4 Модели распознавания, основанные на различных способах обучения

Статистические методы распознавания нередко обеспечивали достаточно высокую точность в прикладных исследованиях**.** Однако в различных областях науки и практической деятельности возникали задачи диагностики и прогнозирования, которые могли быть сведены к задачам распознавания. При этом исследователям удавалось собрать обучающую выборку весьма ограниченного объёма. а число показателей, которые потенциально могли быть использованы оказывалось достаточно большим. Для решения таких задач стали предлагаться новые подходы, не содержащие предположений о лежащих в основе изучаемого процесса вероятностных распределений. Оказалось, что такие подходы часто имеют более высокую эффективность, чем статистические методы.

4.1 Метод Линейная машина

. Метод «Линейная машина» предназначен для решения задачи распознавания с классами $K_1,\ldots,K_L$ . .

В процессе обучения классам $K_1,\ldots,K_L$ ставятся в соответствие линейные функции $f_1,\ldots,f_L$ от переменных $X_1,\ldots,X_n$, являющиеся оценками за классы $K_1,\ldots,K_L$. То есть для произвольного вектора значений переменных $\mathbf{x}=(x_1,\ldots,x_n)$

$$f_1(\mathbf{x}) = w_0^1 + w_1^1 x_1 + \ldots + w_n^1 x_n$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$f_L(\mathbf{x}) = w_0^L + w_1^L x_1 + \ldots + w_n^L x_n$$

Для того, чтобы распознать объект s, описание которого задаётся вектором $\mathbf{x}$. вычисляются значения функций $f_1,\ldots,f_L$ в точке $\mathbf{x}$. Объект s будет отнесён классу K_l, если выполняется набор неравенств: $f_l(\mathbf{x}) > f_j(\mathbf{x}),\ j \in \{1,\ldots,L\} \setminus \{l\}$

Таким образом алгоритм распознавания задаётся матрицей вещественных параметров

$$\mathbf{W} = \begin{pmatrix} w_0^1 \, w_1^1 \ldots \, w_n^1 \\ \ldots\ldots\ldots\ldots \\ w_0^L \, w_1^L \ldots w_n^L \end{pmatrix} .$$

Обучения ведётся по выборке $\tilde{S}_t = \{s_1 = (y_1, \mathbf{x}_1), \ldots, s_m = (y_m, \mathbf{x}_m)\}$, где $\{y_1, \ldots, y_m\}$ являются значениями дискретной прогнозируемой переменной, указывающей на номер класса, которому принадлежит соответствующий объект. Обучение состоит в поиске таких значений параметров из матрицы $\mathbf{W}$, при которых максимальное число объектов $\tilde{S}_t$ оказывается правильно распознанным. Обозначим через $r(i)$ номер класса, которому принадлежит объект s_i из обучающей выборки. Максимальная точность на $\tilde{S}_t$ соответствует выполнению максимального числа блоков неравенств:

$$f_{r(1)}(\mathbf{x}_1) > f_j(\mathbf{x}_1),\ j \in \{1, \ldots, L\} \setminus \{r(1)\}$$

$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$

$$f_{r(m)}(\mathbf{x}_m) > f_j(\mathbf{x}_m),\ j \in \{1, \ldots, L\} \setminus \{r(m)\}$$

Каждый из блоков соответствует одному из объектов выборки включает $l-1$ неравенство. Таким образом суммарное число неравенств составляет $(l-1)m$.

Поиск оптимальной матрицы коэффициентов $\mathbf{W}$ производится с помощью релаксационного алгоритма, подробно описанного в книге [10].

Приведём графический пример алгоритма распознавания, построенного с помощью метода линейная машина. Имеется задача распознавания с классами 1, 2, 3 по признакам X_1 и X_2

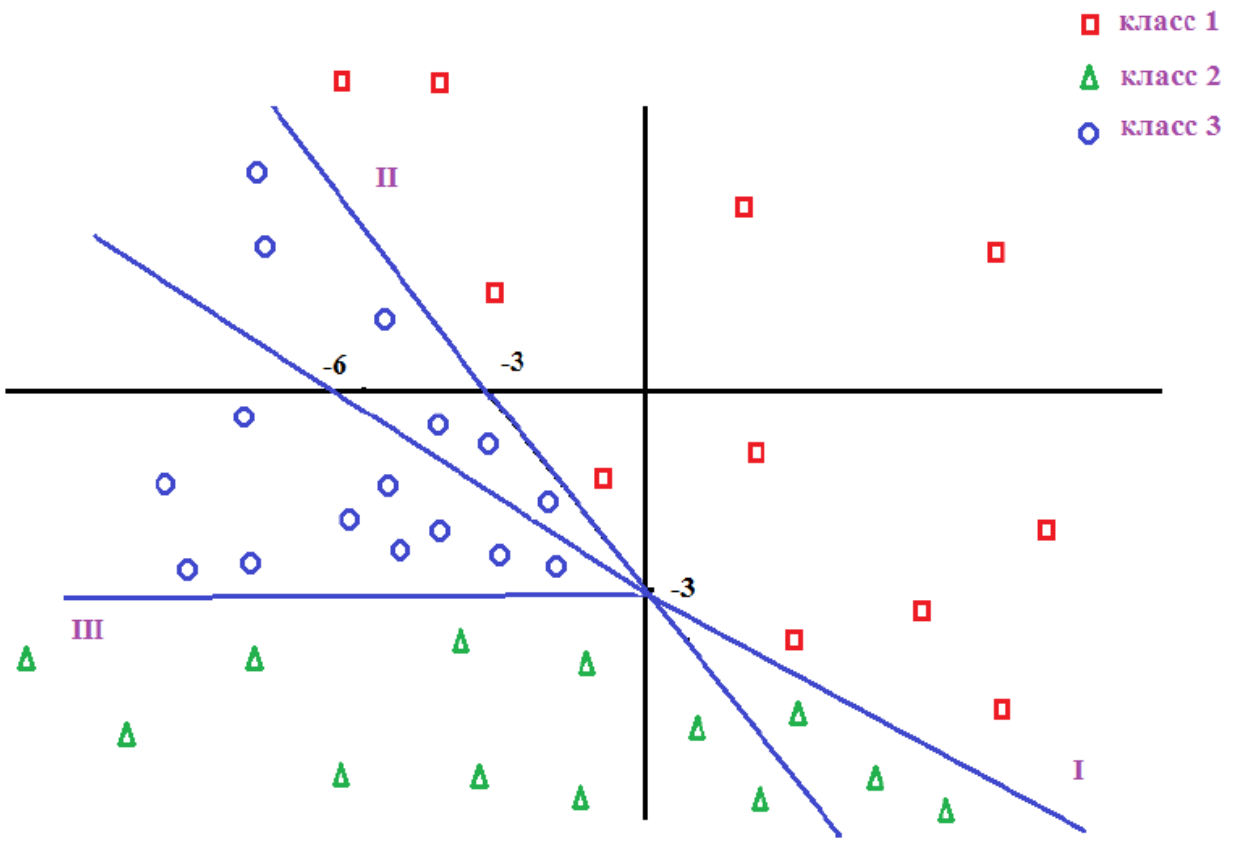

Рис. 1 Области, соответствующие отнесению распознаваемых объектов классам $K_1,\ldots,K_3$ методом линейная машина, вычисляющим оценки за классы по формулам (1).

Предполагается, что с использованием метода ЛМ для каждого класса найдены линейные функции оценок:

$$f_1(x_1,x_2)=4.0+2x_1-x_2 \quad \text{для класса 1;}$$

$$f_2(x_1,x_2)=-2.0+x_1-3x_2 \quad \text{для класса 2;} \qquad (1)$$

$$f_3(x_1,x_2)=1.0+x_1-2x_2 \quad \text{для класса 3.}$$

Изобразим на двумерной диаграмме области, соответствующие отнесению классов.

Очевидно, что классу 1 соответствует область, для которой выполняются неравенства $f_1(x_1,x_2)>f_2(x_1,x_2)$ и $f_1(x_1,x_2)>f_3(x_1,x_2)$. Неравенство $f_1(x_1,x_2)>f_2(x_1,x_2)$ эквивалентны неравенству $6+x_1+2x_2>0$, задающему границу I. Неравенство

$f_1(x_1, x_2) > f_3(x_1, x_2)$ эквивалентны неравенству $3 + x_1 + x_2 > 0$, задающему границу II. Область, соответствующая классу 1, помечена красными квадратами. Область, не относящаяся классу 1 при выполнении неравенства $f_2(x_1, x_2) > f_3(x_1, x_2)$.

Метод линейная машина подробно описан в книге [10].

4.2 Нейросетевые методы

4.2.1 Модель искусственного нейрона.

В основе нейросетевых методов лежит попытка компьютерного моделирования процессов обучения, используемых в живых организмах. Когнитивные способности живых существ связаны с функционированием сетей связанных между собой биологических нейронов – клеток нервной системы. Для моделирования биологических нейросетей используются сети, узлами которых являются искусственные нейроны (т.е. математические модели нейронов), Можно выделить три типа искусственных нейронов: нейроны-рецепторы, внутренние нейроны и реагирующие нейроны. Каждый внутренний или реагирующий нейрон имеет множество входных связей, по которым поступают сигналы от рецепторов или других внутренних нейронов. Пример модели внутреннего или реагирующего нейрона представлен на рисунке 1.

Представленный на рисунке 1 нейрон имеет r внешних связей, по которым на него поступают входные сигналы $u_1, \ldots, u_r$. Поступившие сигналы суммируются с весами $w_1, \ldots, w_r$. На выходе нейрона вырабатывается сигнал $\Phi(z)$, где $z = w_0 + \sum_{i=1}^{r} w_i u_i$, w_0 - параметр сдвига. Может быть использована также форма записи $z = \sum_{i=0}^{r} w_i u_i$, где фиктивный «сигнал» u_0 тождественно равен 1.

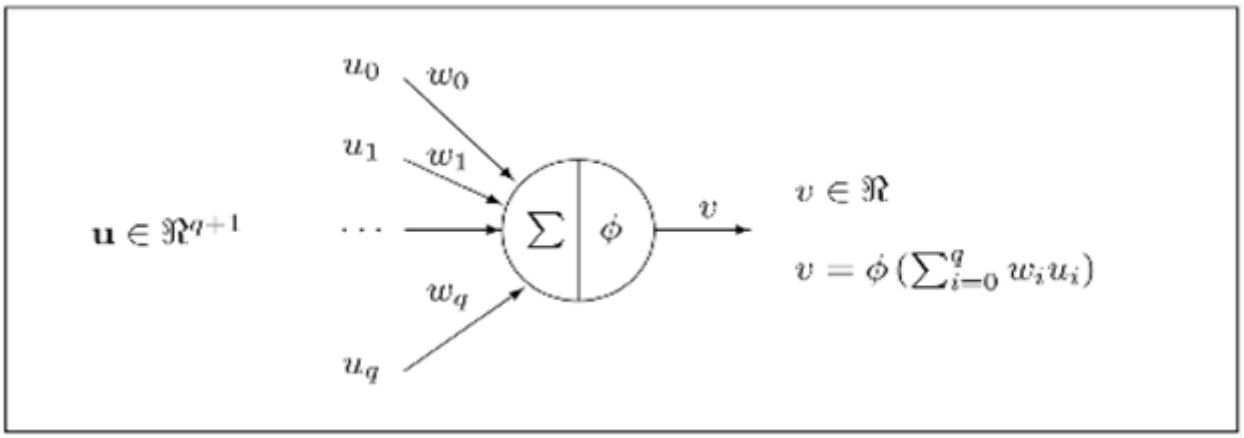

Рис.1. Модель внутреннего или реагирующего нейрона.

Функцию $\Phi(z)$ обычно называют активационной функцией. Могут использоваться различные виды активационных функций, включая

a) пороговую функцию, задаваемую с помощью пороговой величины b:

$$\Phi(z) = \begin{cases} 1 \text{ при } z > b \\ -1 \text{ при } z \le b \end{cases},$$

b) сигмоидная функция $\Phi(z) = \dfrac{1}{1+e^{-az}}$, где a - вещественная константа;

c) гиперболический тангенс;

d) тождественное преобразование $\Phi(z) = z$.

Первой нейросетевой моделью стал перцептрон Розенблатта, предложенный в 1957 году. В данной модели используется единственный реагирующий нейрон. Модель, реализующая линейную разделяющую функцию в пространстве входных сигналов, может быть использована для решении задач распознавания с двумя классами, помеченными метками 1 или -1. В качестве активационной функции используется пороговая функция:

$$\Phi(z) = \begin{cases} 1 \text{ при } z > 0 \\ -1 \text{ при } z \le 0 \end{cases}.$$

Особенностью модели Розенблатта является очень простая, но вместе с тем эффективная, процедура обучения, вычисляющая значения весовых коэффициентов

$w_0,\ldots,w_n$. Настройка параметров производится по обучающим выборкам, совершенно аналогичных тем, которые используются для обучения статистических алгоритмов.

На первом этапе производится преобразование векторов сигналов (признаковых описаний) для объектов обучающей выборки. В набор исходных признаков добавляется тождественно равная 1 нулевая компонента. Затем вектора описаний из класса K_2 умножаются на -1. Вектора описаний из класса K_1 не изменяются.

Нулевое приближение вектора весовых коэффициентов $w_0^0,\ldots,w_n^0$ выбирается случайным образом. Преобразованные описания объектов обучающей выборки $\tilde{S}_t$ последовательно подаются на вход перцептрона. В случае если описание $\mathbf{x}^{(k)}$, поданное на шаге k классифицируется неправильно, то происходит коррекция по правилу $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)} + \mathbf{x}^{(k)}$. В случае правильной классификации $\mathbf{w}^{(k+1)} = \mathbf{w}^{(k)}$.

Отметим, что правильной классификации всегда соответствует выполнение равенства $(\mathbf{w}^{(k)}, \mathbf{x}^{(k)}) \geq 0$ а неправильной классификации соответствует выполнение равенства $(\mathbf{w}^{(k)}, \mathbf{x}^{(k)}) < 0$. Процедура повторяется до тех пор, пока не будет выполнено одно из следующих условий:

- достигается полное разделение объектов из классов K_1 и K_2;

- повторение подряд заранее заданного числа итераций не приводит к улучшению разделения;

- оказывается исчерпанным заранее заданный лимит итераций. Для описанной процедуры справедлива следующая теорема.

Теорема. В случае, если описания объектов обучающей выборки линейно разделимы в пространстве признаковых описаний, то процедура обучения перцептрона построит линейную гиперплоскость разделяющую объекты двух классов за конечное число шагов.

Отсутствие линейной разделимости двух классов приводит к бесконечному зацикливанию процедуры обучения перцептрона.

Существенно более высокой аппроксимирующей способностью обладают нейросетевые методы распознавания, задаваемые комбинациями является связанных между собой нейронов. Таким методом является многослойный перцептрон.

4.2.2 Многослойный перцептрон.

В методе многослойный перцептрон сеть формируется из нескольких слоёв нейронов.

В их число входит слой входных рецепторов, подающих сигналы на нейроны из внутренних слоёв. Слои внутренних нейронов осуществляют преобразование сигналов. Слой реагирующих нейронов производит окончательную классификацию объектов на основании сигналов, поступающих от нейронов, принадлежащих внутренним слоям. Обычно соблюдаются следующие правила формирования структуры сети.

Допускаются связи между только между нейронами, находящимися в соседних слоях.

Связи между нейронами внутри одного слоя отсутствуют.

Активационные функции для всех внутренних нейронов идентичны и задаются сигмоидными функциями.

Для решения задач распознавания с L классами $K_1,\ldots,K_L$ используется конфигурация с L реагирующими нейронами. Схема многослойного перцептрона с двумя внутренними слоями представлена на рисунке 3.

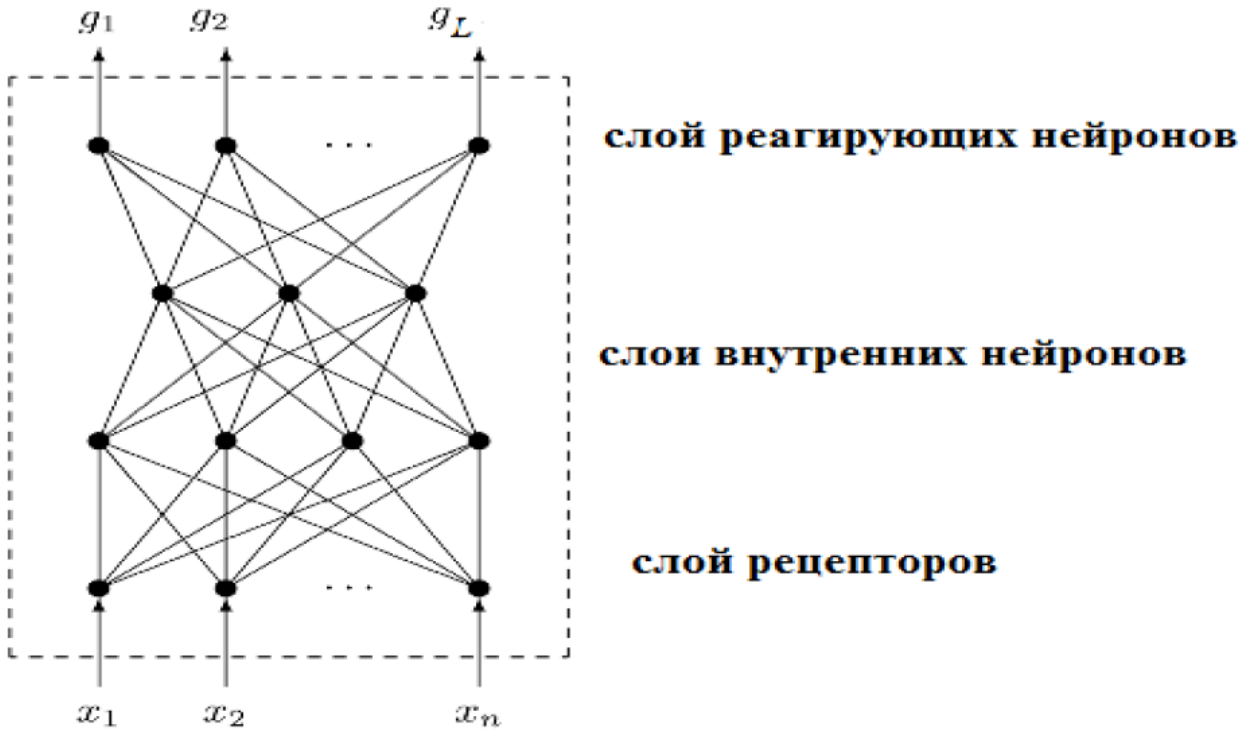

Рис. 3 Схема многослойного перцептрона с двумя внутренними слоями.

Отметим, что сигналы $g_1,\ldots,g_L$, вычисляемые на выходе реагирующих нейронов, интерпретируются как оценки за классы $K_1,\ldots,K_L$. Весовые коэффициенты w сопоставлены каждой из связей между нейронами из различных слоёв. Рассмотрим процедуру распознавания объектов с использованием многослойного перцептрона. Предположим, что конфигурация нейронной сети включает наряду со слоем рецепторов и слоем реагирующих нейронов также H внутренних слоёв искусственных нейронов. Заданы также количества нейронов в каждом слое. Пусть n – число входных нейронов-рецепторов, $r(h)$- число нейронов в внутреннем слое h.

На первом этапе вектор рецепторы формируют по информации, поступающей из внешней среды, вектор входных переменных (сигналов) $u_1^0,\ldots,u_n^0$. Отметим, что входные сигналы $u_1^0,\ldots,u_n^0$ могут интерпретироваться как признаки $X_1,\ldots,X_n$ в общей постановке задачи распознавания.

Предположим, что для нейрона с номером i из первого внутреннего слоя связь с рецепторами осуществляется с помощью весовых коэффициентов $w_1^{i0},\ldots,w_n^{i0}$. Сумматор нейрона i первого внутреннего слоя вычисляет взвешенную сумму $\xi^{i0}=\sum_{t=0}^{n} w_t^{i0}u_t^0$.

Сигнал на выходе нейрона i первого внутреннего слоя вычисляется по формуле $u_i^1=\Phi(\xi^{i0})$. Аналогичным образом вычисляются сигналы на выходе нейронов второго внутреннего слоя. Сигналы $g_1,\ldots,g_L$ рассчитываются с помощью той же самой процедуры, которая используется при вычислении сигналов на выходе нейронов из внутренних слоёв. То есть при вычислении g_i на первом шаге соответствующий сумматор вычисляет взвешенную сумму

$$\xi^{iH}=\sum_{t=0}^{r(H)} w_t^{iH}u_t^H,$$

где $w_1^{iH},\ldots,w_{r(h)}^{iH}$ - весовые коэффициенты, характеризующие связь реагирующего нейрона i с нейронами последнего внутреннего слоя H, $u_1^H,\ldots,u_{r(H)}^H$ - сигналы на выходе внутреннего слоя H. Сигнал на выходе реагирующего нейрона i

вычисляется по формуле $g_i = \Phi(\xi^{iH})$. Очевидно, что вектор выходных сигналов является функцией вектора входных сигналов (вектора признаков) и матрицы весовых коэффициентов связей между нейронами.

Аппроксимирующие способности многослойных перцептронов. Один реагирующий нейрон позволяет аппроксимировать области, являющиеся полупространствами, ограниченными гиперплоскостями. Нейронная сеть с одним внутренним слоем позволяет аппроксимировать произвольную выпуклую область в многомерном признаковом пространстве (открытую или закрытую).

Было доказано также, что МП с двумя внутренними слоями позволяет аппроксимировать произвольные области многомерного признакового пространства. Аппроксимирующая способность способность многослойного перцептрона с различным числом внутренних слоёв проиллюстрирована на рисунке 3.

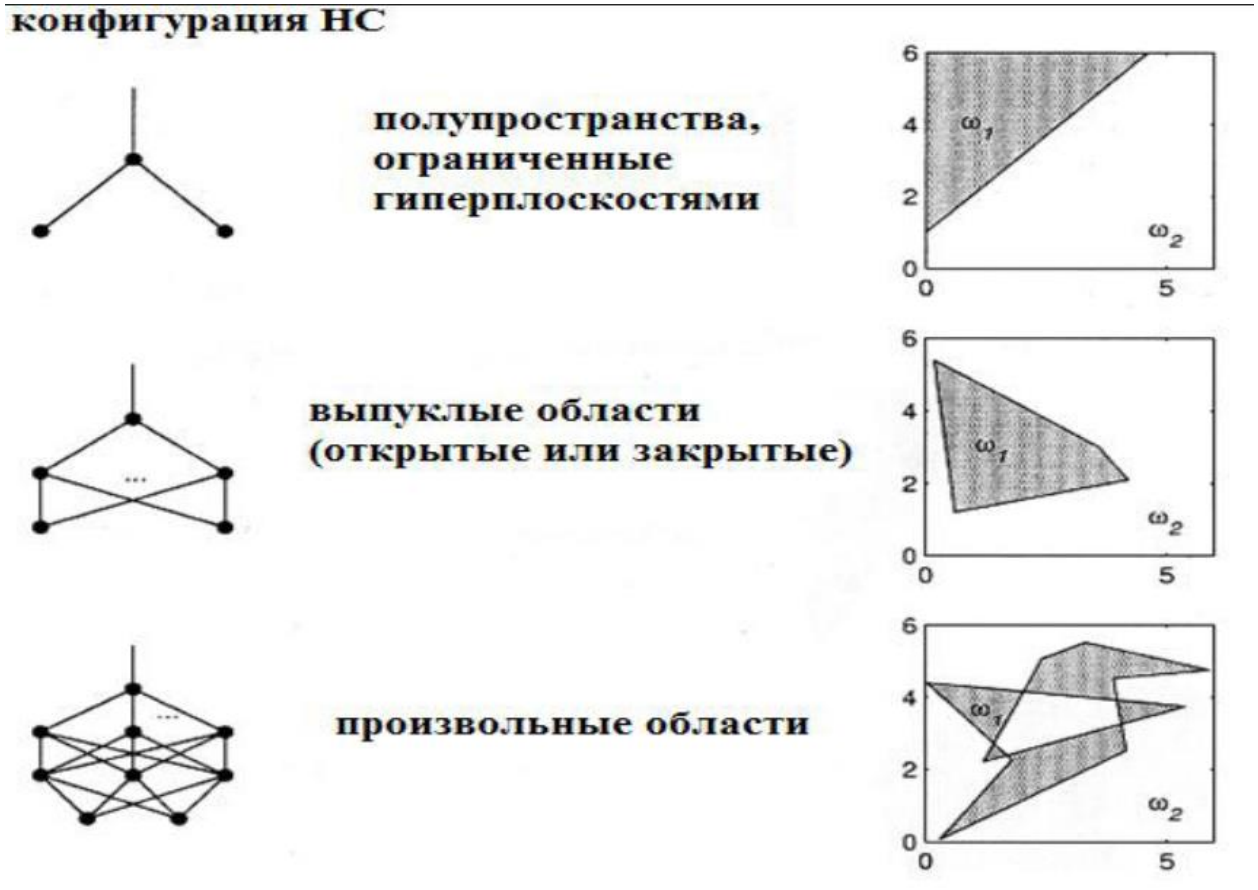

Рис. 3 На рисунке проиллюстрирована аппроксимирующая способность нейронных сетей. с различным числом внутренних слоёв.

Области, соответствующие классам ω_1 и ω_2 разделяются с помощью простого нейрона, а также с помощью многослойных перцептронов с одним и двумя внутренними слоями**.**

Верхняя конфигурация иллюстрирует разделяющую способность отдельного искусственного нейрон, функционирующего в соответствии с моделью Розенблатта.

Ниже представлена конфигурация с одним внутренним слоем нейронов. Данная конфигурация позволяет выделять в многомерном пространстве признаков выпуклые области произвольного типа. Наконец, в нижней части рисунка иллюстрируется разделяющая способность многослойного перцептрона с двумя внутренним слоями. Данная конфигурация позволяет выделять в многомерном пространстве признаков области, которые могут быть получены из набора выпуклых областей с помощью операций объединеия и пересечения. Очевидно, что многослойный перцептрон обладает очень высокой аппроксимирующей способностью.

Обучение многослойных перцептронов. Для обучения метода многослойный перцептрон обычно используется метод обратного распространения ошибки. Данный метод сходен с обучением перцептрона Розенблатта тем, что коррекция изначально произвольных значений весовых коэффициентов ω производится для каждого предъявленного в процессе обучения объекта. Коррекция производится с использованием метода градиентного спуска. То есть коррекция производится в направлении в пространстве коэффициентов ω, в котором максимально снижается целевой функционал. В качестве целевого функционала используется функционал эмпирического риска с квадратичными потерями. Принимается эффективный метод расчёта градиента, основанный на использовании аналитических формул.

4.3 Решающие деревья и леса

4.3.1 Решающие деревья

Структура решающих деревьев. Решающие деревья воспроизводят логические схемы, позволяющие получить окончательное решение о классификации объекта с помощью ответов на иерархически организованную систему вопросов. Причём вопрос, задаваемый на последующем иерархическом уровне, зависит от ответа, полученного на предыдущем уровне. Подобные логические модели издавна используются в ботанике, зоологии, минералогии, медицине и других областях. Пример, решающего дерева, позволяющая грубо оценить стоимость квадратного метра жилья в предполагаемом городе приведена на рисунке 4.

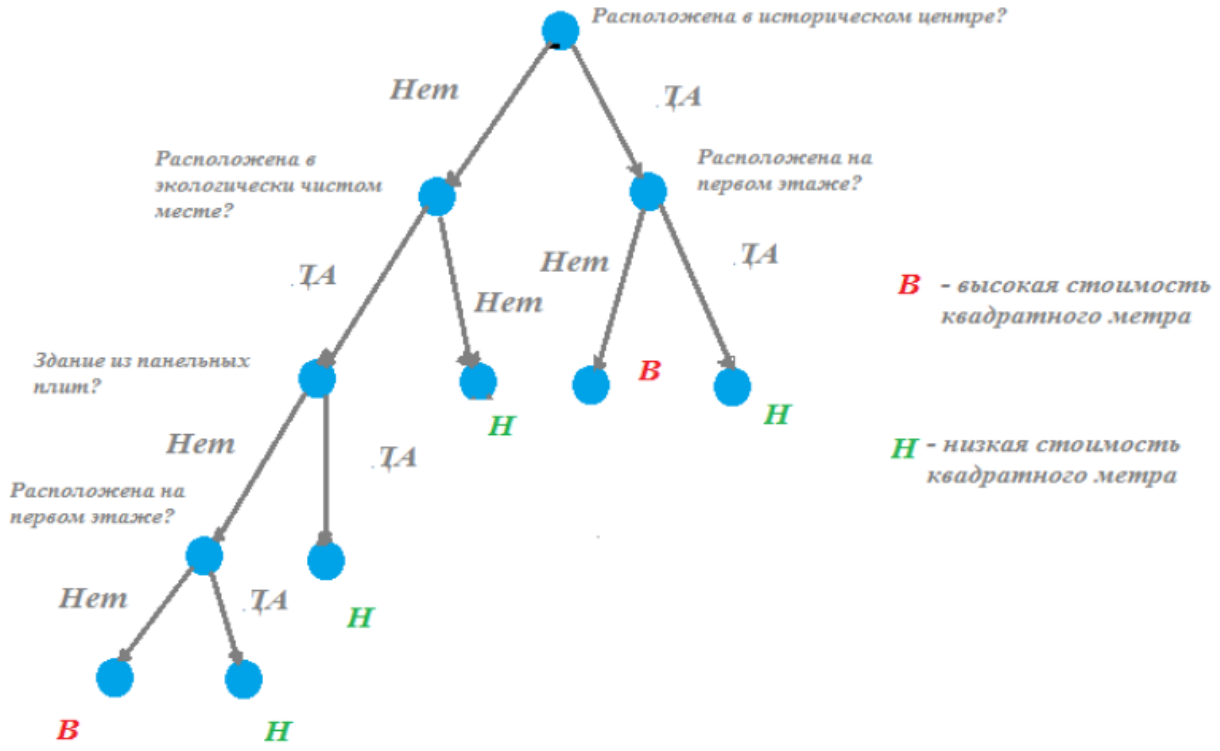

Рис. 4. Изображена структура решающего дерева, оценивающего стоимость квадратного метра жилых помещений. Для простоты выделяются два уровня стоимости – высокий и низкий.

Схеме принятия решений, изображённой на рисунке 1, соответствует связный ориентированный ациклический граф – ориентированное дерево. Дерево включает в себя корневую вершину, инцидентную только выходящим рёбрами, внутренние вершины, инцидентную одному входящему ребру и нескольким выходящим, и листья – концевые

Каждой из вершин дерева за исключением листьев соответствует некоторый вопрос, подразумевающий несколько вариантов ответов, соответствующих выходящим рёбрам. В зависимости от выбранного варианта ответа осуществляется переход к вершине следующего уровня. Концевым вершинам поставлены в соответствие метки, указывающие на отнесение распознаваемого объекта к одному из классов. Решающее дерево называется бинарным, если каждая внутренняя или корневая вершина инцидентна только двум выходящим рёбрам. Бинарные деревья удобно использовать в моделях машинного обучения.

Распознавание с помощью решающих деревьев. Предположим, что бинарное дерево $\mathbf{T}$ используется для распознавания объектов, описываемых набором признаков $X_1,\ldots,X_n$. Каждой вершине ν дерева $\mathbf{T}$ ставится в соответствие предикат, касающийся значения одного из признаков. Непрерывному признаку X_j соответствует предикат вида $"X_j \geq \delta_j^\nu"$, где δ_j^ν - некоторый пороговый параметр. Выбор одного из двух, выходящих из вершины ν рёбер производится в зависимости от значения предиката. Категориальному признаку X_j, принимающему значения из множества $M_{j'} = \{a_1^{j'},\ldots,a_{r(j')}^{j'}\}$ ставится в соответствие предикат вида $"X_{j'} \in M_{j'}^{\nu 1}"$, где $M_{j'}^{\nu 1}$ является элементом дихотомического разбиения $\{M_{j'}^{\nu 1}, M_{j'}^{\nu 2}\}$ множества $M_{j'}$. Выбор одного из двух, выходящих из вершины ν рёбер производится в зависимости от значения предиката. Процесс распознавания заканчивается при достижении концевой вершины (листа). Объект относится классу согласно метке, поставленной в соответствие данному листу.

Обучение решающих деревьев. Рассмотрим задачу распознавания с классами $K_1,\ldots,K_L$. Обучение алгоритма решающее дерево производится по обучающей выборке $\tilde{S}_t$ и включает в себя поиск оптимальных пороговых параметров или оптимальных дихотомических разбиений для признаков $X_1,\ldots,X_n$. При этом поиск производится исходя из требования снижения среднего индекса неоднородности в выборках, порождаемых искомым дихотомическим разбиением обучающей выборки $\tilde{S}_t$.

Индексы неоднородности вычисляется для произвольной выборки $\tilde{S}$, содержащей объекты из классов $K_1,\ldots,K_L$.

При этом используется несколько видов индексов, включая:

- энтропийный индекс неоднородности,

- индекс Джини,

- индекс ошибочной классификации.

Энтропийный индекс неоднородности вычисляется по формуле

$$\gamma_e(\tilde{S}) = -\sum_{i=1}^{L} P_i \ln(P_i),$$

где P_i - доля объектов класса в выборке $\tilde{S}$. При этом принимается, что $0\ln(0) = 0$. Наибольшее значение $\gamma_e(\tilde{S})$ принимает при равенстве долей классов. Наименьшее значение $\gamma_e(\tilde{S})$ достигается при принадлежности всех объектов одному классу. Индекс Джини вычисляется по формуле

$$\gamma_g(\tilde{S}) = 1 - \sum_{i=1}^{L} P_i^2 .$$

Индекс ошибочной классификации вычисляется по формуле

$$\gamma_m(\tilde{S}) = 1 - \max_{i\in\{1,..,L\}}(P_i).$$

Нетрудно понять, что индексы (2) и (3) также достигают минимального значения при принадлежности всех объектов обучающей выборке одному классу. Предположим, что в методе обучения используется индекс неоднородности $\gamma_*(\tilde{S})$. Для оценки эффективности разбиения обучающей выборки $\tilde{S}_t$ на непересекающиеся подвыборки $\tilde{S}_t^l$ и $\tilde{S}_t^r$ используется уменьшение среднего индекса неоднородности в $\tilde{S}_t^l$ и $\tilde{S}_t^r$ по отношению к $\tilde{S}_t$. Данное уменьшение вычисляется по формуле

$$\Delta(\gamma_*, \tilde{S}_t) = \gamma_*(\tilde{S}_t) - P_l\gamma_*(\tilde{S}_t^l) - P_r\gamma_*(\tilde{S}_t^r),$$

где P_l и P_r являются долями $\tilde{S}_t^l$ и $\tilde{S}_t^r$ в полной обучающей выборке $\tilde{S}_t$.

На первом этапе обучения бинарного решающего дерева ищется оптимальный предикат соответствующий корневой вершине. С этой целью оптимальные разбиения строятся для каждого из признаков из набора $X_1,\ldots,X_n$. Выбирается признак $X_{i_{\max}}$ с максимальным значением индекса $\Delta(\gamma_*,\tilde{S}_t)$. Подвыбороки $\tilde{S}_t^l$ и $\tilde{S}_t^r$, задаваемые оптимальным предикатом для $X_{i_{\max}}$ оцениваются с помощью критерия остановки. В качестве критерия остановки может быть использован простейший критерий достижения полной однородности по одному из классов. В случае, если какая-нибудь из выборок $\tilde{S}_t^*$ удовлетворяет критерию остановки, то соответствующая вершина дерева объявляется концевой и для неё вычисляется метка класса. В случае, если выборка $\tilde{S}_t^*$ не удовлетворяет критерию остановки, то формируется новая внутренняя вершина, для которой процесс построения дерева продолжается. Однако вместо обучающей выборки $\tilde{S}_t$ используется соответствующая вновь образованной внутренней вершине v выборка $\tilde{S}_v$, которая равна $\tilde{S}_t^*$. Для данной выборки производятся те же самые построения, которые на начальном этапе проводились для обучающей выборки $\tilde{S}_t$. Обучение может проводиться до тех пор, пока все вновь построенные вершины не окажутся однородными по классам. Такое дерево может быть построено всегда, когда обучающая выборка не содержит объектов с одним и тем же значениям каждого из признаков, принадлежащих разным классам. Однако абсолютная точность на обучающей выборке не всегда приводить к высокой обобщающей способности в результате эффекта переобучения.

Одним из способов достижения более высокой обобщающей способности является использования критериев остановки, позволяющих остановит процесс построения дерева до того, как будет достигнута полная однородность концевых вершин.

Рассмотри несколько таких критериев.

1. *Критерий остановки по минимальному допустимому числу объектов в выборках, соответствующих концевым вершинам.*

2. *Критерий остановки по минимально допустимой величине индекса* $\Delta(\gamma_*,\tilde{S}_t)$. Предположим, что некоторой вершине v соответствует выборка $\tilde{S}_v$, для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\tilde{S}_v^l,\tilde{S}_v^r\}$. Вершина v считается внутренней, если индекс $\Delta(\gamma_*,\tilde{S}_t)$ превысил пороговое значение τ и считается концевой в противном случае.

3. Критерий остановки по точности на контрольной выборке. Исходная выборка данных

случайным образом разбивается на обучающую выборку $\tilde{S}_t$ и контрольную выборку $\tilde{S}_c$. Выборка $\tilde{S}_t$ используется для построения бинарного решающего дерева. Предположим, что некоторой вершине v соответствует выборка $\tilde{S}_v$, для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\tilde{S}_v^l, \tilde{S}_v^r\}$.

На контрольной выборке $\tilde{S}_c$ производится сравнение эффективность распознающей способности деревьев $\mathbf{T}_v$ и $\mathbf{T}_v^{++}$.

Деревья $\mathbf{T}_v$ и $\mathbf{T}_v^{++}$ включает все вершины и рёбра, построенные до построения вершины v. В дереве $\mathbf{T}_v$ вершина v считается концевой. В дереве $\mathbf{T}_v^{++}$ вершина v считается внутренней, а концевыми считаются вершины, соответствующие подвыборкам $\tilde{S}_v^l$ и $\tilde{S}_v^r$.

Распознающая способность деревьев $\mathbf{T}_v$ и $\mathbf{T}_v^{++}$ сравнивается на контрольной выборке $\tilde{S}_c$. В том, случае если распознающая способность $\mathbf{T}_v^{++}$ превосходит распознающую способность $\mathbf{T}_v$ все дальнейшие построения исходят из того, что вершина v является концевой. В противном случае производится исследование $\tilde{S}_v^l$ и $\tilde{S}_v^r$.

4.Статистический критерий. Заранее фиксируется пороговый уровень значимости (P<0.05,p<0.01 или p<0.001). Предположим, что нам требуется оценить, является ли концевой вершина, для которой найдены оптимальный признак вместе с оптимальным предикатом, задающим разбиение $\{\tilde{S}_v^l, \tilde{S}_v^r\}$. Исследуется статистическая достоверность различий между содержанием объектов распознаваемых классов в подвыборках $\tilde{S}_v^l$ и $\tilde{S}_v^r$. Для этих целей может быть использованы известные статистический критерий: Хи-квадрат и другие критерии. По выборкам $\tilde{S}_v^l$ и $\tilde{S}_v^r$ рассчитывается статистика критерия и устанавливается соответствующее p-значение. В том случае, если полученное p-значение оказывается меньше заранее фиксированного уровня значимости вершина v считается внутренней. В противном случае вершина v считается концевой.

Использование критериев ранней остановки не всегда позволяет адекватно оценить необходимую глубину дерева. Слишком ранняя остановка ветвления может привести к потере информативных предикатов, которые могут быть на самом деле найдены только при достаточно большой глубине ветвления.

В связи с этим нередко целесообразным оказывается построение сначала полного дерева, которое затем уменьшается до оптимального с точки зрения достижения максимальной обучающей способности размера путём объединения некоторых концевых вершин. Такой процесс в литературе принято называть «pruning» («подрезка»).\\

При подрезке дерева может быть использован критерий целесообразности объединения двух вершин, основанный на сравнение на контрольной выборке точности распознавания до и после проведения «подрезки».

Ещё один способ оптимизации обобщающей способности деревьев основан на учёте при «подрезке» дерева до некоторой внутренней вершины v одновременно увеличения точности разделения классов на обучающей выборке и увеличения сложности, которые возникают благодаря ветвлению из v.

При этом прирост сложности, связанный с ветвлением из вершины v, может быть оценён через число листьев в поддереве $\mathbf{T}_v^{sub}$ полного решающего дерева с корневой вершиной v. Следует отметить, что рост сложности является штрафующим фактором, компенсирующим прирост точности разделения на обучающей выборке с помощью включения поддерева $\mathbf{T}_v^{sub}$ в решающее дерево. Разработан целый ряд эвристических критериев, которые позволяют оценить целесообразность включения $\mathbf{T}_v^{sub}$. Данные критерии учитывают одновременно сложность и разделяющую способность.

4.3.2 Решающие леса

В результате многочисленных экспериментов было установлено, что точность нередко значительно возрастает, если вместо отдельных решающих деревьев использовать коллективы (ансамбли) решающих деревьев, которые принято называть решающими лесами. Коллективное решение вычисляется по результатам распознавания отдельными членами ансамбля. В методах решающих лесов в качестве членов ансамблей принято использовать решающих деревьев, которые строятся по искусственно сгенерированным обучающим выборок, статистически сходных с исходной обучающей выборкой.

Получили распространение процедуры построения решающих лесов «бэггинг» и «бустинг»., основанные на различных способах генерации «искусственных» выборок из исходной обучающей выборки.

В методе «бэггинг» (bagging) каждая искусственная случайная выборка является выборкой с возвращениями из исходной обучающей выборки $\tilde{S}_t = \{s_1, \ldots, s_m\}$, также содержащей m объектов. Подобный способ генерации выборок

называют методом «бутрэп» (bootstrap). Название bagging является сокращённым и происходит от полного названия «бутстрэп агрегирование» (Bootstrap Aggregating). Отметим, что искусственная выборка состоит только из объектов исходной обучающей выборки $\tilde{S}_t$. Однако некоторые объекты $\tilde{S}_t$ могут встречаться искусственной выборке по нескольку раз, а некоторые могут вообще отсутствовать.

Для построения коллективного решения может быть использован простейшее решающее правило голосования по большинству: объект относится к тому классу, в который его отнесло большинство деревьев, формирующих лес.

Основной идеей метода бустинг (boosting) является пошаговое наращивание ансамбля деревьев. При этом на каждом шаге к ансамблю присоединяется алгоритм, который был обучен по выборке, искусственно сгенерированной из исходной обучающей выборки $\tilde{S}_t$. В отличие от метода «бэггинг», простая выборка с возвращениями, предполагающая равновероятность всех объектов $\tilde{S}_t$, используется для обучения только на первом шаге. На каждом последующем шаге k объекты в искусственные выборки выбираются с учётом вероятностей, приписанных объектам исходной выборки $\tilde{S}_t$. Последнее распределение вероятностей вычисляется с учётом результатов классификации с помощью ансамбля, использованного на предыдущем шаге. При этом объектам, которые на предыдущем шаге были классифицированы неверно приписываются более высокие веса.

Существуют различные варианты реализации схемы «бустинг», зависящие от способа вычисления вероятностей, приписываемых объектам $\tilde{S}_t$. а также способов вычисления коллективного решения. Одной из наиболее известных вариантов метода «бустинг» является метод Adaptive Boosting (AdaBoost).

4.4 Комбинаторно-логические методы, основанные на принципе частичной прецедентности

Многие прикладные задачи распознавания могут быть успешно решены с помощью методов, основанных на принципе частичной прецедентности. Данный принцип подразумевает поиск по обучающей выборке фрагментов описаний, позволяющих с разной степенью точности разделить распознаваемые классы $K_1,\ldots,K_L$. Распознаваемый объект оценивается по совокупности найденных фрагментов. Одной из первых реализаций принципа частичной прецедентности является тестовый алгоритм, предложенный в 1966 году. Данный алгоритм основан на понятии тупикового теста. Исходный вариант тестового алгоритма предназначен для распознавания объектов, описываемых с помощью бинарных или категориальных признаков $X_1,\ldots,X_n$. Иными словами $X_i \in \{a_1^i,\ldots,a_{k(i)}^i\}$, $i=1,\ldots,n$. Пусть обучающая выборка $\tilde{S}_t$ содержит объекты из классов $K_1,\ldots,K_L$. При этом общее число объектов равно m.

Выборке $\tilde{S}_t$ ставится в соответствие таблица $\mathbf{T}_{nml}$. В строке j таблицы $\mathbf{T}_{nml}$ находятся значения признаков $X_1,\ldots,X_n$ на объекте s_j.

Определение 1. Тестом таблицы $\mathbf{T}_{nml}$ называется такая совокупность столбцов $\{i_1,\ldots,i_r\}$, что для произвольной пары строк s' и s'', соответствующих объектам из разных классов, существует такой столбец i^* из множества $\{i_1,\ldots,i_r\}$, что значения на пересечении i^* со строками s' и s'' различны.

Иными словами набор признаков считается тестом, если описания любых двух объектов из разных классов отличаются хотя бы по одному из признаков, входящих в тест.

Определение 2. Тест $T=\{i_1,\ldots,i_r\}$ называется тупиковым, если никакое его отличное от T подмножество (собственное подмножество) тестом не является

На этапе обучения ищется множество всевозможных тупиковых тестов $\tilde{T}(\tilde{S}_t)$ для таблицы $\mathbf{T}_{nml}$. Предположим что нам требуется распознать объект s_* с векторным описанием $(x_{*1},\ldots,x_{*n})$. Выделим в векторном описании фрагмент

$(x_{i_1},\ldots,x_{i_r})$, соответствующий тесту T из множества $\tilde{T}(\tilde{S}_t)$. Фрагмент $(x_{i_1},\ldots,x_{i_r})$ сравнивается с множеством фрагментов строк $(x_{ji_1}^T,\ldots,x_{ji_r}^T)$ таблицы $\mathbf{T}_{nml}$,

соответствующих классу K_l : $\{(x_{ji_1}^T,\ldots,x_{ji_r}^T)\,|\,s_j \in K_l\}$ $(x_{i_1},\ldots,x_{i_r})$. В

случаях, когда выполняются равенства $x_{ji_1}^T = x_{*i_1},\ldots,x_{ji_r}^T = x_{*i_r}$

фиксируем полное совпадение. Обозначим число полных совпадений распознаваемого объекта s_* с объектами K_l из $\tilde{S}_t$ через $G_l(T,s_*)$.

Оценка объекта s_* за класс K_l вычисляется по формуле:

$$\gamma_l(s_*) = \frac{1}{m_l}\sum_{T\in\tilde{T}(\tilde{S}_t)} G_l(T,s_*) \ ,$$

где m_l - число объектов обучающей выборки из класса K_l . Классификация объекта может производится с помощью по вектору оценок $[\gamma_1(s_*),\ldots,\gamma_L(s_*)]$ с помощью стандартного решающего правила, т.е. объект относится в тот класс, оценка за который максимальна

Задача о поиске всевозможных тупиковых тестов сводится к известной задаче комбинаторного анализа о поиске всевозможных тупиковых покрытий элементам.

Нахождение всех тупиковых тестов является сложной комбинаторной задачей. Однако эффективные алгоритмы поиска разработаны для некоторых типов таблиц. При решении практических задач эффективен подход , основанный на вычислении только части тупиковых тестов.

Другим известным классом алгоритмов распознавания , основанным на принципе частичной прецедентности, являются алгоритмы типа КОРА. В отличие от тестового алгоритма, где в качестве информативных элементов используются несжимаемые наборы признаков – тупиковые тесты, в алгоритмах типа КОРА в качестве информативных элементов используются несжимаемые фрагменты описаний эталонных объектов обучающей выборки.

Определение 3.Пусть $(x_{v1},\ldots,x_{vn})$ - признаковое описание объекта $s_v \in K_l$. Набор $(x_{vj_1},\ldots,x_{vj_r})$ называется представительным набором для класса K_l , если для произвольной строки K_l таблицы $\mathbf{T}_{nml}$ соответствующей объекту

$s_u \notin K_i$ такое, что существует такое j' из множества $\{j_1,\ldots,j_r\}$, что $x_{vj'} \neq x_{uj'}$.

Определение 4. Представительный набор называется тупиковым, если никакое его собственное подмножество представительным набором не является.

На этапе обучения для каждого из классов $K_1,\ldots,K_L$ по таблице $\mathbf{T}_{nml}$ ищется множество всевозможных тупиковых представительных наборов. Обозначим через $\tilde{V}_l$ - множество всевозможных представительных наборов для класса K_l. Предположим, что нам требуется распознать объект s_* с описанием $(x_{*1},\ldots,x_{*n})$. Пусть $v=(x_{ui_1},\ldots,x_{ui_r})$ - представительный набор. Функция $\mathrm{B}(s_*,v)$ равна 1, если $(x_{*i_1}=x_{vi_1},\ldots,x_{*i_r}=x_{vi_r})$, и $\mathrm{B}(s_*,v)$ равна 0 в противном случае.

Оценка s_* за класс K_l вычисляется по формуле

$$\Gamma_l(s_*)=\frac{1}{|\tilde{V}_i|}\sum_{u\in\tilde{V}_i}\mathrm{B}(s_*,u)\ .$$

Первоначальные варианты тестового алгоритма и алгоритма типа КОРА были разработаны для бинарных или категориальных переменных. Они не могут быть напрямую использованы в задачах с признаками, принимающими значения из интервалов вещественной оси. Для того, чтобы обеспечить возможность работы с подобной информацией могут быть использованы два подхода**.**

а) Первый подход основан на разбиении области возможных значений каждого вещественнозначного признака на k связных подмножеств (интервалов, полуинтервалов, отрезков). Значению признака, принадлежащего элементу j разбиения присваивается само значение j. Разбиение оптимизируется с целью достижения максимального разделения классов. Выбирается такое число элементов разбиения k, при котором достигается максимальная точность распознавания.

Другой подход основан на модификации понятий теста и представительного набора с использованием пороговых параметров $\varepsilon_1,\ldots,\varepsilon_n$, которые задаются для признаков $X_1,\ldots,X_n$.

Определение 5. Тестом таблицы $\mathbf{T}_{nml}$ называется такая совокупность столбцов $\{i_1,\ldots,i_r\}$, что для произвольной пары строк s' и s'', соответствующих объектам из разных классов, существует такой столбец i^* из множества $\{i_1,\ldots,i_r\}$, что абсолютная

величина разницы значений, стоящих на пересечении i^* со строками s' и s'' превышает ε_{i^*} .

Аналогичным образом вводится модифицированное определение представительного набора**.**

Главным требованием при выборе ε - порогов является достижение максимальной отделимости объектов разных классов при сохранении сходства внутри классов.

Поиск тупиковых тестов и тупиковых представительных наборов при модифицированных определениях аналогичен их поиску в первоначальных вариантах методов.

Тестовый алгоритм и алгоритм с представительными наборами являются частью более общей конструкции, основанной на принципе частичной прецедентности и носящей название алгоритмов вычисления оценок.

Существует много вариантов моделей данного типа. Причём конкретный вид модели определяется выбранными способами задания различных её элементов. Рассмотрим основные составляющие модели

Задание системы опорных множеств. Под опорными множествами модели АВО понимается наборы признаков, по которым осуществляется сравнение распознаваемых и эталонных объектов. Примером системы опорных множеств является множество тупиковых тестов. Система опорных множеств Ω_A некоторого алгоритма A может задаваться через систему подмножеств множества $\{1,\ldots,n\}$ или через систему характеристических бинарных векторов.

Каждому подмножеству $\{1,\ldots,n\}$ может быть сопоставлен бинарный вектор размерности n. Пусть $\{i_1,\ldots,i_k\} \subseteq \{1,\ldots,n\}$. Тогда $\{i_1,\ldots,i_k\}$ сопоставляется вектор $\boldsymbol{\omega} = \{\omega_1,\ldots,\omega_n\}$, все компоненты которого равны 0 кроме равных 1 компонент $\omega_{i_1},\ldots,\omega_{i_k}$. Теоретические исследования свойств тупиковых тестов для случайных бинарных таблиц показали, что характеристические векторы для почти всех тупиковых тестов имеют асимптотически (при неограниченном возрастании размерности таблицы обучения) одну и ту же длину.

Данный результат является обоснованием выбора в качестве системы опорных векторов всевозможных наборов, включающих фиксированное число признаков k или

$\Omega_A = \{\boldsymbol{\omega} : |\boldsymbol{\omega}| = k\}$. Оптимальное значение k находится в процессе обучения или задаётся экспертом. Другой часто используемой системе опорных множеств соответствует множество всех подмножеств $\{1,\ldots,n\}$ за исключением пустого множества.

Задание функции близости. Близость между объектами по опорноным множествам задаётся аналогично тому, как задаётся близость между объектами по тупиковым тестам или представительным наборам. Пусть набор номеров $\{i_1,\ldots,i_k\}$ соответствует характеристическому вектору $\boldsymbol{\omega}$.

Фрагмент $(x_{\mu i_1},\ldots,x_{\mu i_k})$ описания $(x_{\mu 1},\ldots,x_{\mu n})$ объекта $\mathbf{x}_\mu$ называется $\boldsymbol{\omega}$ - частью объекта s_μ. Под функцией близости $\mathbf{B}_{\boldsymbol{\omega}}(s_\mu, s_\upsilon)$ понимается функция от соответствующих $\boldsymbol{\omega}$ -частей сравниваемых объектов, принимающая значения 1 (объекты близки) или 0 (объекты удалены).

Функции близости обычно задаются с помощью пороговых параметров $(\varepsilon_1,\ldots,\varepsilon_n)$, характеризующих близость объектов по отдельным признакам.

Примеры функций близости.

1) $\mathbf{B}_{\boldsymbol{\omega}}(s_\mu, s_\upsilon) = 1$, если при произвольном $i \in \{1,\ldots,n\}$, при котором $\omega_i = 1$, всегда выполняется неравенство $|x_{\mu i} - x_{\nu i}| < \varepsilon_i$. $\mathbf{B}_{\boldsymbol{\omega}}(s_\mu, s_\upsilon) = 0$, если существует такое $i' \in \{1,\ldots,n\}$, что одновременно $\omega_{i'} = 1$ и $|x_{\mu i'} - x_{\nu i'}| > \varepsilon_{i'}$.

2) Пусть ε - скалярный пороговый параметр. Функция $\mathbf{B}_{\boldsymbol{\omega}}(s_\mu, s_\upsilon) = 1$, если выполняется неравенство $\sum_{i=1}^{n} \omega_i |x_{\mu i} - x_{\nu i}| < \varepsilon$. В противном случае $\mathbf{B}_{\boldsymbol{\omega}}(s_\mu, s_\upsilon) = 0$.

Важным элементом АВО является оценка близости распознаваемого объекта s_* к эталону s_μ по заданной ω- части. Данная оценка близости, которая будет обозначаться $G(\boldsymbol{\omega}, s_*, s_\mu)$, формируется на основе введённых ранее функций близости и, возможно, дополнительных параметров. Приведём примеры функций близости различного уровня сложности:

А) $G(\boldsymbol{\omega}, s_*, s_\mu) = \mathrm{B}_{\boldsymbol{\omega}}(s_*, s_\mu)$,

B) $G(\boldsymbol{\omega}, s_*, s_\mu) = p_{\boldsymbol{\omega}} \mathrm{B}_{\boldsymbol{\omega}}(s_*, s_\mu)$, где $p_{\boldsymbol{\omega}}$ - параметр, характеризующий информативность опорного множества с характеристическим вектором $\boldsymbol{\omega}$.

C) $G(\boldsymbol{\omega}, s_*, s_\mu) = \gamma_\mu (\sum_{i=1}^{n} p_i \omega_i) \mathrm{B}_{\boldsymbol{\omega}}(s_*, s_\mu)$, где γ_μ - параметр, характеризующий информативность эталона s_μ, параметры $p_1, \ldots, p_n$ характеризуют информативность отдельных признаков.

Оценка объекта s_* за класс K_l при фиксированном характеристическом векторе $\boldsymbol{\omega}$ может вычисляться как среднее значение близости s_* к эталонным объектам из класса K_l: $\Gamma_l(\boldsymbol{\omega}, s_*) = \frac{1}{m_l} \sum_{s_\mu \in K_l} G(\boldsymbol{\omega}, s_*, s_\mu)$, где m_l - число объектов обучающей выборки из класса K_l.

Общая оценка s_* за класс K_l вычисляется как сумма оценок $\Gamma_l(\boldsymbol{\omega}, s_*)$ по опорным множествам из системы Ω_A:

$$\hat{\boldsymbol{\Gamma}}_l(s_*) = \sum_{\boldsymbol{\omega} \in \Omega_A} \Gamma_l(\boldsymbol{\omega}, s_*) \qquad (1)$$

Наряду с формулой (1) используется формула

$$\hat{\boldsymbol{\Gamma}}_l(s_*) = w_l \sum_{\boldsymbol{\omega} \in \Omega_A} \Gamma_l(\boldsymbol{\omega}, s_*) \qquad (2)$$

Использование взвешивающих параметров $w_1, \ldots, w_L$ позволяет регулировать доли правильно распознанных объектов $K_1, \ldots, K_L$.

Прямое вычисление оценок за классы по формулам (1) и (2) в случаях, когда в качестве систем опорных множеств используются наборы с фиксированным числом признаков или всевозможные наборы признаков, оказывается практически невозможным при сколь либо высокой размерности признакового пространства из-за необходимости вычисления огромного числа значений функций близости.

Однако при равенстве весов всех признаков существуют эффективные формулы для вычисления оценок по формуле (1). Предположим, что оценки близости распознаваемого объекта s_* к эталону s_μ по заданной $\boldsymbol{\omega}$ - части вычисляются по формуле (A).

Тогда оценка по формуле (1) принимает вид $\hat{\boldsymbol{\Gamma}}_l(s_*) = \sum_{s_\mu \in K_l} \sum_{\boldsymbol{\omega} \in \Omega_A} \mathrm{B}_{\boldsymbol{\omega}}(s_*, s_\mu)$

Рассмотрим сумму $\sum_{\omega\in\Omega_A} B_\omega(s_*, s_\mu$. Предположим, что общее число признаков, по которым объект s_* близок объекту s_μ равно $d(s_*, s_\mu)$. Иными словами $d(s_*, s_\mu) = |D(s_*, s_\mu)|$, где $D(s_*, s_\mu) = \{i : | x_{*i} - x_{\mu i} | < \varepsilon_i\}$. Очевидно функция близости $B_\omega(s_*, s_\mu)$ равна 1 тогда и только тогда, когда опорное множество, задаваемое характеристическим вектором ω, полностью входит в множество $D(s_*, s_\mu)$. Во всех остальных случаях $B_\omega(s_*, s_\mu) = 0$.

Предположим, что система опорных множеств удовлетворяет условию $\Omega_A = \{\omega : |\omega| = k\}$. Очевидно, что число опорных множеств, удовлетворяющих условию $B_\omega(s_*, s_\mu) = 1$, равно $C^k_{d(s_*, s_\mu)}$. Откуда следует, что $\sum_{\omega\in\Omega_A} B_\omega(s_*, s_\mu) = C^k_{d(s_*, s_\mu)}$. Следовательно оценка по формуле (1) может быть записана в виде

$$\hat{\Gamma}_l(s_*) = \frac{1}{m_l} \sum_{s_\mu \in K_l} \gamma_\mu C^k_{d(s_*, s_\mu)} \ . \qquad (3)$$

Предположим, что система Ω_A включает в себя всевозможные опорные множества. В этом случае число опорных множеств в Ω_A, удовлетворяющих условию $B_\omega(s_*, s_\mu) = 1$ равно $2^{d(s_*, s_\mu)}$-1. Следовательно оценка по формуле (1) может быть записана в виде

$$\hat{\Gamma}_l(s_*) = \frac{1}{m_l} \sum_{s_\mu \in K_l} \gamma_\mu [2^{d(s_*, s_\mu)} - 1]$$

Обучение алгоритмов вычисления оценок. Для обучения алгоритмов АВО в общем случае может быть использован тот же самый подход, который используется для обучения в методе «Линейная машина». Предположим, что решается задача обучения алгоритмов

для распознавания объектов , принадлежащих классам $K_1,\ldots,K_L$. При правильного распознавания объекта $s_j \in K_l$ должна выполняться система неравенств

$$\tilde{\mathbf{\Gamma}}_l(s_l) > \tilde{\mathbf{\Gamma}}_{l'}(s_j), \text{ где } l' \in \{1,\ldots,L\} \setminus l .$$

В наиболее общем из приведённых выше вариантов модели АВО обучение может быть сведено к поиску максимальной совместной подсистемы системы неравенств

$$\forall s_j \in K_l \cap \tilde{S}_t \qquad (4)$$
$$\frac{1}{m_l}\sum_{S_\mu \in K_l} \gamma_\mu \left(\sum_{t=1}^{n} p_t \omega_t\right) \mathrm{B}_{\boldsymbol{\omega}}(s_j, s_\mu) > \frac{1}{m_{l'}}\sum_{S_\nu \in K_{l'}} \gamma_\nu \left(\sum_{t=1}^{n} p_t \omega_t\right) \mathrm{B}_{\boldsymbol{\omega}}(s_j, s_\nu)$$

Поиск максимальной совместной подсистемы системы (2) может производиться с использованием эвристического релаксационного метода, аналогичного тому, что был использован при обучении алгоритма «Линейная машина».

Тестовый алгоритм был впервые предложен в работе [], где для решении задачи геологического прогнозирования.