремин И.И. О системах неравенств с выпуклыми функциями в левых частях. Известия АН СССР, серия математическая, 30 (1966), стр. 265-278.
25. Журавлев Ю.И., ИЗБРАННЫЕ НАУЧНЫЕ ТРУДЫ. - М.: Издательство Магистр, 1998. - 420 с.
26. Журавлев Ю.И., Об алгебраическом подходе к решению задач распознавания или классификации. Проблемы кибернетики. М.: Наука, 1978. Вып.33. С.5-68.
27. Журавлев Ю.И., Никифоров В.В. Алгоритмы распознавания, основанные на вычислении оценок // Кибернетика. 1971. №3. С. 1-11.
28. Журавлев Ю.И. Корректные алгебры над множествами не корректных (эвристических) алгоритмов. I. // Кибернетика. 1977. N4. С. 5-17. , II. Кибернетика, N6, 1977, III. // Кибернетика. 1978. N2. С. 35-43.
29. Журавлев Ю.И., Зенкин А.И. Зенкин А.А., Дюкова Е.В., Исаев И.В., Кочетков Д.В., Кольцов П.П., Рязанов В.В. Пакет прикладных программ для решения задач распознавания и классификации (ПАРК). ВЦ АН СССР, Москва, 1981. 24 с.
30. Загоруйко Н.Г. Методы распознавания и их применение.-М. : Сов.радио, 1972. -206 с.
31. Загоруйко Н.Г. Прикладные методы анализа данных и знаний. Новосибирск: Изд-во Института математики, 1999.
32. Зуев Ю.А. Метод повышения надежности классификации при наличии нескольких классификаторов, основанный на принципе монотонности// ЖВМиМФ. 1981. Т.21. № 1. С.157-167.
33. Ивахненко А.Г. Системы эвристической самоорганизации в технической кибернетике. –Киев: Техніка, 1971.-372 с.
34. Катериночкина Н.Н.. Методы выделения максимальной совместной подсистемы системы линейных неравенств. Сообщение по прикладной математике. ВЦ РАН. 1997. С. 1-15.
35. Кочетков Д.В. Распознающие алгоритмы, инвариантные относительно преобразований пространства признаков // Распознавание , классификация, прогноз: Мат. методы и их применение. М.: Наука. 1989. Вып. 11. С. 178-206.
36. Краснопрошин В.В. Об оптимальном корректоре совокупности алгоритмов распознавания//ЖВМиМФ. 1979. Т.19. №1. С. 204-215.
37. Кузнецов В.А., Сенько О.В., Кузнецова А.В. и др. // Распознавание нечетких систем по методу статистически взвешенных синдромов и его применение для иммуногематологической нормы и хронической патологии. // Химическая физика, 1996, т.15 . №1. С. 81-100.
38. Кузнецов Н.П., Ромасевич А.Н., Рязанов В.В. Опыт решения инженерно-технических задач на основе методов таксономии и распознавания. Применение прогрессивных технологий в добыче нефти на месторождениях Западной Сибири, Сб. научных трудов. Тюмень. СибНИИНП, 1988, С.62-66.
39. Кузоваткин Р.И., Зенкин А.И., Рязанов В.В., Егоров П.И. Распознавание солеобразующей способности нефтяных скважин по эксплуатационным признакам. Проблемы нефти и газа Тюмени: Научн.техн. сб. Тюмень. 1978. Вып. 38. С. 39-41
40. Кузоваткин Р.И., Зенкин А.И., Рязанов В.В., Егоров П.И., Чернобай Л.А.,

Прогнозирование процесса солеобразования при эксплуатации нефтяных скважин, Проблемы нефти и газа Тюмени: Научн.техн. сб. Тюмень. 1979. Вып.42. С. 42-44.
41. Лбов Г.С. Методы обработки разнотипных экспериментальных данных// Новосибирск. Наука, 1981. 160 с.
42. Мазуров Вл.Д. Комитеты систем неравенств и задача распознавания // Кибернетика. 1971. №3. С. 140-146.
43. Мазуров Вл.Д., Хачай М.Ю. Комитеты систем линейных неравенств// Автоматика и телемеханика. 2004. вып.2, С. 43-54.
44. Матросов В.Л. Синтез оптимальных алгоритмов в алгебраических замыканиях моделей алгоритмов распознавания// Распознавание, классификация, прогноз: Матем. методы и их применение. М.: Наука, 1988. Вып.1, С.229-279.
45. Метод комитетов в распознавании образов. Свердловск: ИММ УНЦ АН СССР, 1974. 165 с.
46. Обухов А.С., Рязанов В.В. Применение релаксационных алгоритмов при оптимизации линейных решающих правил. Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001, 102-104.
47. Растригин Л.А., Эренштейн Р.Х. Принятие решений коллективом решающих правил в задачах распознавания образов// АиТ. 1975. №9. С.133-144.
48. Розенблатт Ф. Принципы нейродинамики (перцептрон и теория механизмов мозга). – М. : Мир, 1965.-480 с.
49. Рудаков К.В. Об алгебраической теории универсальных и локальных ограничений для задач классификации // Распознавание, классификация, прогноз: Матем. методы и их применение. М.: Наука, 1988. Вып.1, С.176-200.
50. Рязанов В.В. О построении оптимальных алгоритмов распознавания и таксономии (классификации) при решении прикладных задач // Распознавание, классификация, прогноз: Матем. методы и их применение. М.: Наука, 1988. Вып.1, С.229-279.
51. Рязанов В.В. Комитетный синтез алгоритмов распознавания и классификации // ЖВМ и МФ. 1981. Том 21, №6. С.1533-1543.
52. Рязанов В.В. О синтезе классифицирующих алгоритмов на конечных множествах алгоритмов классификации (таксономии) // ЖВМ и МФ, 1982. Том 22, №2. С.429-440.
53. Рязанов В.В., Рязанова Н.В., Смольянинова З.А. Прогноз урожайности сельскохозяйственных культур с использованием методов распознавания образов, Методы моделирования в сельском хозяйстве. Сб. научн. трудов. М.: ВНИПТИК. 1982. С.50-60.
54. Рязанов В.В., Челноков Ф.Б., О склеивании нейросетевых и комбинаторно-логических подходов в задачах распознавания, Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001,115-118.
55. Себастьян Г.С. Процессы принятия решений при распознавании образов. М., Изд-во «Техника», 1965
56. Сенько О.В. Использование процедуры взвешенного голосования по системе базовых множеств в задачах прогнозирования// М. Наука, Ж. вычисл. матем. и матем. физ. 1995, т. 35, № 10, С. 1552-1563.
57. Уоссермен Ф., Нейрокомпьютерная техника, М.,Мир, 1992.
58. Фукунага К. Введение в статистическую теорию распознавания образов. М.:Наука. 1979. 367 с.
59. Чегис И.А., Яблонский С.В. Логические способы контроля электрических схем // Труды Матем. ин-та им. В.А.Стеклова АН СССР. 1958. Т. 51. С. 270-360.
60. Aslanyan L., Zhuravlev Yu,.Logic Separation Principle, Computer Science & Information Technologies Conference, Yerevan, September 17-20, 2001, 151-156.
61. Bogomolov V.P., Larin S.B., Vinogradov A.P., Voronchihin V.A., Ryazanov V.V., Senco O.V., Zhuravlev Yu.I., Segmentation and Recognition of Handwritten Digits with the Use of Precedent-Based Models. Pattern Recognition and Image Analysis. 1998. Vol.8. no.2.

62. Christopher J.C. Burges. A Tutorial. on Support Vector Machines for Pattern Recognition, Appeared in: Data Mining and Knowledge Discovery 2, 121-167, 1998.
63. Fisher R.A. The use of multiple measurements in taxonomic problems, Ann. Eugenics, 7, Part II, 179-188 (1936).
64. Fu K. S. Sequential Methods in Pattern Recognition and Machine Learning (Academic Press, New York, 1968).
65. Ganster H., Gelautz M., Pinz A., Binder M., Pehamberger H., Bammer M., Krocza J. Initial Results of Automated Melanoma Recognition //Proceedings of the 9th Scandinavian Conference on Image Analysis, Uppsala, Sweden, June 1995, Vol.1, pp. 209-218.
66. Harrison, D. and Rubinfeld, D.L. Hedonic prices and the demand for clean air, J. Environ. Economics & Management, vol.5, 81-102, 1978.
67. Horton Paul, Nakai Kenta //"A Probablistic Classification System for Predicting the Cellular Localization Sites of Proteins",Intelligent Systems in Molecular Biology, 109-115. St. Louis, USA 1996.
68. Kiselyova N.N. // Search for efficient methods of prediction of multicomponent inorganic compounds (final report). EOARD SPC-00-4014, 2001. - 46 p.
69. Kuncheva L.I. Combining Pattern Classifiers: Methods and Algorithms, Wiley, 2004.
70. Kuznetsova A.V., Sen'ko O.V., Matchak G.N., Vakhotsky V.V., Zabotina T.N., Korotkova O.V. The Prognosis of Survivance in Solid Tumor Patients Based on Optimal Partitions of Immunological Parameters Ranges, Journal Theoretical Medicine, 2000, Vol. 2, pp.317-327.
71. Larin S.B., Ryazanov V.V. The Search of Precedent-Based Logical Regularities for Recognition and Data Analysis Problems // Pattern Recognition and Image Analysis. 1997. Vol.7. no.3. P.322-333.
72. Mangasarian O. L., Wolberg W.H.: "Cancer diagnosis via linear programming", SIAM News, Volume 23, Number 5, September 1990, pp 1 - 18.
73. Minsky M., Papert S. Perceptrons: An Introduction to Computational Geometry (MIT Press, Cambridge, Mass., 1969). Русский перевод : Перцептроны, М., «Мир», 1971.
74. Neyman J., Pearson E.S. On the problem of the most efficient tests of statistical hypothesis,
75. Nilsson N.J. Learning Machines (McGraw-Hill, New York, 1965). Русский перевод: Обучающие машины, М., «Мир», 1967.
Phill. Trans. Royal Soc. London, 231, 289-337 (1933).
76. Ryazanov V.V., Recognition Algorithms Based on Local Optimality Criteria , Pattern Recognition and Image Analysis. 1994. Vol.4. no.2. pp. 98-109.
77. Ryazanov V.V. About some approach for automatic knowledge extraction from precendent data // Proceedings of the 7th international conference "Pattern recognition and image processing", Minsk, May 21-23, 2003, vol. 2, pp. 35-40.
78. Ryazanov V.V., Senko O.V., and Zhuravlev Yu. I.. Methods of recognition and prediction based on voting procedures.// Pattern Recognition and Image Analysis, Vol. 9, No. 4, 1999, p.713—718.
79. Ryazanov V.V. One approach for classification (taxonomy) problem solution by sets of heuristic algorithms, Proceedings of the 9-th Scandinavian Conference on Image Analysis, Uppsala, Sweden, 6-9 June 1995, Vol.2, P.997-1002.
80. Ryazanov V.V., Sen'ko O.V., Zhuravlev Yu.I., Mathematical Methods for Pattern Recognition : Logical, Optimization, Algebraic Approaches Proceedings of the 14th International Conference on Pattern Recognition. Brisbane, Australia, August 1998, pp. 831-834.
81. Ryazanov V.V.,Vorontchikhin V.A. Discrete Approach for Automatic Knowledge Extraction from Precedent Large-scale Data, and Classification Proceedings of the 16th International Conference on Pattern Recognition. Quebec, Canada, 2002, 11-15 August , 188-191.
82. Ryazanov V.V. On the Optimization of a Class of Recognition Models// Pattern Recognition and Image Analysis. 1991. Vol.1. No.1. P.108-118.

83. Sigillito, V. G., Wing, S. P., Hutton, L. V., \& Baker, K. B. (1989). Classification of radar returns from the ionosphere using neural networks. Johns Hopkins APL Technical Digest, 10, 262-266
84. Vetrov D.P. On the Stability of the Pattern Recogntion Algorithms. // Pattern Recognition and Image Analysis, Vol.13, No.3, 2003, pp.470-475.
85. Wald A. Contributions to the theory of statistical estimation and testing of hypotheses, Ann.Math.Stat.,10, 299-326 (1939).

Предметный указатель

СОДЕРЖАНИЕ