# Основные определения курса «Распознавание Образов»

Скрибцов П.В., 13.03.2005

Цели науки распознавания образов:

1. замена человеческого эксперта или сложной экспертной системы более простой системой (автоматизация деятельности человека или упрощение сложных систем).
2. построение обучающихся систем, которые умеют принимать решения без указания четких правил, а именно, систем, которые умеют сами синтезировать правила принятия решений на основе некоторого конечного количества «продемонстрированных» системе примеров правильных решений.

**Множество объектов задачи распознавания** – множество всех объектов, которые могут теоретически встретиться в конкретной задаче их распознавания.

**Образ** (также pattern, shape, объект) – любой объект, для которого можно измерить набор определенных числовых признаков. Пример образа: буква, изображение, кардиограмма, и т.п.

**Числовой** **признак** (или просто **признак**). Формула или иное описание способа сопоставления объекту некоторой числовой характеристики, которое действует в рамках конкретной задачи распознавания образов. Для каждого объекта может быть определено несколько различных *признаков*, то есть несколько числовых характеристик.

При этом считается, что если все числовые значения всех признаков двух объектов совпадают, то такие объекты считаются идентичными, несмотря на то, что по сути объекты могут различаться. Например, если в какой-то задаче распознавания образов для шаров определен один только признак – радиус, то при этом красный шар с радиусом 10 в данной постановке задачи будет считаться идентичным синему шару с радиусом 10, поскольку значения их единственного признака совпадают.

**Пространство признаков**. N-мерное пространство, определенное для данной задачи распознавание, где N – фиксированное число измеряемых признаков для любых объектов. Вектор из пространства признаков, соответствующий объекту задачи распознавания это N-мерный вектор с компонентами (х1,х2, …, хN), которые являются значениями признаков данного объекта.

ОБЪЕКТ->N признаков->N-мерный вектор признаков

**Класс**. *Неформализируемое (как правило) представление* о возможности отнесения произвольного объекта из множества объектов задачи распознавания к определенной группе объектов. Для объектов одного класса предполагается наличие «*схожести»*. Для задачи распознавания образов может быть определено произвольное количество классов, большее 1. Количество классов обозначается числом S.

**Гипотеза о схожести**: Если для объекта А, характеризуемого вектором признаков х1 и для объекта Б, характеризуемого вектором признаков х2 выполняется следующее:

|x1-x2| -> 0, то вероятность что объекты А и Б принадлежат одному и тому же классу тоже стремится к единице.

**Обучающая выборка**. Неформальное определение классов данной задачи распознавания при помощи примеров объектов, отношение которых к тому или иному классу заранее определено (известно). Математически обучающая выборка – это две сущности:

* последовательность N-мерных векторов x(k), где индекс k = 1..K пробегает по всем номерам примеров.
* последовательность чисел D\*(k), значения которых равны номеру класса для объекта, характеризуемого вектором х(k).

Обучающая выборка может быть представлена в виде таблицы вида

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | k | | | |
| I | 1 | 2 | … | K |
| i=1 | x11 | x21 |  | xK1 |
| i=2 | x21 | x22 |  | xK2 |
| .. | x31 | x23 |  | xK3 |
| i=N | xN1 | xN3 |  | xKN |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **D\*(k)** | **D\*(1)** | **D\*(2)** | **…** | **D\*(K)** |

т.е. по горизонтали идет итерирование по номеру объекта-примера, а по вертикали по признакам и значению номера класса.

Значения классов примеров D\*(1)…D\*(K) называют **указаниями учителя**.

Обучающая выборка должна содержать примеры каждого класса из задачи распознавания.

**Истинная классификация объекта.** Классификация объекта, которая была бы сделана человеком или сложной экспертной системой, которые имели бы неограниченное количество примеров представителей классов данной задачи распознавания.

**Задача распознавания** образов формулируется следующим образом:

ДАНО:

1. Определено множество объектов распознавания
2. Определено K классов
3. Сформулированы N признаков объектов
4. Имеется обучающая выборка из M примеров, т.е. M объектов для каждого из которых известны
   1. значения N признаков
   2. указана принадлежность к классу
5. Имеется объект, про который известны его N признаков (задан вектор **х** из N-мерного пространства признаков)

НАЙТИ: для объекта, характеризуемого вектором признаков **х**, определить номер класса, по возможности, наиболее соответствующему *истинному*.

**Классификатор**. Функция (алгоритм) вида Ф(**x**,**w**), которая возвращает предполагаемый номер класса объекта, характеризуемого вектором признаков **x**. Работа функции также зависит от набора параметров, задаваемых **вектором состояния** классификатора **w**.

Например, для персептрона данная функция может иметь вид Ф(**x**,**w**) = sign(**x**\***w**), где \*-скалярное умножение векторов. Очевидно, что для персептрона в таком случае размерность вектора w должна совпадать с размерностью вектора признаков х. Для классификатора методом построения эталонов (для задачи с S классами) классификатор может быть задан функцией

Ф(x,w) = arg min [ |x-x1| , |x-x2| , …, |x-xS| ], где под вектором **w** подразумевается следующий составной вектор, состоящий из вертикально «склеенной» цепочки из векторов x1…xS.

приставка arg означает, что ищется не минимальное значение, а НОМЕР минимального аргумента функции min (соответствующий, очевидно, номеру класса).

Структуру вектора **w** можно описать следующей диаграммой:

|  |  |  |  |
| --- | --- | --- | --- |
| w = | x1 | x1[1] | w1 |
| x1[2] | w2 |
| … | … |
| x1[N] |
| x2 | x2[1] |
| x2[2] |
| … |
| x2[N] |
| … | … |
| xS | xS[1] |
| xS[2] |
| … |
| xS[N] | wS\*N |

Соответственно, количество параметров классификатора (размерность вектора **w**) для метода построения эталона = S\*N.

Для метода ближайшего соседа:

Ф(x,w) = D\*(arg min [ |x-x1| , |x-x2|, …, |x-xK| ]), где K – количество примеров в обучающей последовательности. x1…xK – вектора признаков примеров, D\*(1)…D\*(K) – указания учителя. Очевидно структура вектора w состояния классификатора аналогична для случая метода построения эталонов и размерность вектора w равна N\*K+K (К параметров прибавляется за счет необходимости хранить К значений указаний учителя). То есть вектор w хранит ВСЮ обучающую последовательность, которая может оказаться очень большой.

**Среднеквадратичная ошибка обучения.** Функция вида

E(w) =

[Ф(x1,w)-D\*(1)]^2 +

[Ф(х2,w)-D\*(2)]^2 +

…

[Ф(хK,w)-D\*(K)]^2,

где x1…xK, D\*(1)…D\*(K) – обучающая последовательность.

**Задача обучения классфикатора**.

Дано: постановка задачи распознавания.

Найти: arg min E(w), т.е. такое значение вектора w, при котором значение ошибки будет минимальным.

**Способ обучения классификатора**. Определенный математически, метод (алгоритм) решения задачи обучения классфикатора.

**Применение обученного (настроенного) классификатора**. Описание работы формулы Ф(x,w).

**Неитеративные методы обучения**. Методы для которых можно определить формулу

w = F(x1..xK,D\*(1)…D\*(K)),

такую, что найденное значение соответствует некоторму локальному минимуму функции ошибки.

**Итеративные методы обучения**. Методы для которых задается итеративная формула

w(n+1) = F(w(n)), которая служит поиску минимума функции ошибки.

**Обобщающее свойство классификатора**. Способность классификатора выдавать истинные значения классов для объектов, не входивших в обучающую выборку.

Типовые достоинства методов: простота, хорошие обобщающие свойства, низкие требования к памяти (маленький размер вектора состояния w), потребность в короткой обучающей последовательности, устойчивость к ошибкам в указаниях учителя.

Типовые недостатки методов: сложность, большая потребность в памяти для хранения вектора w, большая потребность в памяти для хранения промежуточных переменных, большое количество вычислений, необходимость в длинной обучающей последовательности, неустойчивость к ошибкам в указаниях учителя, невозможность применения для решения произвольных задач распознавания (а только для частных), для итеративных алгоритмов – плохая (медленная или нестабильная) сходимость алгоритма.

**Кластеризация** (Таксономия, самообучение, обучение без учителя). Процесс выделения в пространстве признаков областей, которые могли бы быть объединены в группы человеком либо согласно гипотезе о компактности. Кластеризация применяется для поиска закономерностей в данных, например, выделения неизвестных признаков, поиск объектов среди шумов.

Типовая постановка задачи кластеризации:

**Дано:** обучающая последовательность без указаний учителя x1…xK.

**Найти:** число кластеров М, построить Ф(x,w) разбивающую пространство признаков на М связанных областей, таких что:

* в каждой области средняя плотность точек из обучающей последовательности максимальная
* области разделены между собой промежутками с минимальной средней плотностью точек

**Пример алгоритма кластеризации**.

Суть алгоритма заключается в попытке найти устойчивое разбиение обучающей выборки на гиперсферы одинакового радиуса.

Алгоритм состоит из 3х этапов.

1ый этап. Составление М последовательных разбиений пространства на сферы радиусов R1…RM, где Ri +1 = Ri – dR, где dR = Rmax/G. G – параметр, запрашиваемый пользователем (например 20). R0=Rmax – это сфера в центре обучающей последовательности, с радиусом таким, чтобы все элементы обучающей последовательности в нее умещались. На псевдокоде алгоритм можно записать следующим образом:

1. Считать из файла обучающую последовательность в массив X[j], Y[j] j = 1..K (для случая если размерность пространства признаков = 2)
2. Зарезервировать массив меток точек обучающей последовательности T[j], j = 1..K
3. Запросить у пользователя G
4. Посчитать центр обучающей последовательности и Rmax
5. dR = Rmax/G
6. i = 0
7. R[i] = Rmax
8. Цикл А (**разбиение на классы с заданным размером гиперсферы**)
   1. количество обнаруженных классов M = 0
   2. обнулить отметки точек обучающей последовательности
   3. Цикл Б (**поиск всех кластеров с заданным размером гиперсфер**)
      1. Выбор первой непомеченной точки из обучающей последовательности = х, если непомеченных точек не осталось, выход из цикла Б.
      2. M = M + 1
      3. Цикл В (**поиск оптимального центра гиперсферы**)
         1. r = центр всех непомеченных точек, попавших в гиперсферу радиуса R[i], c центром х
         2. Померить расстояние между r и х = d,
         3. x = r
         4. если d меньше установленного порога (например, 0.1) – конец цикла B
      4. Записать в файл i,х,R[i], M
      5. Пометить все точки, которые попали в гиперсферу с центром в х и радиусом R[i]
   4. Конец Цикла Б
   5. i = i+1
   6. R[i] = R[i-1] – dR
9. Если R[i] < 0 конец цикла А

2й этап. Построение графика количества найденных классов от размера гиперсферы и поиск участка, где изменение радиуса «долго» не приводило к изменению количества классов.

Подобный участок графика как раз показывает, что было найдено разбиение на гиперсферы такое, что размер сфер не сильно влияет на разбиение, то есть между кластерами достаточно пустого места. Поскольку центры кластеров согласно первому этапу выбираются в местах максимального скопления элементов обучающей последовательности – выбор именно такого разбиения соответствует определению кластера.

3й этап. Считывание радиуса и центров гиперсфер соответсвующих варианту выбранному на предыдущем шаге из файла созданного на 1ом шаге.

**Статистические методы распознавания образов**

**P(a) –** вероятность наступления события а

**P(a,b)** – вероятность того, что события а и b наступают одновременно. Если а и b – статистически независимые переменные, то P(a,b) = P(a)P(b)

если события зависимые, то P(a,b) = P(b,a) = P(a)\*P(b|a) = P(b)\*P(a|b)

отметим, что из этого равенства следует формула Баеса:

P(a|b) = P(b|a)\*P(a)/P(b)

то есть это можно считать выводом формулы Баеса для распознавания образов, если вместо a и b подставить соответствующие события, описанные ниже.

**P(a|b) – условная вероятность**. Вероятность наступления события a, при условии что событие b наступило

P(Ck,x) = P(x, Ck). Вероятность того что объект имеет признаки, заданные вектором х и при этом принадлежит классу Сk

P(Ck|x) - вероятность того, что объект относится к классу Сk, при условии, что его признаки описываются вектором х. Также называется «апостериорная вероятность класса Ck», т.е. иными словами, вероятность после измерения величины х.

P(x|Сk) - вероятность того, что объект имеет признаки, описываемые вектором х, при условии что объект относится к классу Ck

P(Ck) – вероятность того, что данный объект относится к классу Ck. Также называется «**априорная вероятность класса Ck**»

P(x) – вероятность того, что признаки произвольного объекта из множества объектов задачи распознавания будут иметь вектор признаков х

p(x) – функция плотности распределения случайной величины х.

формула Баеса для дискретных распределений

P(Ck|x) = P(x|Ck)\*P(Ck)/P(x)

P(C1|x) + P(C2|x) + … P(CM|x) = 1 (где М- число классов в задаче распознавания. Это очевидно, так как объект всегда принадлежит одному из классов в задаче распознавания)

Подставляя в эту формулу формулу Баеса получаем:

P(x|C1)\*P(C1)/P(x) + P(x|C2)\*P(C2)/P(x) + … + P(x|CM)\*P(CM)/P(x) = 1

домножаем на P(x), эта вероятность очевидно не нулевая

получаем

P(x|C1)\*P(C1) + P(x|C2)\*P(C2) + … + P(x|CM)\*P(CM) = P(x)

поэтому формулу Баеса можно записать и без P(x):

P(Ck|x) = P(x|Ck)\*P(Ck) /

[P(x|C1)\*P(C1) + P(x|C2)\*P(C2) + … + P(x|CM)\*P(CM)]

**Формула Баеса для непрерывных распределений**:

P(Ck|x) = p(x|Ck)\*P(Ck) /

[p(x|C1)\*P(C1) + p(x|C2)\*P(C2) + … + p(x|CM)\*P(CM)]

Вывод для одномерного случая:



Последнее равенство возможно, так как получается, что выражение не зависит от dx.

**Матрица потерь:**

Uij - диагональные элементы означают бонус, который мы получаем за правильное распознавание, остальные элементы означают потери, которые мы понесем, если объект j распознаем как i.

Пример матрицы потерь:

Задача – построить матрицу потерь для системы «банк купюр». Система распознает денежные купюры достоинством 10 100 и 1000 единиц и раскладывает по соответствующим полкам. Хранение каждой распознанной купюры обходится в 50 единиц. Поэтому после распознавания, все купюры достоинством 10 единиц уничтожаются. Купюры достоинством 1000 единиц нужно проверить на подлинность, что стоит 100 единиц. Несовпадение номиналов при провеке вызывает перепроверку всех купюр, которая стоит 1000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | истиное значение класса | | | |
| значение класса выдаваемое классификатором |  | 10 | 100 | 1000 |
| 10 | 0, десятку как и положено уничтожили, мы ничего не получили в копилку, но и ничего не потеряли на хранении | «сотку» приняли за «десятку», значит ее уничтожат, и на самом деле мы получим убыток в 100-50 = 50 (не 100, так как 50 не придется платить за хранение) | тысячу приняли за десятку и уничтожили – убыток 1000-50 = 950, но не стали проверять на подлинность, так что на самом деле убыток меньше, 850 |
| 100 | десятку приняли за сотку, значит ее сохранят, но 50 придется отдать за хранение, убыток = 1000(за ошибку)+40 | 50 (сто минус 50 за хранение) | тысячу приняли за сотку, ее сохранят, и не будут проверять на подлинность, но заплатят за перепроверку, убыток = 1000-950 = 50 |
| 1000 | десятку приняли за 1000, значит ее сохранят, но придется отдать 50 за хранение, и еще проверить на подлинность и еще заплатить потом за перепроверку, убыток = 50+100+1000-10 = 1140 | сотку приняли за тысячу. ее сохранят, и будут проверять на подлинность, и еще заплатят потом за перепроверку убыток = 1050 | 850 (1000 минус 50 за хранение, минус 100 за проверку) |

итоговая матрица:

|  |  |  |
| --- | --- | --- |
| **0** | -50 | -850 |
| -1040 | **50** | -50 |
| -1140 | -1050 | **850** |

Предположим система выдает равные вероятности на все три купюры (по 1/3). Какое нужно принять решение чтобы минимизировать риск потерь (максимизировать выгоду)?

Если примем решение, что это «10», то средний риск потерь (через мат. ожидание) =

0\*1/3 + (-50)\*1/3 + (-850)\*1/3 = -900/3 = -300

Если примем решение, что это «100», то средний риск потерь (через мат. ожидание) =

-1040\*1/3 + 50\*1/3 + (-50)\*1/3 = ~ - 347

Если примем решение, что это «1000», то средний риск потерь (через мат. ожидание) =

-1140\*1/3 + (-1050)\*1/3 + 850\*1/3 = ~ - 447

Т.е. если классификатор «не уверен», то выбирать нужно «10», в основном по причине того, что такая купюра выбрасывается и не может сложиться ситуации в необходимости перепровеки всех купюр при нахождении несовпадающих номиналов.

Итак, **решающее правило с учетом матрицы потерь** можно записать так:

arg max [U\*p],

где U – матрица потерь, p – вектор решений классфикатора [P(C1|x)…P(CM|x)]T, \*-умножение матрицы на вектор

Заметим, что если матрица диагональная

|  |  |  |
| --- | --- | --- |
| **1** | 0 | 0 |
| 0 | **1** | 0 |
| 0 | 0 | **1** |

То arg max [U\*p] = arg max p, т.е. выбирается решение, соответствующее максимальному выходу классификатора.

**Рандомизированное решающее правило.** Определение см. в Metogy\_r.doc, стр 38.

**Отказ от гипотезы о компактности.** В статистических методах мы имеем дело с вероятностями событий (условными вероятностями принадлежности объектов к определенным классам при условии, что их признаки принимают определенные значения) и абстрагируемся от истинных причин принадлежности объектов к тем или иным классам. Мы не говорим о «расстоянии» между векторами признаков объектов в пространстве признаков.

Если детерминистические методы основаны на понятии «схожести» и расстоянии между объектами классов, то статистические методы основаны на построении функций распределения.

Детерминистические методы имеют погрешность в том, что расстояние до объекта из обучающей выборки зависит от количества элементов в обучающей выборке. Т.е. например, расстояние до ближайшего элемента может меняться с ростом числа примеров.

Статистические методы имеют погрешность в том, что по обучающей выборке мы можем лишь ***оценить*** распределение с некоторой точностью, и не можем узнать его *истинную* структуру. Чем больше различных элементов обучающей выборки мы имеем, тем более точно мы сможем оценить распределение. В этом прослеживается связь между статистическими и детерминистическими методами распознавания.

**2 способа распознавания по формуле Байеса**. В первом случае моделируются распределения и с учетом априорных вероятностей вычисляются апостериорные вероятности. Во втором случае просто пространство признаков делится на области принятия решений.

**Нейронная сеть**. В данном документе не объясняется, так как это предмет отдельного курса. Однако понимание что это такое требуется. Нейронная сеть позволяет топологически разбить пространство признаков на области, соответствующие определенным классам, минуя шаг построения распределений. Каждый нейрон в таком случае реализует разделяющую поверхность в пространстве признаков.

**Статистическая интерпретация метода к-ближайших соседей**. Очень грубая оценка распределений дает:

Пусть V – гиперсфера заданного радиуса в пространстве признаков, с центром в точке соответствующей вектору (х) признаков распознаваемого объекта.

Пусть Nk – число объектов класса k, попавших в гиперсферу V

TOTALk – число объектов класса k, попавших в обучающую выборку

TOTAL – общее число объектов в обучающей выборке

ALL in V – общее число объектов обучающей выборки, попавших в V

p(x|Ck) ~ P(x in V | Ck) / V, V->0

P(x in V| Ck) ~ Nk (in V) / TOTALk если TOTALk ->inf

P(Ck) ~ TOTALk / TOTAL ~ Nk / ALL in V, V->inf

p(x) = P(x in V) / V, V->0 ~ [ALL in V / TOTAL] / V

P(Ck | x) = p(x | Ck) P(Ck) / p(x) ~

[Nk / (TOTALk\*V)]\*[TOTALk/TOTAL] \* [V\*TOTAL/[ALL in V]] = Nk/[ALL in V]

**То есть P(Ck|x) = Nk/[ALL in V]**

Заметим, что

P(C1|x)+P(C2|x)+… = N1/[ALL in V] + N2/[ALL in V] +… = [ALL in V]/[ALL in V] = 1

Получился **метод к-ближайших соседей** (к соседей заключенных в гиперкуб объемом V)

Т.е. апостерироная вероятность класса для вектора признаков х примерно равна отношению числа элементов данного класса в гиперсфере с центром в х к общему числу элементов обучающей последовательности в гиперсфере V.

Если [ALL in V] = 1, то получается метод **ближайшего соседа**.

Другие способы оценки распределений:

**Параметрическая оценка нормального распределения**: p(x|Ck) = Aexp(-(x-m)^2/g), где m – среднее для векторов признаков класса Ck, g – пропорциональна дисперсии, А – нормировочная константа (зависит от g).

**Последовательная процедура распознавания** – если измерение признаков дорого обходится, можно решать задачу последовательно для n измерений пространства признаков n = 1…n(max). На каждом шаге оценивать разрешаюшую способность

max P(Ck|x)

---------------------

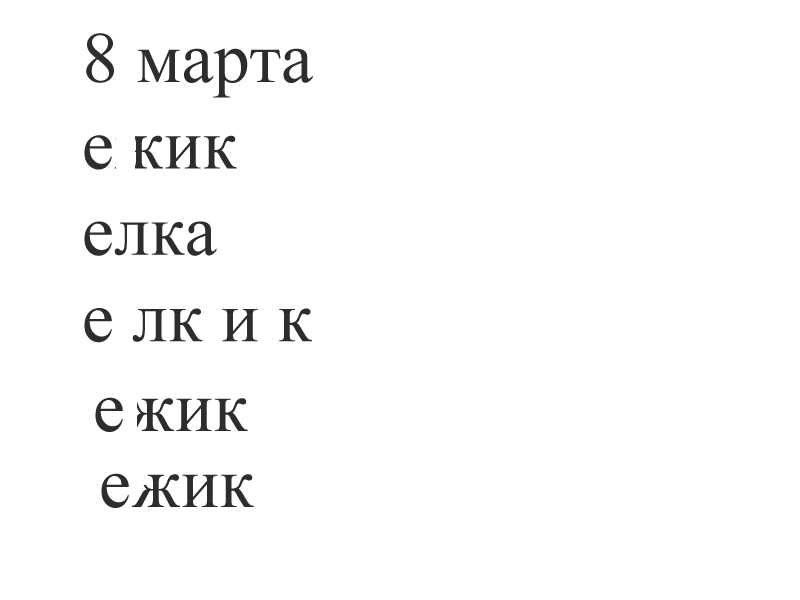
max P(Cj|x) , j<>k

Если разрешающая способность превышает заданный порог (например 2 будет означать, что мы как минимум в 2 раза более уверены в одном из решений относительно любого другого), то принимается решение на этом шаге либо по максимальной вероятности либо исходя из матрицы потерь. В противном случае мы измеряем следующий признак до тех пор пока признаки не кончатся. Если разрешающий критерий не достигается при всех измеренных признаках необходимо принимать решение с теми вероятностями P(Ck|x), которые получились, либо вообще отказаться от принятия решения и выдать ответ «не могу решить задачу с требуемой точностью».

**Распознавание при неизвестных априорных вероятностях**. Самый простой способ – метод монте-карло. Подробности см. metogy.doc

**Иерархические системы распознавания.** Пример – распознавание слов. Вначале распознаются буквы, затем уже из букв слова. При этом решение на первом уровне распознавания не принимается, а лишь вычисляются вероятности. Затем вычисляются вероятности возможных образов получаемых на следующем шаге при всех возможных комбинациях распознавания первого уровня. Вероятности первого уровня и второго комбинируются и выбирается решение, учитывая обе вероятности (максимально правдоподобное решение с учетом вероятностей первого и второго уровня). Пример:

Допустим для следующего образа



система выдала следующие варианты на первом уровне распознавания:

|  |  |  |
| --- | --- | --- |
| Ё-1 |  |  |
| Ж-0.2 | К-0.5 | С-0.3 |
| И-1 |  |  |
| К-0.3 | Х-0.7 |  |

Из этих вариантов можно составить следующие образы второго уровня (цифра внизу означает встречаемость данного слова)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Е** | **Е** | **Е** | **Е** | **Е** | **Е** |
| **К** | **С** | **Ж** | **К** | **С** | **Ж** |
| **И** | **И** | **И** | **И** | **И** | **И** |
| **К** | **К** | **К** | **Х** | **Х** | **Х** |
| 0 | 0 | 0.0009 | 0 | 0 | 0.0001 |

Для удобства напишем снизу все вероятностные составляющие решения первого уровня, и перемножим:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Е** | **Е** | **Е** | **Е** | **Е** | **Е** |
| **К** | **С** | **Ж** | **К** | **С** | **Ж** |
| **И** | **И** | **И** | **И** | **И** | **И** |
| **К** | **К** | **К** | **Х** | **Х** | **Х** |
| 0 | 0 | 0.0009 | 0 | 0 | 0.0001 | |
| 0.5 | 0.3 | 0.2 | 0.5 | 0.3 | 0.2 | |
| 0.3 | 0.3 | 0.3 | 0.7 | 0.7 | 0.7 | |
| **0** | **0** | **0.000054** | **0** | **0** | **0.000014** | |

Отсюда видно, что система примет решение «ЕЖИК», не смотря на то, что максимальную вероятность на первом уровне распознавания получило бы несуществующее сочетание «ЕКИХ», и более того, даже если рассматривать существующие комбинации «ЕЖИХ» и «ЕЖИК», комбинация «ЕЖИХ» имеет большую суммарную вероятность на первом уровне, тем не менее за большего счет встречаемости слова «ЕЖИК», выиграла именно эта «интуитивно угадываемая» комбинация букв.

ЗАДАЧИ.

1. Система распознает заглавные буквы латинского алфавита, предварительно выделяя из изображений символов 10 числовых признаков. Для обучения классификатора используется обучающая последовательность из 100 примеров каждой буквы. Найти количество параметров состояния классификатора (размерность вектора w) для методов
   1. построения эталона
   2. ближайшего соседа
   3. к-ближайших соседей для радиуса гиперсферы R = 10
2. Предложите минимально достаточный набор признаков, для решения задачи распознавания овощей – картофель, свекла, огурец, помидор. Напишите примеры (эталоны) векторов признаков для всех классов. Для каждого признака и каждого класса постройте грубо график одномерного распределения p(x|Ck) с указанием масштаба. Для оценки распределения можно считать, что оно нормальное.
3. При решении задачи распознавания объекта с вектором признаков х методом к-ближайших соседей (радиус гиперсферы = 24, число признаков 2, количество элементов обучающей последовательности 170), классификатор выдал для всех классов одинаковую оценку P(Ck|x). При этом в гиперсфере с центром в точке х всего содержится 17 элементов обучающей последовательности. Определить количество элементов алфавита.
4. Две машины были распознаны роботом по совпадающим признакам – форма колес, радиатора, бампера, а также **форме** и **цвету** всех остальных деталей кузова и салона и отнесены к одному классу машин. Однако человек счел один объект пригодной машиной, а другой абсолютно нет. Какой дополнительный признак анализировал человек и не анализировал робот? Дайте математическое объяснение обоим случаям, применяя гипотезу о схожести.
5. Человек, ведя машину, с высокой вероятностью распознаёт, попадает ли правое или левое переднее колесо его автомобиля в ямку на дороге или нет. Разумеется человек никогда с линейкой не меряет ширину свого автомобиля или ширину дороги, равно как и не может зрительно точно оценить абсолютные расстояния. Поставьте задачу распознавания (алфавит, признаки, обучающая последовательность).
6. Может ли персептрон безошибочно решать задачу распознавания поступления или не поступления студента в вуз по признакам балл по математике и балл по физике, если для них заданы «проходные значения»? Почему? Нарисуйте пространство признаков, разметьте на нем области классов, покажите на рисунке разделяющую поверхность персептрона, которая всегда правильно «поступала» хороших студентов, но при этом принимала бы иногда и непроходных. Покажите разделяющую поверхность персептрона, которая точно «отправляла» бы всех непроходных студентов, но при этом страдала бы и часть «проходных».
7. Для задачи 6 укажите два способа , которыми можно было бы регулировать настройку классификатора чтобы склонить его к большей или меньшей лояльности к студентам. Подсказка: см. априорная вероятность, риск.
8. Выведите формулу Байеса, приведите знаменатель к форме, содержащий только выражения p(x|Ck) и P(Ck), покажите вывод формулы Байеса для непрерывных распределений в одномерном случае.
9. Выведите статистическую интерпретацию метода к-ближайших соседей и ближайшего соседа.
10. Алфавит состоит из 2х классов А и Б. В обучающую выборку входит 5 элементов класса А. При этом для априорных вероятностей классов известно, что P(А) = 0.5 P(Б). Если соотношение объектов А и Б в обучающей выборке соответствует наблюдаемому в «жизни», то какого примерно количество элементов класса Б в обучающей последовательности.