Класификация 2

Класификация и прогнозиране

- Задача
  - определяне на класове и принадлежността на обект към някой от класовете със задоволителна точност
- Принцип
  - всеки обект има атрибути
  - един от атрибутите е ключов
  - ключовият атрибут определя принадлежността към определен клас
- Модел
  - търси се модел, чрез който се намира стойността на ключовия атрибут, като функция на неключовите

Данни

- Обучаваща извадка (*training set*) за построяване на модела
- Тестова извадка (*test set*) за валидиране на модела
- Реални данни и прогнози при прилагане на модела

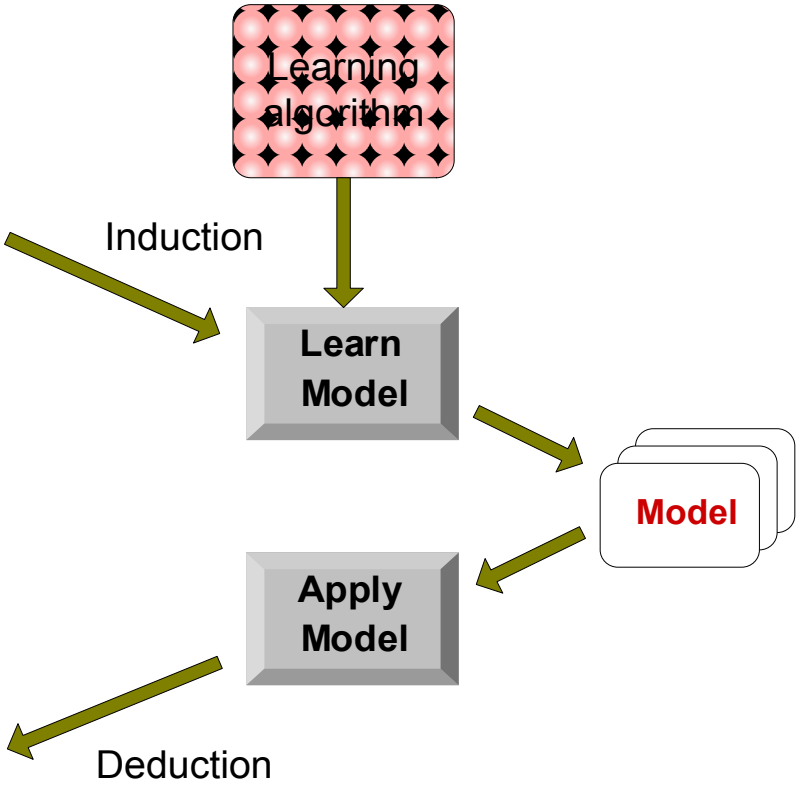
Пример

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



Методи и алгоритми за построяване на модел

- Дърво на решенията
- Вероятностни методи
- Правила if-then
- Невронни мрежи
- и др.

Вероятностни методи

Класификатори на Бейс

Пример

Обучаваща извадка D

Атрибути

age, income, student, rating, computer

Класове

C1: buys_computer = ‘yes’

C2: buys_computer = ‘no’

Данни за класифициране

X = (age <=30,
Income = medium,
Student = yes
Credit_rating = Fair
)

Каква е вероятността X да принадлежи на C1?

age	income	student	credit_rating	_comp
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

Класификация на Бейс

- Bayesian Classification
- По името на създателя на теоремата, послужила за основа на метода, Томас Бейс
  - теорема на Бейс
- Статистически метод
  - вероятностна прогноза
  - изчисляват се вероятностите за принадлежност към определени класове и се избира най-голямата
- Всеки нов пример в обучашащата извадка допълнително уточнява и настройва резултата
- Резултатите са съизмерими с резултатите на методите дърво на решенията и невронни класификатори

Математически апарат

- Теория на вероятностите
  - Условна вероятност $$P(C \mid A) = \frac{P(A,C)}{P(A)}$$ $$P(A \mid C) = \frac{P(A,C)}{P(C)}$$
  - Теорема на Бейс $$P(C \mid A) = \frac{P(A \mid C)P(C)}{P(A)}$$

Пример: условна вероятност

- Събитие (**A**): подхвърлена монета се обръща на “ези”, вместо на “тура” – началната вероятност $P(A) = 50\%$
- Доказателство (**B**): монетата се подхвърля няколко пъти за събиране на доказателства – вероятността за A се променя, примерно $P(A \mid B) = 70\%$

Теорема на Бейс

- Вероятността се разглежда като степен на очакване на събитие, хипотеза
- Теоремата свързва степените на очакване преди и след наличие на доказателство/свидетелство, *evidence*
- За събитието A и доказателството B е в сила следното:
  - $P(A)$ е предварителната, *prior probability*, степен на очакване да се случи A
  - $P(A \mid B)$ е последващата, *posterior probability*, степен на очакване да се случи A, ако вече се е случило доказателството B
  - $P(B \mid A) \ / \ P(B)$ е поддръжката, която B осигурява на A
- Тогава

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

posterior = likelihood x prior / evidence

Приложение

- Задача: за човек X, на *възраст=35 год.* и с *доходи=1200 лв. месечно,* да се предвиди дали ще си купи компютър или не
- Хипотеза H: човекът X ще си купи компютър
- Метод
  - P(H): вероятността един човек, независимо от възрастта и доходите си, да си купи компютър
  - P(H|**X**): вероятността човек с възраст и доходи, като X да си купи компютър (базирано на повече информация!)
  - P(**X**|H): вероятността човек, който си е купил компютър, да е на 35 години и да има доходи 1200 лв.
  - P(**X**): вероятността един клиент да е 35-годишни, с месечен доход 1200 лв.
- P(**X**), P(**X**|H) и P(H) се изчисляват от данните, P(H|**X**) – по теоремата на Бейс

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})} = P(\mathbf{X}|H) \times P(H) / P(\mathbf{X})$$

Приложение

- Ако **H** е хипотезата "записът X принадлежи на клас C"
  - **X** е доказателство, *evidence*
  - C е един от класовете, за който се търси принадлежност
- Тогава
  - P(H), *prior probability* – началната вероятност кой да е запис да принадлежи на C
  - P(**X**) - вероятността да има такъв запис, който принадлежи на C
  - P(**X**|H), *likelihood*, вероятността **X** да поддържа хипотезата
  - Класификацията трябва да определи P(H|**X**), *posteriori probability* – вероятността за сбъдване на H при условие, че е сбъднато X
- Предполага се, че данните са независими помежду си =>

Naïve Bayesian Classification

Алгоритъм

1. Нека D е обучаваща извадка, в която за всяко множество X, представено чрез n-D вектор на атрибутите A_1, A_2, ... $A_п$, $\mathbf{X} = (x_1, x_2, \ldots, x_n)$ е даден и етикетът на класа, на който множеството принадлежи
2. Нека класовете са *m* на брой: $C_1, C_2, \ldots, C_m$. Класификаторът трябва да постави X в този от класовете, за който постериорната вероятност е най-голяма, т.е. $\max P(C_i | \mathbf{X})$

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

3. Тъй като P(X) е константа, еднаква за всички класове, максимизира се само

$$P(\mathbf{X}|C_i)P(C_i)$$

Ако се приеме, че всички $P(C_i)$ са равновероятни, може да се максимизира само $P(X | C_i)$

Алгоритъм

4. Като се вземе предвид *наивното допускане*, че атрибутите не са взаимозависими, изчисленията се свеждат до

$$P(\mathbf{X}\,|\,C_i) = \prod_{k=1}^{n} P(x_k\,|\,C_i) = P(x_1\,|\,C_i)\times P(x_2\,|\,C_i)\times\ldots\times P(x_n\,|\,C_i)$$

$x_к$ е стойността на атрибута $A_к$ за извадката X

а) Ако A_k е дискретна величина, $P(x_k\,|\,C_i)$ е отношението на записите от клас C_i в D, в които атрибутът A_k има стойност x_k към всички записи от клас C_i в D $|\,C_{i,D}\,|$

б) Ако A_k е непрекъсната величина, $P(x_k\,|\,C_i)$ се изчислява със средното μ и стандартното отклонение σ при нормално Гаусово разпределение, където

$$g(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

и $$P(\mathbf{X}_k\,|\,C_i) = g(x_k,\mu_{C_i},\sigma_{C_i})$$

μ_{c_i} и σ_{c_i} са стойности за атрибута A_k във всички записи от класа C_i

5. За да се определи класът на X, $P(\mathbf{X}|C_i)P(C_i)$ се изчислява за всеки клас и се избира класът с най-голямата изчислена стойност

Пример

- **X = (age <= 30 , income = medium, student = yes, credit_rating = fair)**
- **P(C_i):** P(buys_computer = "yes") = 9/14 = 0.643
  P(buys_computer = "no") = 5/14= 0.357
- Изчисляване на **P(X | C_i)** за всеки атрибут и клас – C1 и C2

age	income	student	credit_rating	com
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

P(age = "<=30" | buys_computer = "yes") = 2/9 = 0.222
P(age = "<= 30" | buys_computer = "no") = 3/5 = 0.6
P(income = "medium" | buys_computer = "yes") = 4/9 = 0.444
P(income = "medium" | buys_computer = "no") = 2/5 = 0.4
P(student = "yes" | buys_computer = "yes) = 6/9 = 0.667
P(student = "yes" | buys_computer = "no") = 1/5 = 0.2
P(credit_rating = "fair" | buys_computer = "yes") = 6/9 = 0.667
P(credit_rating = "fair" | buys_computer = "no") = 2/5 = 0.4

P(X | C_i) : P(X | buys_computer = "yes") = 0.222 x 0.444 x 0.667 x 0.667 = 0.044
P(X | buys_computer = "no") = 0.6 x 0.4 x 0.2 x 0.4 = 0.019

P(X | C_i)*P(C_i) : P(X | buys_computer = "yes") * P(buys_computer = "yes") = 0.028
P(X | buys_computer = "no") * P(buys_computer = "no") = 0.007

Следователно, X принадлежи на класа

("buys_computer = yes")

Друг пример

- Като се знае, че
  - менингит причинява схващане на врата през 50% от времето
  - предварителната вероятност пациент да има менингит е 1/50,000
  - предварителната вероятност някой пациент да има схванат врат е 1/20
- каква е вероятността един пациент да има менингит, ако има схванат врат?

$$P(M \mid S) = \frac{P(S \mid M)P(M)}{P(S)} = \frac{0.5 \times 1/50000}{1/20} = 0.0002$$

Друг пример

Name	Give Birth	Can Fly	Live in Water	Have Legs	Class
human	yes	no	no	yes	mammals
python	no	no	no	no	non-mammals
salmon	no	no	yes	no	non-mammals
whale	yes	no	yes	no	mammals
frog	no	no	sometimes	yes	non-mammals
komodo	no	no	no	yes	non-mammals
bat	yes	yes	no	yes	mammals
pigeon	no	yes	no	yes	non-mammals
cat	yes	no	no	yes	mammals
leopard shark	yes	no	yes	no	non-mammals
turtle	no	no	sometimes	yes	non-mammals
penguin	no	no	sometimes	yes	non-mammals
porcupine	yes	no	no	yes	mammals
eel	no	no	yes	no	non-mammals
salamander	no	no	sometimes	yes	non-mammals
gila monster	no	no	no	yes	non-mammals
platypus	no	no	no	yes	mammals
owl	no	yes	no	yes	non-mammals
dolphin	yes	no	yes	no	mammals
eagle	no	yes	no	yes	non-mammals

D: обучаваща извадка (20)

Класове M: mammals (7)

N: non-mammals (13)

A: атрибути (4)

Give Birth	Can Fly	Live in Water	Have Legs	Class
yes	no	yes	no	?

$$P(A \mid M) = \frac{6}{7} \times \frac{6}{7} \times \frac{2}{7} \times \frac{2}{7} = 0.06$$

$$P(A \mid N) = \frac{1}{13} \times \frac{10}{13} \times \frac{3}{13} \times \frac{4}{13} = 0.0042$$

$$P(A \mid M)P(M) = 0.06 \times \frac{7}{20} = 0.021$$

$$P(A \mid N)P(N) = 0.004 \times \frac{13}{20} = 0.0027$$

P(A|M)P(M) > P(A|N)P(N)

=> Mammals

Друг пример

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

D: обучаваща извадка (10)

Класове Yes (3)

No: (7)

A: атрибути (4)

Принадлежност към клас $P(C) = N_c/N$

$P(No) = 7/10$
$P(Yes) = 3/10$

За дискретни атрибути:

$$P(A_i \mid C_k) = |A_{ik}| / N_c$$

където $|A_{ik}|$ е броят на извадките, съдържащи A_i и принадлежащи на C_k

$P(Status=Married|No) = 4/7$
$P(Refund=Yes|Yes)=0$

Tid	Refund	Marital Status	Taxable Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

- За непрекъснати атрибути
  - нормално разпределение

$$P(A_i \mid c_j) = \frac{1}{\sqrt{2\pi\sigma_{ij}^2}} e^{-\frac{(A_i-\mu_{ij})^2}{2\sigma_{ij}^2}}$$

за всяка двойка (A_i, c_i)

- За (Income, Class=No):
  - средно аритметично = 110
  - стандартно отклонение = 2975

$$P(Income = 120 \mid No) = \frac{1}{\sqrt{2\pi}(54.54)} e^{\frac{(120-110)^2}{2(2975)}} = 0.0072$$

Приложение

$$X = (\text{Refund} = \text{No}, \text{Married}, \text{Income} = 120\text{K})$$

naive Bayes Classifier:

P(Refund=Yes|No) = 3/7
P(Refund=No|No) = 4/7
P(Refund=Yes|Yes) = 0
P(Refund=No|Yes) = 1
P(Marital Status=Single|No) = 2/7
P(Marital Status=Divorced|No)=1/7
P(Marital Status=Married|No) = 4/7
P(Marital Status=Single|Yes) = 2/7
P(Marital Status=Divorced|Yes)=1/7
P(Marital Status=Married|Yes) = 0

For taxable income:
If class=No: sample mean=110
sample variance=2975
If class=Yes: sample mean=90
sample variance=25

- P(X|Class=No) = P(Refund=No|Class=No)
  × P(Married| Class=No)
  × P(Income=120K| Class=No)
  = 4/7 × 4/7 × 0.0072 = 0.0024

- P(X|Class=Yes) = P(Refund=No| Class=Yes)
  × P(Married| Class=Yes)
  × P(Income=120K| Class=Yes)
  = 1 × 0 × 1.2 × 10^{-9} = 0

Тъй като P(X|No)P(No) > P(X|Yes)P(Yes),

следователно P(No|X) > P(Yes|X)

=> Class = No

Корекция на Лаплас

- Ако някой клас е празен, няма данни за представянето му, условната вероятност за принадлежност към този клас ще е нула и целият израз за изчисляване на вероятноститите ще приеме стойност нула
- Корекция: добавя се 1 към размерите на класовете

$$\text{Original} : P(A_i \mid C) = \frac{N_{ic}}{N_c}$$

$$\text{Laplace} : P(A_i \mid C) = \frac{N_{ic} + 1}{N_c + c}$$

$$\text{m-estimate} : P(A_i \mid C) = \frac{N_{ic} + mp}{N_c + m}$$

c: number of classes

p: prior probability

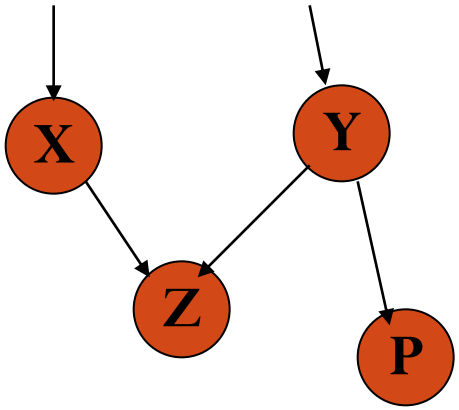
m: parameter

Обобщение

- Предимства на метода
  - лесен за прилагане
  - добри резултати в повечето случаи
- Недостатъци
  - неточност, поради допускането за независимост на атрибутите:
  - на практика, такава зависимост съществува, напр. при пациенти и диагнози и др.

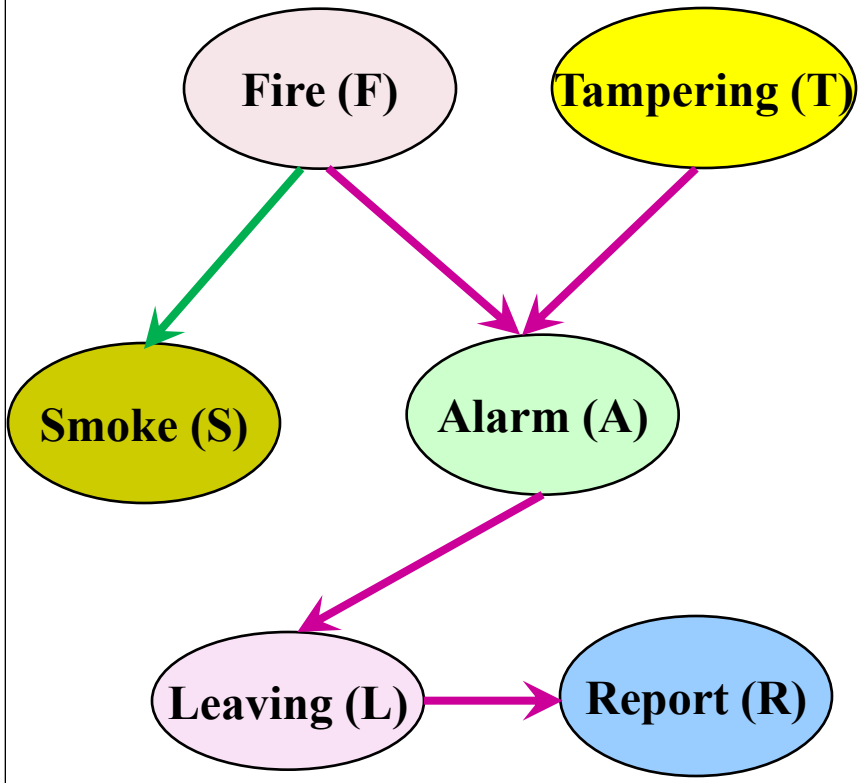
Bayesian Belief Networks

- Bayesian belief network (Bayesian network, probabilistic network)
  - представя зависимости между променливи
  - позволява свързани условни вероятностни разпределения
- Два компонента
  - ацикличен граф (структура) на влияния
  - набор от таблици на условни вероятности *conditional probability tables* (CPT)



- Nodes: променливи
- Links: зависимости

Пример



CPT: Conditional Probability Tables

Fire	Smoke	$\Theta_{s\mid f}$
True	True	.90
False	True	.01

Fire	Tampering	Alarm	$\Theta_{a\mid f,t}$
True	True	True	.5
True	False	True	.99
False	True	True	.85
False	False	True	.0001

CPT показва условната вероятност за всяка възможна комбинация от стойности на X, произтичаща от родителите

$$P(x_1,\ldots,x_n)=\prod_{i=1}^{n} P(x_i \mid Parents(x_i))$$

Построяване на мрежи

- Субективно - идентификация на причинно-следствени връзки
- Синтез от други спецификации – блок-диаграми, потоци от данни и др.
- Самообучение от данните -

D. Heckerman. A Tutorial on Learning with Bayesian Networks. In *Learning in Graphical Models,* M. Jordan, ed. MIT Press, 1999.

Правила

IF … THEN …

Rule-Based Classifier

- Класифициране на записи чрез прилагане на правила "if…then…"

  R: IF *age* = youth AND *student* = yes THEN *buys_computer* = yes

- Правило: $(Condition) \rightarrow y$
  - където
    - *Condition* логически израз от атрибути (конюнкция)
    - y е етикет на клас
  - *LHS*: rule antecedent
  - *RHS*: rule consequent
  - Примери
    - (Blood Type=Warm) $\wedge$ (Lay Eggs=Yes) $\rightarrow$ Birds
    - (Taxable Income < 50K) $\wedge$ (Refund=Yes) $\rightarrow$ Evade=No

Пример

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
human	warm	yes	no	no	mammals
python	cold	no	no	no	reptiles
salmon	cold	no	no	yes	fishes
whale	warm	yes	no	yes	mammals
frog	cold	no	no	sometimes	amphibians
komodo	cold	no	no	no	reptiles
bat	warm	yes	yes	no	mammals
pigeon	warm	no	yes	no	birds
cat	warm	yes	no	no	mammals
leopard shark	cold	yes	no	yes	fishes
turtle	cold	no	no	sometimes	reptiles
penguin	warm	no	no	sometimes	birds
porcupine	warm	yes	no	no	mammals
eel	cold	no	no	yes	fishes
salamander	cold	no	no	sometimes	amphibians
gila monster	cold	no	no	no	reptiles
platypus	warm	no	no	no	mammals
owl	warm	no	yes	no	birds
dolphin	warm	yes	no	yes	mammals
eagle	warm	no	yes	no	birds

R1: (Give Birth = no) $\wedge$ (Can Fly = yes) $\rightarrow$ Birds

R2: (Give Birth = no) $\wedge$ (Live in Water = yes) $\rightarrow$ Fishes

R3: (Give Birth = yes) $\wedge$ (Blood Type = warm) $\rightarrow$ Mammals

R4: (Give Birth = no) $\wedge$ (Can Fly = no) $\rightarrow$ Reptiles

R5: (Live in Water = sometimes) $\rightarrow$ Amphibians

Приложение

- Правило *r* покрива извадка **x**, ако атрибутите на извадката удовлетворяват условието

R1: (Give Birth = no) ∧ (Can Fly = yes) → Birds

R2: (Give Birth = no) ∧ (Live in Water = yes) → Fishes

R3: (Give Birth = yes) ∧ (Blood Type = warm) → Mammals

R4: (Give Birth = no) ∧ (Can Fly = no) → Reptiles

R5: (Live in Water = sometimes) → Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
hawk	warm	no	yes	no	?
grizzly bear	warm	yes	no	no	?

Според правило R1 hawk => Bird (покрива се)

Според правило R3 grizzly bear => Mammal (покрива се)

Свойства на правило

- Покритие
  - Частта от записи, които удовлетворяват условието на правилото
  - n_{covers} = брой записи, покрити от R
- Точност
  - Частта от записи, които удовлетворяват и условието, и следствието от правилото
  - $n_{correct}$ = брой записи, коректно класифицирани от R

$$coverage(R) = n_{covers} / |D|$$
$$accuracy(R) = n_{correct} / n_{covers}$$

Tid	Refund	Marital Status	Taxable Income	Class
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

(Status=Single) → No

Coverage = 40%, Accuracy = 50%

Примери

R1: (Give Birth = no) ∧ (Can Fly = yes) → Birds

R2: (Give Birth = no) ∧ (Live in Water = yes) → Fishes

R3: (Give Birth = yes) ∧ (Blood Type = warm) → Mammals

R4: (Give Birth = no) ∧ (Can Fly = no) → Reptiles

R5: (Live in Water = sometimes) → Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
lemur	warm	yes	no	no	?
turtle	cold	no	no	sometimes	?
dogfish shark	cold	yes	no	yes	?

lemur – правило R3 => mammal

turtle – правила R4 и R5

dogfish shark – не се покрива от нито едно правило

Видове класификатори

- Взаимно-изключващи се правила
  - всички правила са независими едно от друго
  - всеки запис се покрива от най-много едно правило

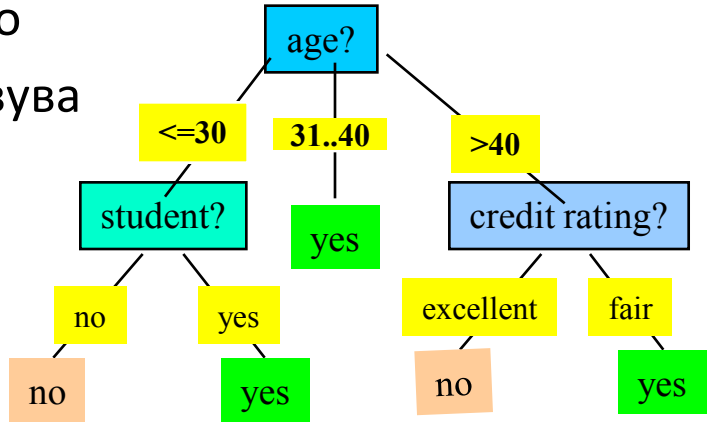
- Изчерпателни правила *Exhaustive rules*
  - включват всички възможни комбинации от стойности на атрибутите
  - всеки запис се покрива от поне едно правило

Построяване на правила

- Директни методи
  - извеждане от данните
  - методи: RIPPER, CN2, Holte's 1R
- Индиректни методи
  - извеждане от други класификационни модели (напр. дърво на решенията, невронна мрежа и др.)
  - метод: C4.5rules

Правила и Дърво на решенията

- Едно правило за всеки път от корена до листо
- Всяка двойка атрибут-стойност по пътя образува една конюнкция
- Всяко листо съдържа прогноза за клас
- Правилата са взаимно изключващи се и изчерпващи
- По-лесни за разбиране



- Пример: извеждане на правила от дървото *buys_computer*

IF *age* = young AND *student* = *no* THEN *buys_computer* = *no*

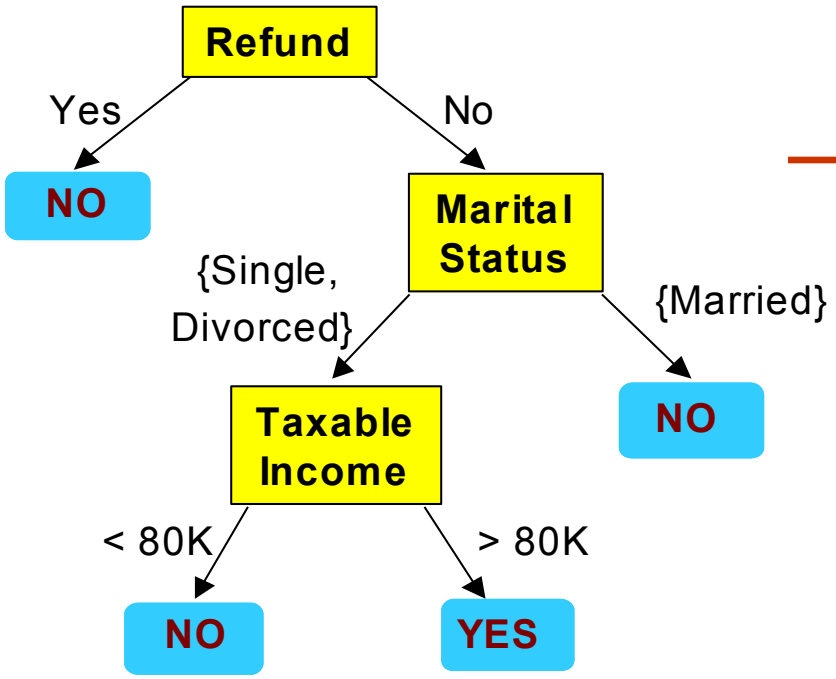
IF *age* = young AND *student* = *yes* THEN *buys_computer* = *yes*

IF *age* = mid-age THEN *buys_computer* = *yes*

IF *age* = old AND *credit_rating* = *excellent* THEN *buys_computer* = *no*

IF *age* = old AND *credit_rating* = *fair* THEN *buys_computer* = *yes*

Друго дърво на решенията и правила



Classification Rules

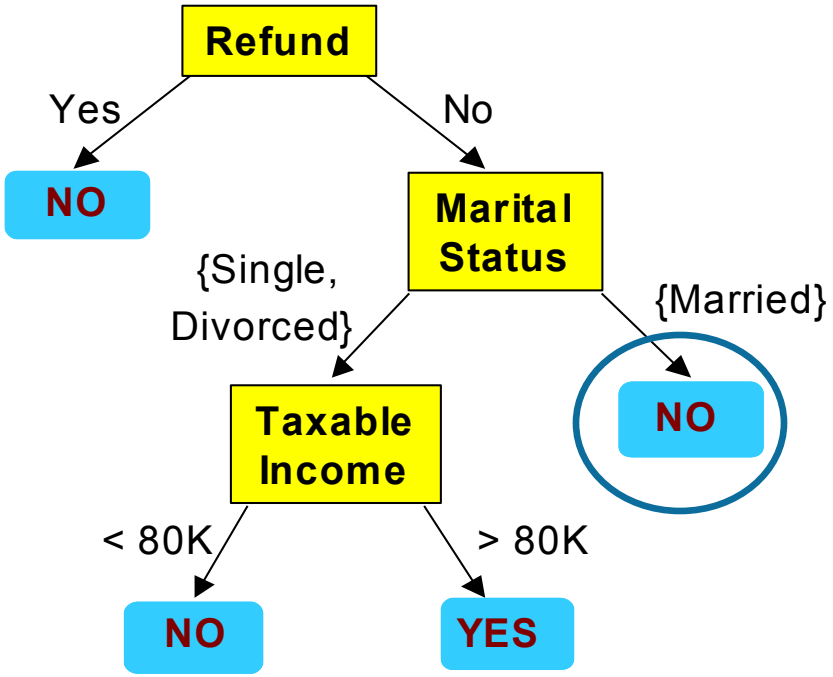
(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes

(Refund=No, Marital Status={Married}) ==> No

Опростяване на правилата



Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Начално правило: $(\text{Refund}=\text{No}) \wedge (\text{Status}=\text{Married}) \rightarrow \text{No}$

Опростено правило: $(\text{Status}=\text{Married}) \rightarrow \text{No}$

Ефект от опростяването

- Правилата не са взаимно изключващи се
  - Един запис – много правила
  - Решение: подредено множество правила

- Правилата не са изчерпателни
  - Един запис – няма правило
  - Решение: подразбиращ се клас

Подредено множество от правила

- Подредба по приоритет
  - decision list

R1: (Give Birth = no) ∧ (Can Fly = yes) → Birds

R2: (Give Birth = no) ∧ (Live in Water = yes) → Fishes

R3: (Give Birth = yes) ∧ (Blood Type = warm) → Mammals

R4: (Give Birth = no) ∧ (Can Fly = no) → Reptiles

R5: (Live in Water = sometimes) → Amphibians

Name	Blood Type	Give Birth	Can Fly	Live in Water	Class
turtle	cold	no	no	sometimes	?

Схеми за подреждане

- Rule-based ordering
  - Според качеството на правилата
- Class-based ordering
  - Правила от един клас се подреждат заедно

Rule-based Ordering

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes

(Refund=No, Marital Status={Married}) ==> No

Class-based Ordering

(Refund=Yes) ==> No

(Refund=No, Marital Status={Single,Divorced}, Taxable Income<80K) ==> No

(Refund=No, Marital Status={Married}) ==> No

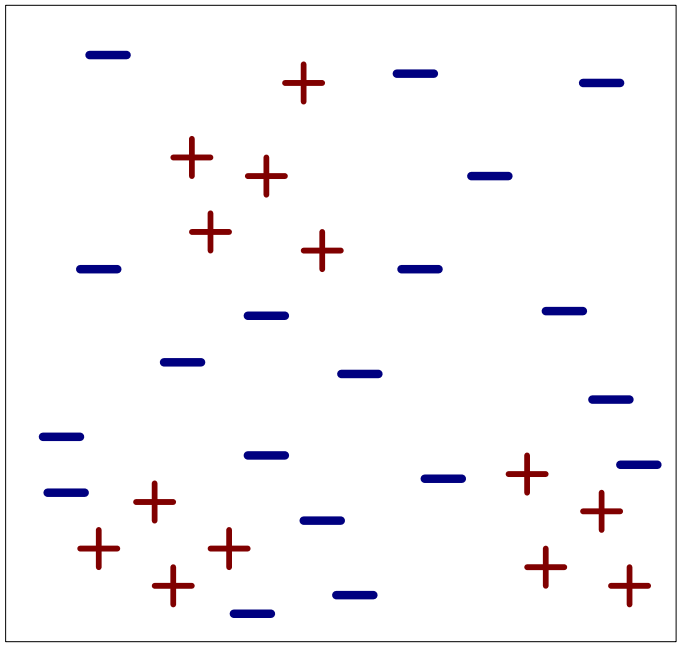
(Refund=No, Marital Status={Single,Divorced}, Taxable Income>80K) ==> Yes

Директен метод за извеждане на правила

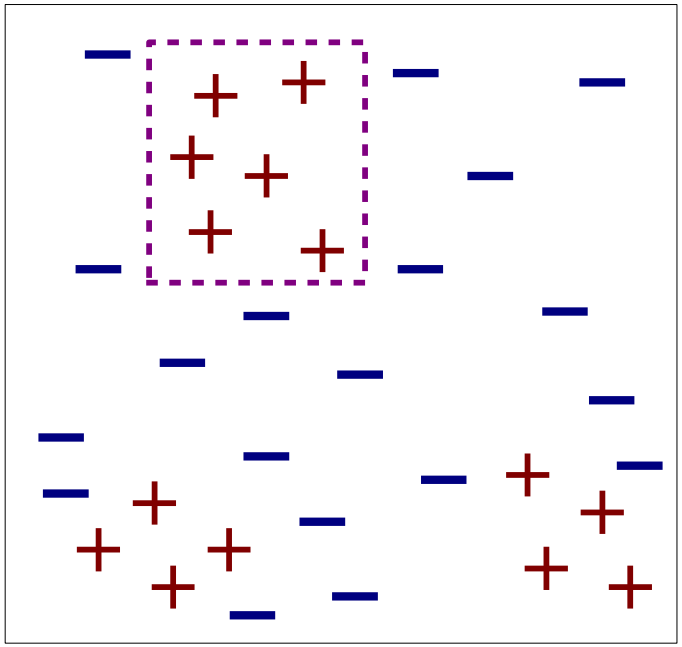
Последователно покриване

1. Започва се с празно множество от правила
2. Добавя се правило чрез функция Learn-One-Rule
3. Изтрива се подмножеството от обучаващата извадка, покрито от новото правило
4. Повтарят се стъпки (2) и (3) до срещане на критерий за край

Пример

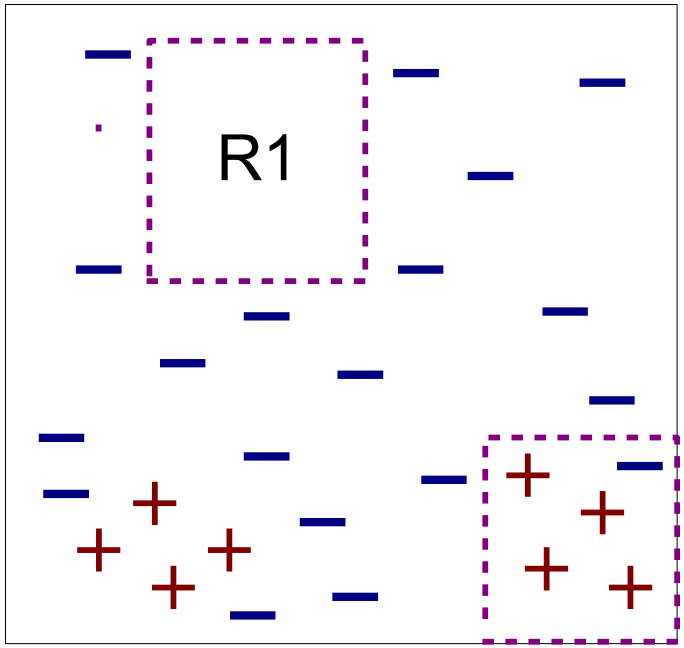


(i) Original Data

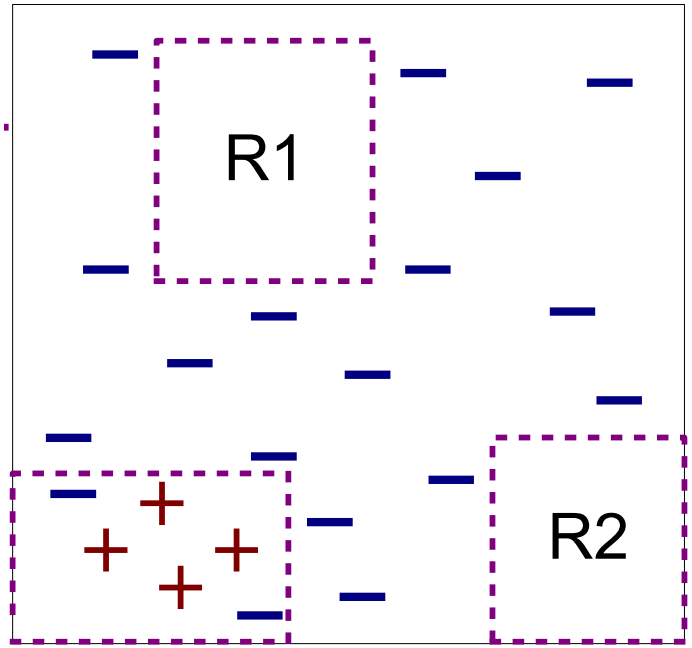


(ii) Step 1

Пример



(iii) Step 2



(iv) Step 3

Learn-One-Rule

- Започва се с най-общото възможно правило

  condition = empty
- Добавят се нови атрибути към условието
  - избира се този, който най-много подобрява качеството на правилото
- Оценка на качеството
  - max покритие coverage и точност accuracy

Мерки за оценяване на правилата

- Accuracy $= \frac{n_c}{n}$
- Laplace $= \frac{n_c + 1}{n + k}$
- M-estimate $= \frac{n_c + kp}{n + k}$

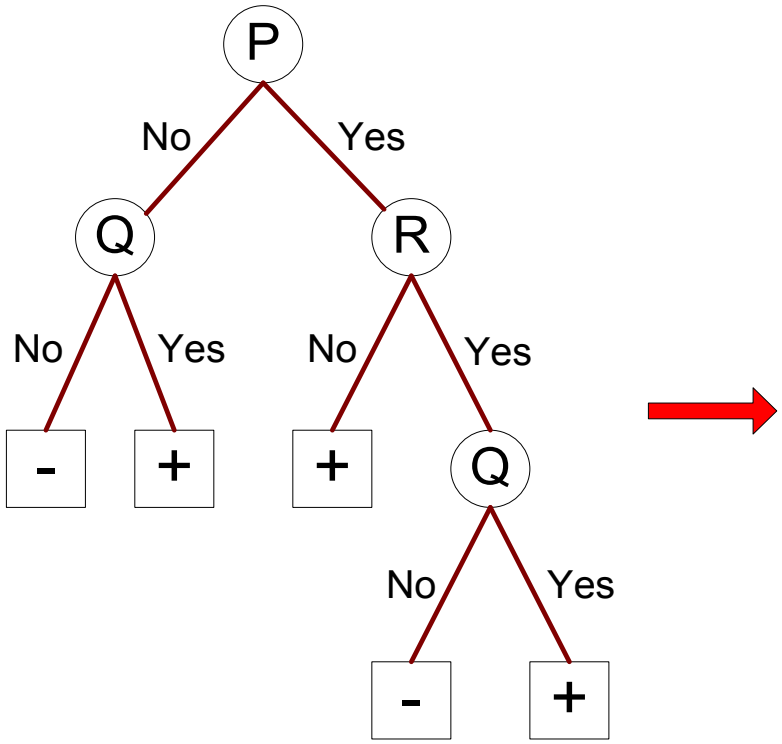
n : Number of instances covered by rule

n_c : Number of instances covered by rule

k : Number of classes

p : Prior probability

Индиректни методи за извличане на правила



Rule Set

r1: (P=No,Q=No) ==> -
r2: (P=No,Q=Yes) ==> +
r3: (P=Yes,R=No) ==> +
r4: (P=Yes,R=Yes,Q=No) ==> -
r5: (P=Yes,R=Yes,Q=Yes) ==> +

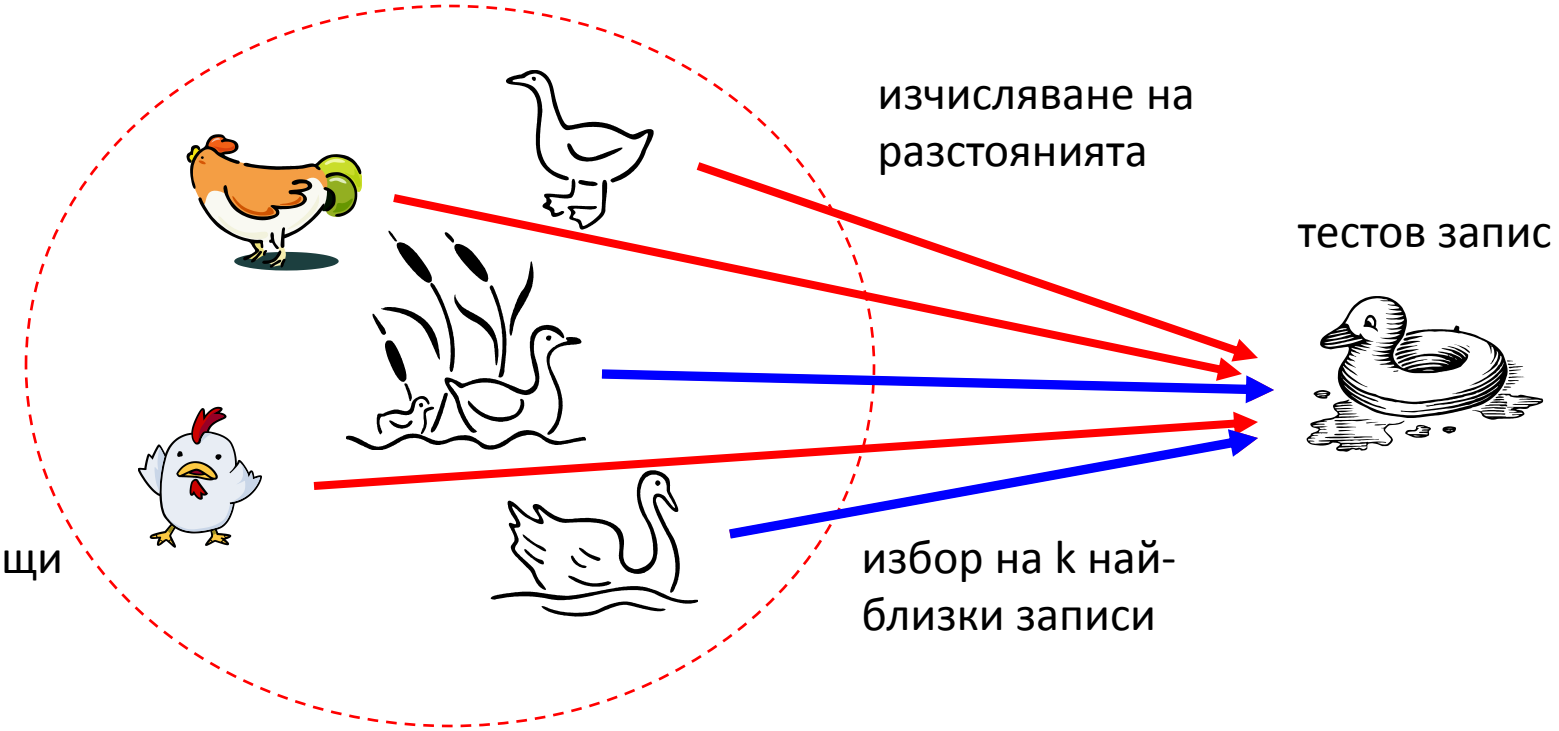
Предимства на методите с правила

- Бързи
- Лесни за интерпретиране
- Лесни за генериране
- Дават добри резултати, близки до дърво на решенията

Най-близките съседи

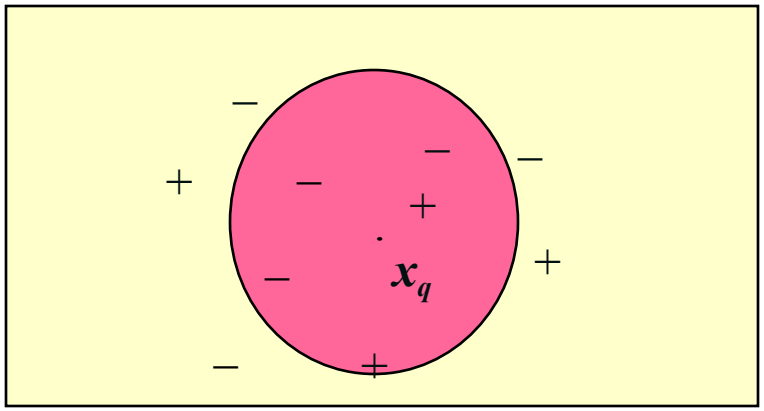
Най-близките съседи

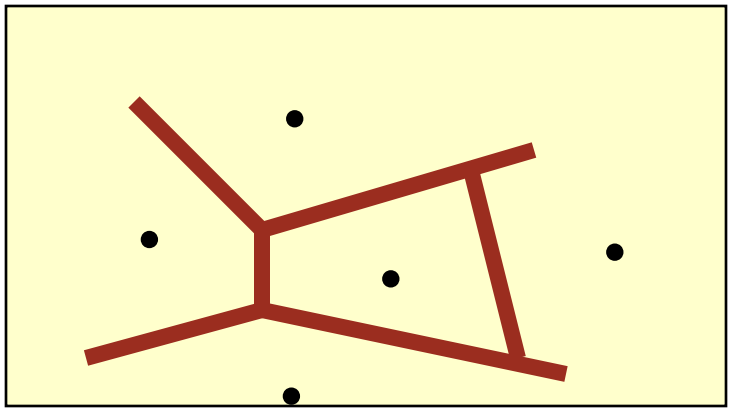
- Основна идея
  - Ако ходи като патица, плува като патица и квака като патица, най-вероятно е патица



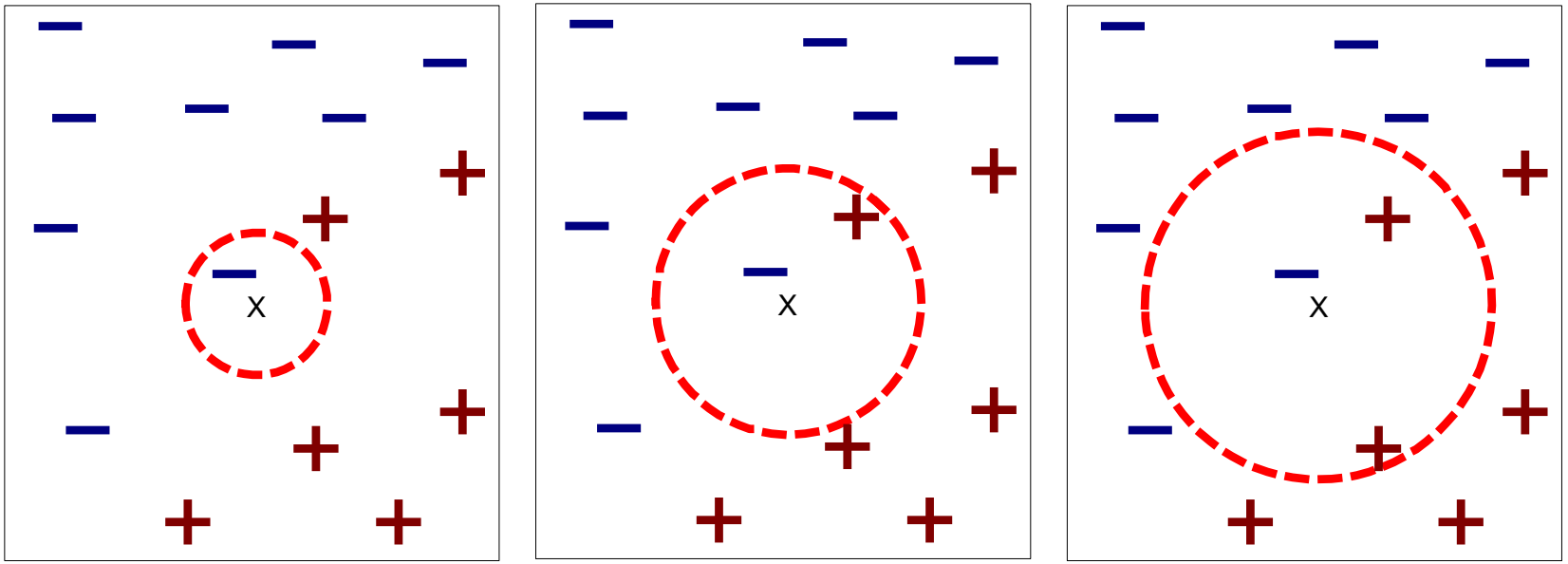
Алгоритъм

- Всички записи могат да бъдат представени като точки в n-D пространство
- Най-близкият съсед се дефинира в термините на Евклидовото разстояние между две точки $\mathrm{dist}(\mathbf{X}_1, \mathbf{X}_2)$
- Функцията за избор може да бъде дискретна или непрекъсната





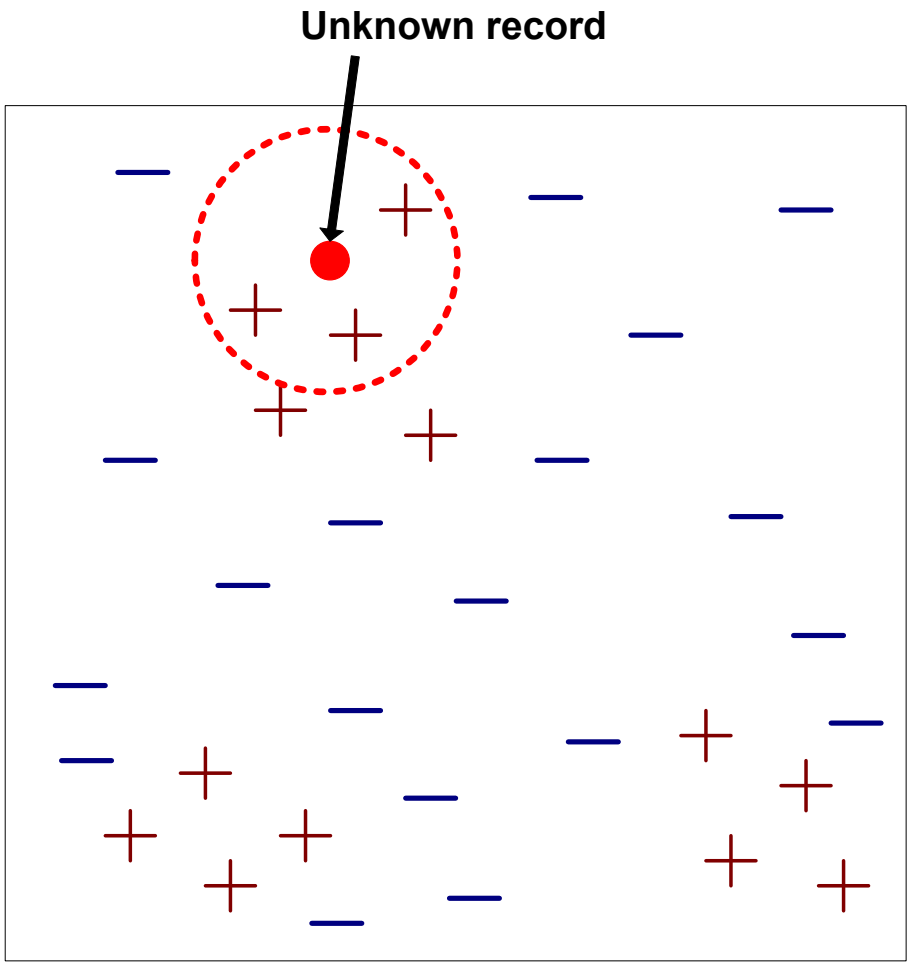
Дефиниция на най-близък съсед



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K-най-близки съседи на запис x са тези k точки, отстоянието на x до които е най-малко

Класификация



- Вход
  - обучаваща извадка
  - метрика за отстояние
  - брой на най-близките съседи, които да се изведат като кандидати - стойност *k*
- Процедура
  - изчисляване на отстоянията
  - избор на *k* най-близки съседи
  - определяне на най-вероятния клас

Класификация

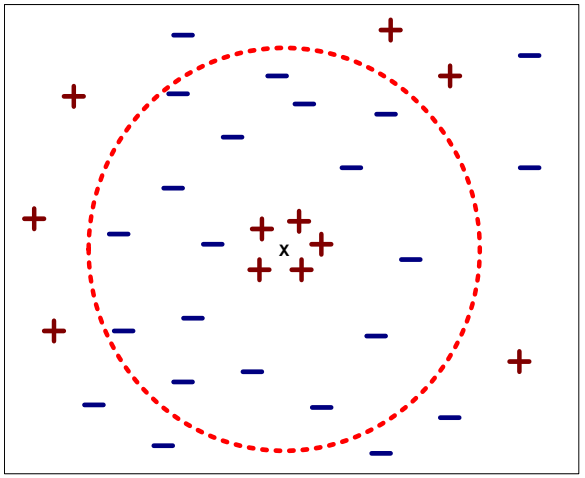
- Изчисляване на растоянието между две точки
  - Евклидово разстояние

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$

  - категорийните атрибути се сравняват логически
    - d(p,q)=0, ако съответните стойности за равни
    - d(p,q)=1, в противен случай
- Избор на клас на принадлежност
  - вот / сравняване на k най-близки съседи
  - претегляне на вота
    - поставят се тегла на значението на съседите, според отстоянието им до x_q
    - тегловен коефициент $w = 1/d^2$

Допълнителни съображения

- Избор на k:
  - Ако k е много малко, опасност от шум
  - Ако k е много голямо, близките съседи могат да съдържат точки от друг клас
  - Проверява се експериментално, започвайки от k=1



Допълнителни съображения

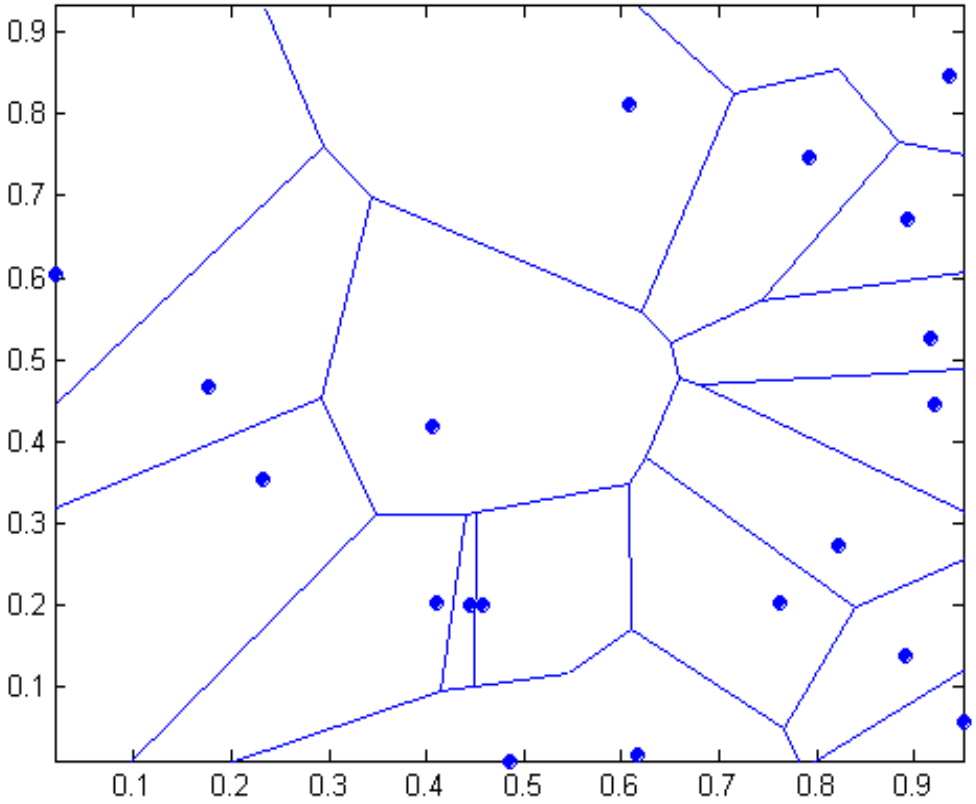
- Скалиране и нормализация на атрибутите
  - Ако има голяма разлика в метриките на атрибутите се налага нормализация
  - Примери
    - височината на човек варира от 1.5m до1.8m
    - теглото може да варира от 40 кг до 140 кг
    - доходът на човек може да варира от 250 лв. до 25000 лв
    - ако се обработват заедно, следва да са от един порядък
- Ако има липсващи стойности, заменят се с такива, които биха довели до най-голямо разстояние

Допълнителни съображения

- k-NN за прогнозиране на реални стойности - връща средните стойности от k съседи
- Усредняване на стойностите на k-съседи за намаляване на шума
- За избягване на доминирането на част от атрибутите на запис върху други, някои атрибути могат да бъдат изключени от изчисленията

K=1

Диаграма на Вороной



Пример: PEBLS

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Parallel Examplar-Based Learning System (Cost & Salzberg)

Разстояния между атрибути с номинални стойности:

d(Single,Married)

= | 2/4 – 0/4 | + | 2/4 – 4/4 | = 1

d(Single,Divorced)

= | 2/4 – 1/2 | + | 2/4 – 1/2 | = 0

d(Married,Divorced)

= | 0/4 – 1/2 | + | 4/4 – 1/2 | = 1

d(Refund=Yes,Refund=No)

= | 0/3 – 3/7 | + | 3/3 – 4/7 | = 6/7

Class	Marital Status		
	Single	Married	Divorced
Yes	2	0	1
No	2	4	1

Class	Refund	
	Yes	No
Yes	0	3
No	3	4

$$d(V_1, V_2) = \sum_i \left| \frac{n_{1i}}{n_1} - \frac{n_{2i}}{n_2} \right|$$

Недостатъци на метода

- Не се построява модел
- Относително скъпо класифициране на неизвестни записи
  - по отношение на изчислителните операции
- “Мързеливи класификатори”
  - в сравнение с дърво на решенията и система от правила