Лекция 12. Самообучение чрез запомняне - основи

12.1. Екстенсионално описание на понятия

Начините, по които хората научават различни понятия, са предмет на интензивно изследване в когнитивните науки. В началото усилията на изследователите са били концентрирани предимно върху научаването на *логически понятия.* Характерна за тях е възможността за конструиране на определен набор от необходими и достатъчни условия, позволяващи всеки произволно избран обект да бъде определен като принадлежащ или не към дадено логическо понятие. Освен това всички примери на едно логическо понятие го представят в еднаква степен. Да разгледаме като пример понятията “четни числа” и “баба”. Всяко число е или четно, или нечетно и не съществува число, което е “по-четно" от друго. Понятието “баба" може да се дефинира като майка на един от родителите и всяка конкретна баба в една и съща степен е добър пример на това понятие.

През последните 70 години акцентът на изследванията е бил пренесен върху изучаването на *естествени понятия* - понятията, които се срещат във всекидневния живот. Естествените понятия се различават от логическите по това, че не могат да бъдат определени чрез набор от необходими и достатъчни условия. За първи път този проблем е бил формулиран от австрийски философ Людвиг Витгенщайн [Wittgenstein 1953], който опитва да определи понятието “игра”. Той утвърждава, че това понятие съдържа голямо разнообразие от обекти. Всяка игра е подобна на други игри по някои свои свойства, но не съществуват свойства, които са общи за всички игри. Оттук следва, че естествените понятия като “игра” не могат да бъдат определени чрез правила. Витгенщайн предполага, че представителите на едно естествено понятие имат *фамилно сходство,* т. е. всеки пример на понятието прилича на *няколко* други представители на същото понятие или има *няколко* еднакви свойства с *някои* негови представители. Не е необходимо той да прилича на *всички* други представители на понятието.

Американките Елеонор Рош и Каролин Мервис [Rosh & Mervis 1974] са развили идеята, предполагайки, че естествените понятия имат *структура* - различните примери на понятието играят в него различни роли. Самите понятия се представят в нашия мозък чрез определени образи на някои от неговите примери — т. нар. *прототипи.* Прототипите са обекти, принадлежащи на понятието и представящи го по най-добрия начин. Например враната обикновено се разглежда като по-типична птица от пингвин, който на свой ред е по-типичен представител на семейството птици от щраус.

В настоящата лекция се разглеждат методите за научаване на естествени понятия. Общото за всички тях е, че научаваните понятия се представят *екстенсионално* - чрез свои представители - екземпляри. Терминът *екземпляр* се използва за означаване на конкретни примери на понятието. Характерните особености на представянето на понятията чрез екземпляри са:

- Понятията са представени не чрез свои необходими и достатъчни признаци, получени от свои екземпляри чрез обобщение, а чрез множества от самите екземпляри. Не се извежда никаква обобщаваща информация във вид на правила.

- Свойствата на понятията са функция или комбинация от свойства на техните екземпляри.

Методите на машинно самообучение, създаващи екстенсионалното описание на научаваните понятия, се наричат в англо-езичната литература с различни имена: instance-based learning, case-based learning, memory-based learning, exemplar-based learning, lazy learning. В настоящия курс ще използваме названието *методи за самообучение чрез запомняне.*

Концептуално, методите за самообучение чрез запомняне реализират един директен подход към задачата за апроксимация на дискретна или реална целева функция. Ако сравним тези методи с методи за научаване на *интенсионалното* описание понятията, то първото отличие е, че в нашия случай за описание на хипотези (понятия) се използва *същият език* както и за описание на примери. Екстенсионалното описание на понятие се представя чрез множество от свои примери – екземпляри, *избрани (или направени) от обучаващите примери* на понятието. Оттук следва, че при самообучението чрез запомняне проблемът е не само в това, какви признаци на примери трябва да бъдат включени в описанието на понятието, а и какви обучаващи примери трябва да бъдат *запомнени (или използвани за конструиране)* като негови представители.

Второто отличие е принципно различен начин на класификация. При самообучение, базирано на обобщение, при класификацията се използва правилото за извод *modus ponens* — от наличие в тестовия пример на признаци, *точно съвпадащи с* интенсионалното описание на понятието, следва неговата класификация като екземпляр на това понятие. При екстенсионалното описание на понятия класификацията на примера се прави чрез оценяването на *частично* съвпадение между него и един или няколко екземпляра, представящи понятието. Тази оценка е количествена мярка за *сходство* между примера и понятието и затова методите за формиране на екстенсионалното представяне на понятията често се наричат *базирани на сходство.*

Ключовата разлика на методи за самообучение чрез запомняне от тези подходи, които сме изучили до сега, е че при самообучение чрез запомняне при класификацията на всеки различен пример се правят различни апроксимации на целевата функция. На практиката, те конструират само *локална апроксимация* на целевата функция, която се прилага само в съседство с конкретен пример, подлежащ на класифициране; *глобалната апроксимация на целевата функция никога не се прави*. Това дава значителни предимства, когато целевата функция е много сложна, тъй като позволява тя да бъде представена чрез колекция от по-прости локални апроксимации.

Методите за самообучение чрез запомняне могат да използват много сложни, символни описания на примери, базирани на знанията за проблемната област, което води и до значително по-сложни алгоритми за идентификация на "съседни" примери. Такива методи се наричат *разсъждения, базирани на прецеденти* (Case-based reasoning) и с успех се използват в областта на юриспруденцията, медицинска и техническа диагностика, дизайн, електронна търговия и т.н., намирайки решение на текущия проблем чрез повторно използване и адаптация на решения от предишни, успешно (и/или неуспешно) решени подобни случаи в миналото.

12.2. Класификация на методите за самообучение чрез запомняне

Методите за научаване на екстенсионалното представяне на понятия могат да се класифицират от гледната точка на:

- *Използвания език за описание на примери.* Ще бъдат разгледани само методи, работещи върху примери, описани на *пропозиционален език.* Всеки признак на примера се представя чрез двойка атрибут - стойност. Ще предполагаме (ако явно не е посочено противното), че броят на атрибутите, използвани за описание на примера, е *един и същ* за всички научавани понятия, като се допуска стойностите на някои атрибути да са *неизвестни.*
- *Използваните основни знания.* Методите, в които основните знания са представени *неявно* само чрез обучаващата последователност от примери на понятията от проблемната област, се наричат *слаби* (weak или knowledge-poor). Методите, използващи освен примери и допълнителни знания, се наричат *силни* (strong или knowledge-intensive). В този курс ще разгледаме както слабите, така и силните методи.
- *Компактността на представянето.* Ще разгледаме както методи, включващи в представянето на понятията всички примери от обучаващото множество, така и методи, избиращи за тази цел само определени обучаващи примери или техни абстракции.

Описанието на всеки метод за самообучение чрез запомняне се състои от две части. Първата описва поведението на метода при *обучение,* т. е. при работата с решени примери, предоставени от учителя. Втората фаза описва начина за класифициране на пример с неизвестна класификация. Както вече беше казано, в основата на класификацията е определянето на сходство между тестовия пример и научените по време на първата фаза понятия, представени чрез свои екземпляри. Примерът се класифицира като принадлежащ на понятието, с което той има най-голямо сходство. Следователно за решаване на тази *класификационна* задача всеки метод трябва да описва:

1. Как да се определи сходството между пример и екземпляр на някое понятие[1].
2. Как да се определи сходството между пример и самото понятие, представено чрез множество от своите екземпляри.

Няколко думи за означения. Тъй като при самообучение чрез запомняне голямото внимание се обръща върху конкретни примери и техните атрибути, за по-голямо удобство ще означаваме самият пример с голяма буква (например X или Y), а неговите атрибути – с малка (например x_k или y_k). Обучаващия пример, като винаги, е двойка $<X, f(X)>$, където f е целевата функция. Целевата функция може да бъде както дискретна, така и непрекъсната. В първия случай тя има вид $f : \Re^n \rightarrow C$, където C е крайното множество от *класове* $\{c_1, \ldots, c_m\}$.

12.3. Мерки за сходство

В основата на всички методи за машинно самообучение на понятия чрез запомняне лежи идеята, че *понятията групират сходни обекти.* Следователно въпросът за класифициране на нов обект може да бъде решен чрез определяне на степента на сходството му с обекти, представящи вече известни понятия. Съществуват два съществено различни начина за определяне на сходството между обектите. Първият се нарича *геометричен* и определя сходството чрез изчисляване на разстоянието между обекти, представени като точки в n-мерното пространство на признаците. При втория подход — *теоретико-множествен,* сходството се определя като функция на броя на общите и различаващите се признаци в сравнявани обекти. В раздела е описан и *статистическият* подход, при който метриката на сходство се извежда статистически от *всички примери* (обекти), съдържащи се в обучаващото множество.

10.3.1. Геометричен подход за определяне на сходство

При геометричната интерпретация, сходството $s(X, Y)$ между двата примера X и Y се определя като *близост* - величина, обратна на *разстоянието* $d(X, Y)$ между съответните n-мерни точки $X = (x_1, \ldots, x_n)$ и $Y = (y_1, \ldots, y_n)$. В общия случай тази зависимост се задава от формулата:

$$s(X,Y) = \alpha - \beta d(X,Y)$$

където α и β са числови константи, използвани за настройване на стойностите на сходството в желания интервал. Разстоянието или *метриката* $d(X, Y)$ се дефинира като функция, изобразяваща скаларно произведение на множеството от примери върху самото себе си върху множество от неотрицателни реални числа и

[1] Напомняме, че примерът е или представител на неизвестно понятие *(тестов* пример), или такъв представител на известно понятие, за който още не е взето решение дали той трябва да бъде включен в описанието на понятието като негов екземпляр *(обучаващ* пример).

удовлетворяваща следните три условия:

- *Симетричност:*
  $d(X, Y) = d(Y, X)$
- *Положителна определеност:*
  $d(X, Y) > 0; \qquad d(X, Y) = 0 \Leftrightarrow X = Y.$
- *Неравенството на триъгълника:*
  $d(X, Y) \leq d(X, Z) + d(Z, Y)$
  за произволна тройка от обекти X, Y и Z.

Един общ клас от разстояния се задава чрез *метриките на Минковски*:

$$d_L(X,Y) = \left(\sum_{k=1}^{n} \delta^L(x_k, y_k) \right)^{\frac{1}{L}}, \quad \delta(x_k, y_k) = | x_k - y_k |, \quad L \geq 1.$$

При $L = 2$ получаваме *Еклидовото* разстояние, което най-често се използва (в различни варианти) поради неговата геометрична "интуитивност". $L = 1$ дава *абсолютното* разстояние (което често се нарича *разстояние по Хеминг*). Абсолютното разстояние се използва не само за непрекъснати, но и за номинални атрибути, като се приема, че атрибутното разстояние $\delta(x_k, y_k)$ се изчислява по следната формула:

$$\delta(x_k, y_k) = | x_k - y_k | = \begin{cases} 0 & \text{при } x_k = y_k \\ 1 & \text{при } x_k \neq y_k \end{cases}$$

При $L = \infty$ получаваме *разстояние по Чебишев*, което се редуцира до

$$d_\infty = \max_k | x_k - y_k |$$

Прилагането на описаните метрики към примери, съдържащи непрекъснати атрибути, често се предшества от *процедурата за нормализация,* целта на която е да приведе стойностите на различни признаци в един и същ диапазон и по този начин да гарантира, че всички те имат един и същ принос в изчисляване на разстоянието. Това става чрез изваждане от стойността на атрибута на неговото средно аритметично, изчислено по всички стойности на същия атрибут в обучаващите примери, и разделяне на резултата на стандартното отклонение на признака. Другият често използван начин за нормализацията е чрез разделяне на всяка стойност на атрибута на разликата между максималната и минималната стойност на същия атрибут, съдържащи се в множеството от обучаващите примери.

Ако разглеждаме примерите като n-мерни вектори, описани само *с непрекъснати* атрибути, сходството между два примера X и Y може да бъде оценено чрез коефициента на корелация:

$$r(X,Y) = \frac{\sum_{k=1}^{n}(x_k - \tilde{x})(y_k - \tilde{y})}{\left[\sum_{k=1}^{n}(x_k - \tilde{x})^2 \sum_{k=1}^{n}(y_k - \tilde{y})^2\right]^{\frac{1}{2}}},$$

където $\tilde{x}$ и $\tilde{y}$ са средните аритметични съответно на *X* и *Y*. Коефициентът на корелация представлява косинус на ъгъла между векторите *X* и *Y*, които са центрирани относно средното аритметично и са нормирани, за да имат единична дължина.

12.3.2. Теоретико-множествен подход за определяне на сходство

Теоретико-множественият подход към оценяването на сходство между обекти се основава на знаменитата работа на психолога Амос Тверски "Features of Similarity" [Tversky 1977], която е класика за съвременните изследвания по изучаването на сходство. Тверски твърди, че сходството между два обекта е функция на три променливи:

- броя на признаците, общи и за двата обекта,
- броя на признаците, уникални за първия обект, и
- броя на признаците, уникални за втория обект.

Повечето от съществуващите функции на сходство са симетрични и дефинирани в интервала [0,1] или $[0,\infty]$. Ако някоя функция на сходство *s(X, Y)* е дефинирана в интервала [0, A], то *A - s(X, Y)* определя функцията на *различие* между обекти.

Тази интерпретация на сходството е удобна за обекти, описани чрез *двоични (логически) атрибути*. Конкретен пример на функция, оценяваща различието между два примера, описани чрез двоични атрибути, дава *Жакардовия коефициент:*

$$dis_1(X,Y) = 1 - \frac{card(E)}{card(E) + card(B) + card(C)},$$

където:

c*ard(M)* е броят на елементи на множеството *M*;
E - множеството от признаци, които присъстват едновременно и в двата примера;
B - множеството от признаци, които присъстват в примера *X* и отсъстват в *Y*;
C - множеството от признаци, които присъстват в *Y* и отсъстват в *X*.

Другите варианти на функциите за различие са:

$$dis_2(X,Y) = 1 - \frac{card(E)}{card(E) + card(B) + card(C) + card(D)}$$

и

$$dis_3(X,Y) = 1 - \frac{2card(E)}{2card(E) + card(B) + card(C)}.$$

За разлика от Жакардовия коефициент, отчитащ само наблюдаваните признаци, функцията *dis₂(X,Y)* отчита и признаци, липсващи и в двата примера (множеството *D*), а последната функция се отличава от Жакардовия коефициент само по използваните тегловни коефициенти.

Разгледаните функции за сходство (различие) са приложими за оценяването на обекти, описани с двоични (логически) атрибути (признаци), обаче могат да бъдат използвани и при номинални атрибути, приемащи повече от две дискретни стойности. За тази цел атрибутите се подлагат на *процедура за бинаризация,* при която всеки номинален атрибут, приемащ *n* стойности, се представя чрез n-мерен вектор от двоични признаци.

Пример

Да разгледаме три обекта — праскова, портокал и банан, описани чрез номиналните атрибути цвят, форма, вкус, повърхност и структура:

Атрибут	*Понятия*		
	праскова	*портокал*	*банан*
Цвят	оранжев	оранжев	жълт
Форма	кръгъл	кръгъл	дълъг
Вкус	сладък	сладък	сладък
Повърхност	мъхеста	гладка	гладка
Структура	костилка	семки	сегменти

След процедурата за бинаризация на атрибутите същите обекти приемат следния вид:

Понятие	*Признаци*									
	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
Праскова	1	0	1	0	1	1	0	1	0	0
Портокал	1	0	1	0	1	0	1	0	1	0
Банан	0	1	0	1	1	0	1	0	0	1

където:

f_1: 1 = цвят оранжев	0 = цвят не оранжев
f_2: 1 = цвят жълт	0 = цвят не жълт
f_3: 1 = форма кръгъл	0 = форма не кръгъл
f_4: 1 = форма дълъг	0 = форма не дълъг
f5: 1 = вкус сладък	0 = вкус не сладък

f_6:	1 = повърхност мъхеста	0 = повърхност не мъхеста
f_7:	1 = повърхност гладка	0 = повърхност не гладка
f_8:	1 = структура костилка	0 = структура не костилка
f_9:	1 = структура семки	0 = структура не семки
f_{10}:	1 = структура сегменти	0 = структура не сегменти

12.3.3. Статистически подход към оценяване на сходство

Един ефективен метод за оценяване на сходството между примери, описани чрез *номинални* атрибути, е предложен от Стенфил и Уолтц [Stanfill & Waltz 1986] за решаване на задачи, свързани с произнасяне на английски думи. В основата на метода е *метриката на различия между стойностите* (VDM — Value Difference Metric), която определя разстояния между различни номинални стойности на *един и същ* атрибут. Ще разгледаме по-опростен (модифициран) вариант на тази метрика (MVDM). Метриката MVDM се извежда статистически от всички примери, съдържащи се в *обучаващото* множество. Атрибутното разстояние между две номинални стойности a_i и b_i на i-тия атрибут се определя по следната формула:

$$\delta_{MVDM}(a_i, b_i) = \sum_{k=1}^{|C|} | P(c_k \mid a_i) - P(c_k \mid b_i) |^q$$

където $|C|$ е броят на различните класове (понятия) в обучаващото множество, c_k е k-тият клас (k = 1,..., |C|), а q - целочислен параметър (q = 1, 2, 3,...), който се определя емпирично. $P(c_k \mid a_i)$ е условната вероятност за срещане на класа c_k, съдържащ a_i като стойност на i-тия атрибут. Тази вероятност се определя по следната формула (същото важи и за b_i):

$$P(c_k \mid a_i) = \frac{\#a_i^k}{\#a_i}$$

където $\#a_i^k$ е броят на срещане на a_i в обучаващите примери, описващи понятието c_k, а $\#a_i$ - общият брой на срещане на тази стойност на i-тия атрибут във всички обучаващи примери. Общото разстояние между примерите X и Y се определя по формулата:

$$d_{MVDM}(X, Y) = \sum_{i=1}^{n} \delta_{MVDM}^{L}(x_i, y_i), \quad L = 1, 2$$

Съгласно описаната по този начин метрика, две номинални стойности на атрибута са сходни, ако те се срещат с една и съща относителна честота във всички класове. Различни варианти на MVDM са били използвани с успех в молекулярната биология и други проблемни области.

Пример

Да предположим, че имаме някакво обучаващо множество от примери, описващи две понятия: α и β. Ще изследваме един от описващите понятията атрибути, който приема три възможни стойности *A*, *B* и *C*. Получените данни позволиха да съставим следната таблица, описваща броя на примерите от различни класове, съдържащи съответната стойност на изследвания атрибут.

Стойности на атрибута	*Класове*	
	α	β
A	4	3
B	2	5
C	4	2

Таблица 12-1. Брой на срещане на атрибутните стойности по класове

Честотата на срещане на *A* за клас α е 57,1%, тъй като 4 примера са класифицирани като принадлежащи към класа α от всичките 7 примера, съдържащи *A* като стойност на проверявания атрибут. Аналогично честотите на срещане на *B* и *C* за същия клас са съответно 28,6% и 66,7%. Честотата на срещане на *A* за клас *β* е 42,9% и т. н. За определяне на разстоянието между *A* и *B* ще използваме съответната формула със стойност на параметъра *L = 1*:

$$\delta(A,B) = \left|\frac{4}{7} - \frac{2}{7}\right| + \left|\frac{3}{7} - \frac{5}{7}\right| = 0{,}571.$$

В Таблица 12-2 са посочени всички разстояния между стойностите на атрибута. Трябва да се има предвид, че за всеки атрибут трябва да се конструира по една такава таблица, т. е. ако например понятията се описват с 10 атрибута, трябва да бъдат конструирани 10 различни таблици.

	Стойности на атрибута		
	A	B	C
A	0,000	0,571	0,191
B	0,571	0,000	0,762
C	0,191	0,762	0,000

Таблица 12-2. Различия между стойностите на атрибута

12.4. Обработка на липсващите признаци

Примерите на понятия в *реални* проблемни области (като например медицината) се характеризират с това, че някои от стойностите на техните атрибути могат да бъдат неизвестни (т.е. *липсват).* За да се приложат в тези области методите на самообучение чрез запомняне, е необходимо да се дефинират начини за определяне на сходството между такива непълни примери. С други думи, трябва да бъдат разширени използваните метрики за сходство за случаите на липсващите стойности на атрибути. В настоящия раздел ще бъдат разгледани някои основни методи за работа с липсващите атрибутни стойности.

Един от най-простите методи за обработка на липсващите атрибутни стойности е методът на *игнориране,* предложен от Дейвид Аха [Aha 1990]. Ако стойността на някой атрибут в примера липсва, то атрибутното разстояние до този пример (по този атрибут) е нула (т.е. атрибутът се игнорира), т. е. ако означим с "?" стойността на липсващия i-ти атрибут, то $\delta(x_i, ?) = 0$. При изчисляване на общото разстояние между примери, за да бъдат различени случаи на пълно съвпадение на атрибутните стойности и тези с игнориране на атрибута (и в двата случая атрибутното разстояние е равно на 0), сумата на всички атрибутни разстояния се дели не на общия брой на атрибутите, а на броя на атрибути с *известни стойности*:

$$d(X,Y) = \frac{1}{N} * \sum_{i=1}^{n} \delta^L(x_i, y_i),$$

където $N < n$ е броят на атрибути с известни стойности, а $L \geq 1$ — целочислен параметър.

При друг метод, който може да бъде наречен *песимистичен,* липсващите стойности на непрекъснати атрибути се подразбират максимално различни от известната стойност, а ако и двете стойности липсват, атрибутното разстояние между тях се приема за равно на 1[2]:

$$\delta(x_i, ?) = \begin{cases} x_i & \text{при } x_i \geq 0{,}5; \\ 1 - x_i & \text{при } x_i < 0{,}5; \\ 1 & \text{при } x_i = ? \end{cases}$$

За номинални атрибути $\delta(x_i, ?) = \delta(?, ?) = 1$.

Един от най-разпространените методи за обработка на липсващите номинални признаци е предположението, че $\delta(x_i, ?) = \delta(?, ?) = 0{,}5$[3].

Едно интересно разширение на метриката MVDM, позволяващо нейното използване за примери с липсващи признаци, е предложено от Педро Домингош. Липсващата стойност на атрибута в обучаващите примери се заменя със символа "?", който се третира като *легитимна номинална стойност.* Това означава, че за нея се изчисляват таблиците на разликите до всички останали (известни) номинални стойности на атрибута, които след това се използват за изчисляване на разстоянията между примери (вижте предишния раздел). В контекста на метриката MVDM тази политика е достатъчно разумна: приема се, че липсващата стойност е

[2] Подразбира се, че всички признаци са нормирани.

[3] Предполага се, че за известни признаци $\delta(x_i\, y_i) = 1$ при $x_i \neq y_i$ и $\delta(x_i\, y_i) = 0$ при $x_i = y_i$.

приблизително еднаква с някоя известна номинална стойност, ако тяхното поведение е сходно (т. е. имат близки честоти на срещане), и че в противен случай те са различни.

Приведеният кратък обзор на основните методи за обработка на липсващите признаци ще завършим с описание на метода, който може да бъде наречен *статистически*. При този подход липсващата стойност на номинален атрибут в един *обучаващ* пример се *замества* със стойността на атрибута, най-често срещана сред обучаващите примери от *същия клас*. Липсващата номинална стойност на атрибута в *тестовия* пример се заменя с най-често срещаната стойност на същия атрибут сред *всички* примери от обучаващото множество. При липсващ *непрекъснат* атрибут се използват съответно средните стойности на атрибута, изчислени върху обучаващите примери от същия клас и върху цялото обучаващо множество.

12.5. Базови алгоритми

В предишния раздел бяха разгледани начини, позволяващи да се оцени сходството между примера и *един-единствен* представител на понятие (неговия екземпляр). В настоящия раздел са описани най-често използваните методи за оценяване на сходство между пример и *понятие,* представено чрез *множеството* от своите екземпляри.

12.5.1. Алгоритъм на най-близкия съсед

Най-използваният метод за самообучение чрез запомняне е *алгоритъм на най-близкия съсед* (NN — Nearest Neighbour). При този метод класификацията на тестовия пример зависи от степента на неговото сходство с *един-единствен представител на понятие* - този, който се намира на най-малкото разстояние от него (неговия най-близък съсед). Приема се, че примерът принадлежи към понятието, чийто представител е най-близкият съсед на примера. Описанието на алгоритъма на най-близкия съсед е приведено в Таблица 12-3.

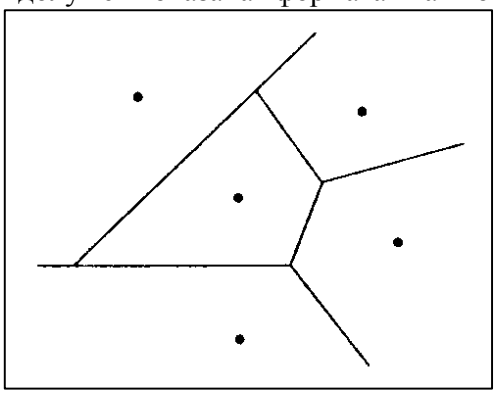
Алгоритъм на най-близък съсед
Дадено:
Обучаващи примери $D = \{<X_i, f(X_i)>\}$
Метриката за различието между примери $d(X, Y)$.

- *Обучение:*
  Запомни всички обучаващи примери $<X_i, f(X_i)>$ като *екземпляри,* формиращи *Описание* на научаваните понятия.
- *Класификация* (класифициране на тестов пример E):
  1. За всеки екземпляр $<X_i, f(X_i)>$ от *Описанието* изчисли разстоянието $d(E, X_i)$ до тестовия пример.

2. Класифицирай E като принадлежащ към класа на най-близкия към E екземпляр: $\hat{f}(E) = f(X_j)$, *където* $j = \arg\min_i d(E, X_i)$

Таблица 12-3. Алгоритъм на най-близък съсед

Каква е природата на пространството на хипотези H, което неявно се разглежда от алгоритъма на най-близкия съсед? Искам да подчертая, че алгоритмът никога не формира в явен вид общата (т.е. отнасяща се за цялото пространство от примери X) хипотеза $\hat{f}$ за вида на целевата функция f. Той просто изчислява класификацията на всеки нов тестов пример при неготово постъпване. Обаче, ние все пак можем да представим, какво представлява тази неявна обща функция, т.е. каква класификация би могла да бъде назначавана, ако представим, че всички обучаващите примери са константи и че пускаме класификационния алгоритъм с всеки възможен тестов пример от пространството на примери X. На диаграмата долу е показана формата на повърхнина на решения (в двумерен вариант), генерирана от алгоритъма за най-близък съсед за цялото пространство на примери. Тази повърхнина е комбинация от изпъкнали n-мерни многоъгълници, окръжаващи всеки обучаващ пример. За всеки обучаващ пример многоъгълникът указва множеството от тестови примери (n-мерни точки), чиято класификация напълно се определя от дадения обучаващ пример. Точките извън многоъгълника са по-близо до някои други обучаващи примери. Този вид на диаграми за множество от обучаващи примери често се наричат *диаграми на Вороной*.



Пример

Да предположим, че една група от хора трябва да бъде разбита на два класа c_1 и c_2 по два признака f_1 и f_2, съответстващи на години и пол (1 означава мъж, а 0 - жена). Екстенсионалното описание на тези класове съдържа следните четири екземпляра:

$\langle X_1, c_1 \rangle = \langle [пол = 1, години = 48], c_1 \rangle$
$\langle X_2, c_1 \rangle = \langle [пол = 0, години = 60], c_1 \rangle$
$\langle X_3, c_2 \rangle = \langle [пол = 1, години = 25], c_2 \rangle$
$\langle X_4, c_2 \rangle = \langle [пол = 0, години = 38]. c_2 \rangle$

Да предположим, *че* трябва да класифицираме следния пример:

$E = [пол = 1, години = 55].$

Прилагайки алгоритъма на най-близкия съсед, който използва като мярка за разстояние например абсолютното разстояние L_1, получаваме, че новият пример

трябва да бъде отнесен към първия клас c_1, тъй като разстоянието до втория екземпляр $d(E, X_2) = |1 — 0| + |60 – 55| = 6$ е най-малко.

12.5.2. Алгоритъм на *k* най-близки съседи

Алгоритмът на най-близкия съсед е представител на една цяла фамилия класификационни методи, носещи название *методи на k най-близки съседи* (k-NN). Числото *k* определя броя на най-близките екземпляри от описанието на понятията, участващи в определяне на решение за класификация на тестовия пример. Примерът се класифицира в съответствие с класа, който най-често се среща сред най-близките *k* съседи на примера. Ако повече от един клас се среща най-често сред най-близките *k* съседи, примерът обикновено се класифицира в съответствие с класа на най-близкия свой съсед сред конкуриращите се класове. Описанието на алгоритъма на *k* най-близки съседи е приведено на Таблица 12-4.

Алгоритъм на k най-близки съседи

Дадено:

Обучаващи примери $D = \{<X_i, f(X_i)>\}$

Метриката за различието между примери $d(X, Y)$.

Параметър k.

- *Обучение:*
  Запомни всички обучаващи примери $<X_i, f(X_i)>$ като *Описание* на научаваните понятия.
- *Класификация* (класифициране на тестов пример *E):*
  1. За всеки екземпляр $<X_i, f(X_i)>$ от *Описанието* изчисли разстоянието $d(E, X_i)$ до тестовия пример.
  2. Създай множеството $Bag(k) = \{X_{i_1},...,X_{i_k}\}$, съдържащо първите k екземпляра най-близки до примера E , $i = 1,..., k$.
  3. Изчисли броя на гласовете (честота на срещане) за всеки клас сред избраните k екземпляра от $Bag(k)$ и класифицирай пример E като принадлежащ към класа, за който са дадени най-много гласове:
     $$\hat{f}(E) \leftarrow \arg\max_{c \in C} \sum_{i=1}^{k} \lambda(c, f(X_i)), \text{ където } \lambda(a,b) = \begin{cases} 1, \text{ако } a = b \\ 0, \text{ако } a \neq b \end{cases}$$
  4. *При еднакъв брой на гласовете* класифицирай E в съответствие с класа на най-близкия до E съсед сред екземплярите от конкуриращите се класове в $Bag(k)$.

Таблица 12-4. Алгоритмът на k най-близки съседи

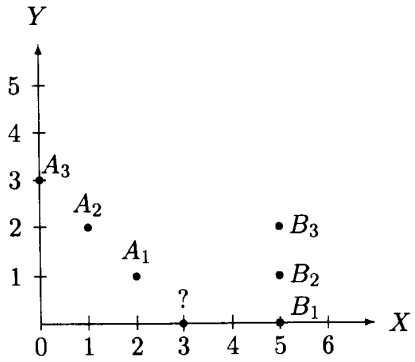
Алгоритъмът на k най-близки съседи може лесно да бъде адаптиран към задачата за апроксимация на непрекъсната целевата функция. За тази цел вместо да изчисляваме най-често срещнатия клас сред най-близките k примера, трябва да изчисляваме тяхната средна стойност. По-точно, за да апроксимираме непрекъсната целева

функция $f : \Re^n \to \Re$ третата точка на описания по-горе алгоритъм трябва да бъде сменена със следното (а четвъртата – просто премахната):

$$\hat{f}(E) \leftarrow \frac{\sum_{i=1}^{k} f(X_i)}{k}, \quad X_i \in Bag(k)$$

Пример

За да сравним работата на алгоритмите NN и k-NN, да разгледаме примера, представен на рисунката по долу. Научаваните понятия *(A* и *B)* са представени чрез три свои екземпляра. За по-голяма нагледност двата описващи понятията атрибута *x* и *y* ще интерпретираме като координати, така че всеки екземпляр се представя като точка в двумерното пространство. Точката, съответстваща на тестовия пример, е означена със знака "?".



Ще приложим алгоритмите NN и 3-NN, използващи една и съща мярка за разстояние - Евклидовото разстояние L_2:

$$d(E, X) = \sqrt{(x_X - x_E)^2 + (y_X - y_E)^2}$$

Ако за класификацията на тестовия пример се използва алгоритъм NN, то примерът ще се класифицира като принадлежащ към *A* съгласно класа на своя най-близък съсед - точката $A_1(2,1)$ ($d(?, A_1) = \sqrt{2}$).

Обаче, ако за класификацията се използва алгоритъм 3-NN, същият тестов пример ще бъде класифициран като принадлежащ към класа *B,* тъй като множеството от трите най-близки до въпросната точка съседи съдържа точките $A_1(\sqrt{2}), B_1(2)$ и $B_2(\sqrt{5})$ (в скобите са посочени разстоянията от съответната точка до точката „?").

12.5.3. Алгоритъм за *k* най-близки съседи, претеглени по разстояние

Една естествена модификация на алгоритъма k-NN е да се претегля гласът на всеки от *k* съседи в зависимост от разстоянието му до примера, подлежащ на класификацията. Очевидно е, че колкото по-близо е разположен един съсед, толкова "по-тежък" трябва да бъде неговият глас. Например, при апроксимация на дискретната целева функция можем да претеглим "глас" на всеки съсед в съответствие с величина, обратна на квадрата на разстояние от този екземпляр до тествания пример. За целта последния ред в т. 3 на алгоритъма k-NN трябва да стане:

$$\hat{f}(E) \leftarrow \arg\max_{c \in C} \sum_{i=1}^{k} w_i \lambda(c, f(X_i)),$$

$$\text{където } w_i = \frac{1}{d(E, X_i)^2},$$

$$\lambda(\mathrm{a}, \mathrm{b}) = \begin{cases} 1, \text{ако } a = b \\ 0, \text{ако } a \neq b \end{cases}$$

При пълно съвпадение на примера *E с* някой екземпляр X_i ,т. е. когато знаменателят $d(E,X_i)$ е равен на нула, примерът се класифицира в съответствие с класа на X_i . (т.е. $\hat{f}(E) \leftarrow f(X_i), \text{ако } d(E, X_i) = 0).$

Ако повече от един екземпляр напълно съвпадат с тестовия пример, за класификацията му се прилага обикновеният (непретегленият) вариант на алгоритъма k-NN, където *k* е броят на точно съвпадащите с примера екземпляри.

По аналогичен начин можем да модифицираме k-NN алгоритъм за апроксимиране на непрекъснати целеви функции, заменяйки последния ред с:

$$\hat{f}(E) \leftarrow \frac{\sum_{i=1}^{u} w_i f(X_i)}{\sum_{i=1}^{k} w_i},$$

където w_i се изчислява по същия начин. Знаменателят в тази формула е константа, използвана за нормализиране на приноси от въвеждането на теглата (т.е. тя осигурява, че ако $f(X_i) = const$ за всички обучаващи примери, то и $\hat{f}(E) \leftarrow const$).

Описаната модификация на алгоритъма се нарича *алгоритъм за k най-близки съседи, претеглени по разстояние,* и представлява един много ефективен метод за решаване на разнообразни задачи. Той е устойчив към наличието на зашумени обучаващи примери и може с успех да се използва при достатъчно голям брой

обучаващи примери. Използването на претеглени гласове на k най-близки съседи позволява да неутрализира влияние на изолирани зашумени обучаващи примери.

Обърнете внимание, че след въвеждането на претеглянето по разстояние можем за класифициране на тестовия пример без всякакви проблеми да използваме и *всички* обучаващи примери, тъй като отдалечени примери няма да оказват значителен ефект на класификацията. Единственият недостатък – алгоритъмът ще работи прекалено бавно. Когато при класификацията на тестовия пример се използват всички обучаващи примери, алгоритмът се нарича *глобален*, а когато само най-близките му съседи – *локален*. Глобалният алгоритъм, приложен към апроксимацията на непрекъсната целевата функция, още е известен под името *методът на Шепард*.

12.5.4. Избор на оптимално *k*

Както се вижда от примери, изборът на k може да окаже съществено влияние върху точността на използваните методи за най-близки съседи в случая на апроксимацията на дискретната целева функция. В общия случай оптималният избор на *k* (от гледната точка на най-голяма предсказваща сила на алгоритъма) може да бъде направен чрез прилагане на една специална техника за *крос-валидацията*, наричана *„един-вън"* (leaving-one-out). Напомням, че крос-валидацията е приближен метод за оценяване на истинската грешка на някой класификационен алгоритъм на базата на известна (обучаваща) извадка от примери.

От извадка, съдържаща *n* примера, *n-1* примера се избират като обучаващо множество, след което класификационният алгоритъм се тества върху "останалия вън" един-единствен пример. Процедурата се повтаря *n* пъти, всеки път оставяйки "вън" по един пример. По този начин всеки пример от извадката се използва като тестов пример и всеки път всички останали примери се използват за неговата класификация. Класификационната грешка на алгоритъма се оценява чрез общия брой на грешките при класифициране на единичен пример, разделен на *n*. За да се получи добро приближение към истинската класификационна грешка на алгоритъма, описаната процедура трябва да бъде приложена към достатъчно голям брой *m* различни, случайно избрани извадки с размер *n* (обикновено *m* не е по-малък от 10).

Техниката "един-вън" за избор на оптималното *k* в алгоритмите на най-близките съседи се прилага по следния начин: След създаването на *m* обучаващи извадки с размер *n* към всяка от тях се прилага описаната процедура за крос-валидация, като се използва за класификация методът на най-близките съседи с избраната стойност на параметъра *k (k* = 1, 2, 3, ...). Избира се тази стойност на *k,* при която класификационната грешка, усреднена по всичките *m* извадки, е минимална.

Теоретичната оценка на коректността на методите k-NN показва, че при неограничен брой примери класификационната грешка на алгоритъма на един най-близък съсед не е по-лоша от двойната класификационна грешка на оптималния Бейсов класификатор. При голяма стойност на k класификационната грешка на алгоритъма k-NN в граничния случай съвпада със стойността на грешката на оптимален Бейсов класификатор.

Методите k-NN работят добре, ако използваните за описание на примерите атрибути са с добра предсказваща сила. В случая на нерелевантни данни, т. е. при използването на слабо информативни или излишни за класификацията атрибути, поведението на алгоритмите може значително да се влоши и подобряването на тяхната точност с нарастване на броя на обучаващите примери става много бавно. Неголеми подобрения могат да бъдат получени чрез избор на подходяща мярка за разстояние и нормализация на непрекъснати атрибути.