Procesy i systemy Business Intelligence

Klasyfikacja

Wprowadzenie

Celem procesu klasyfikacji jest znalezienie ogólnego modelu podziału zbioru predefiniowanych klas obiektów na podstawie pewnego zbioru danych historycznych, a następnie, zastosowanie odkrytego modelu do predykcji klasy nowego obiektu, dla którego klasa nie jest znana

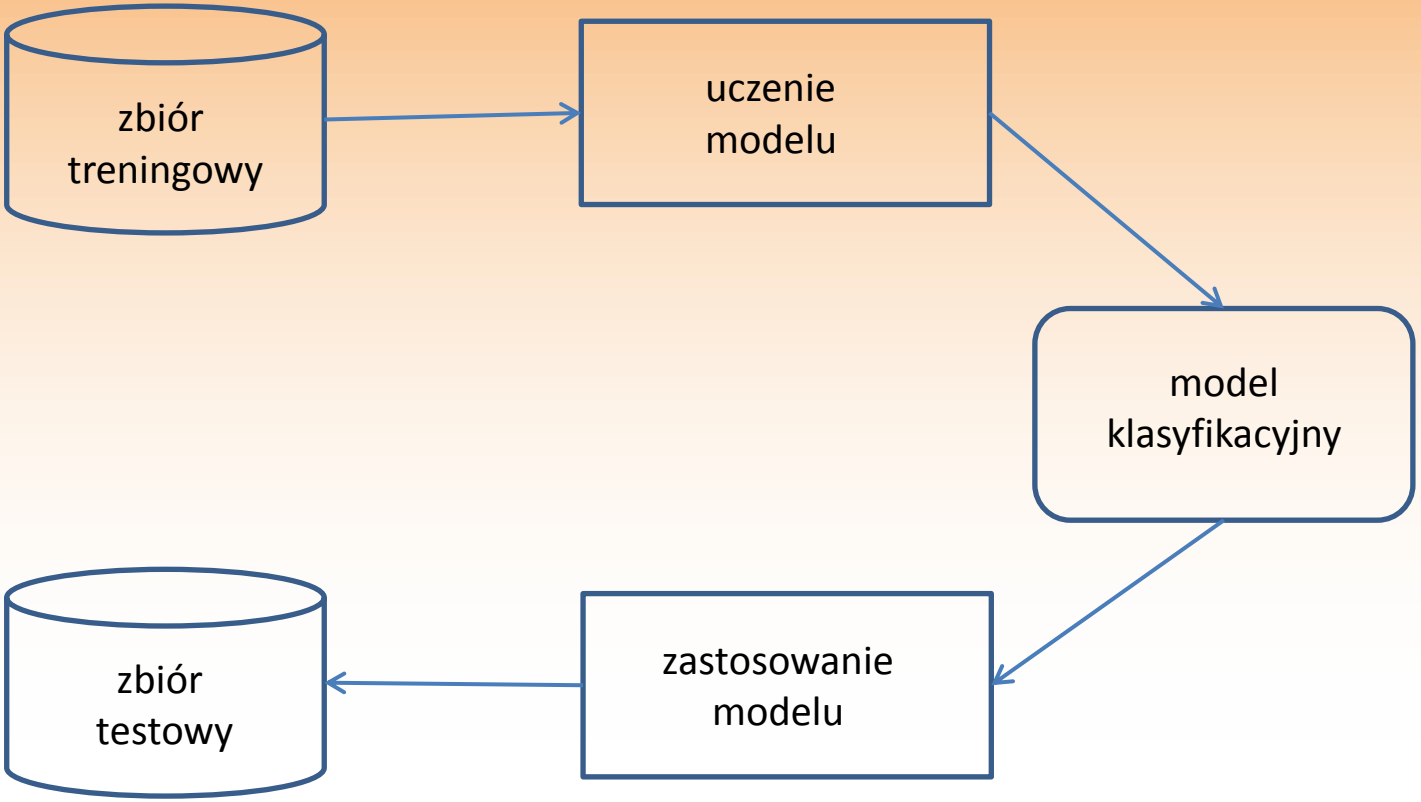
Klasyfikacja

- Dane wejściowe: treningowy zbiór rekordów D (danych, przykładów, obiektów, obserwacji, próbek, wektorów cech), będących listą wartości atrybutów warunkowych A_1, A_2,… , A_n(tzw. deskryptorów lub atrybutów opisowych) i wybranego atrybutu decyzyjnego C (ang. class label attribute)
- Dane wyjściowe: model (klasyfikator), przydziela każdemu rekordowi wartość atrybutu decyzyjnego w oparciu o wartości pozostałych atrybutów (deskryptorów)

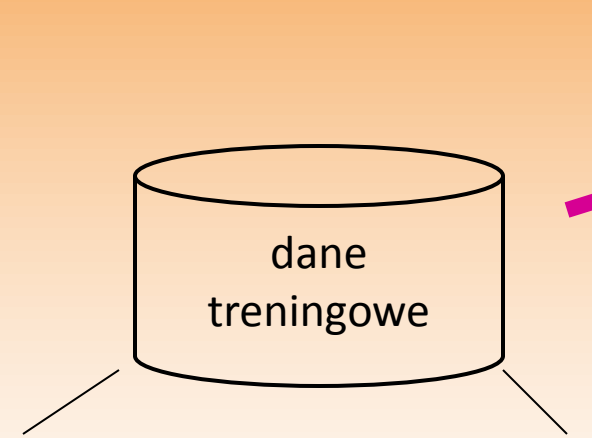
Czym jest klasyfikacja? (2)

- Zbiór dostępnych krotek (przykładów, obserwacji, próbek) dzielimy, w ogólności, na dwa zbiory: zbiór treningowy i zbiór testowy
- Model klasyfikacyjny (klasyfikator) jest budowany dwu-etapowo:
  - **Uczenie** (trening) – klasyfikator jest budowany w oparciu o zbiór treningowy danych
  - **Testowanie** – trafność (jakość) klasyfikatora jest weryfikowana w oparciu o zbiór testowy danych

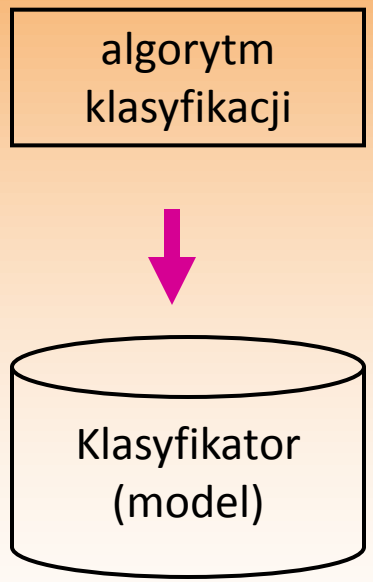
Czym jest klasyfikacja? (3)



Uczenie

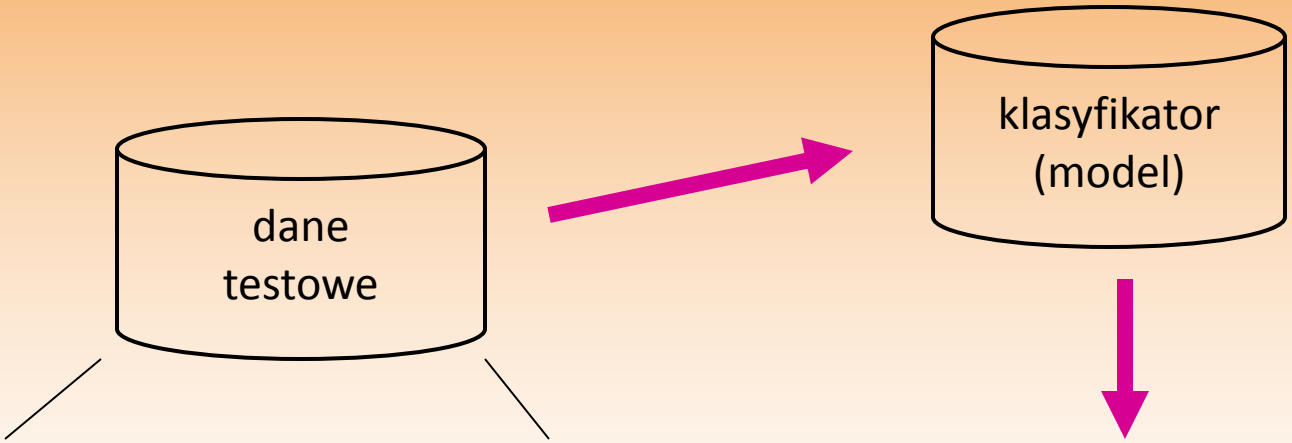


Wiek	Typ_sam	Ryzyko
20	Combi	High
18	Sports	High
40	Sports	High
50	Family	Low
35	Minivan	Low
30	Combi	High
32	Family	Low
40	Combi	Low



if Wiek < 31
or Typ_sam = „Sports”
then Ryzyko = High

Testowanie

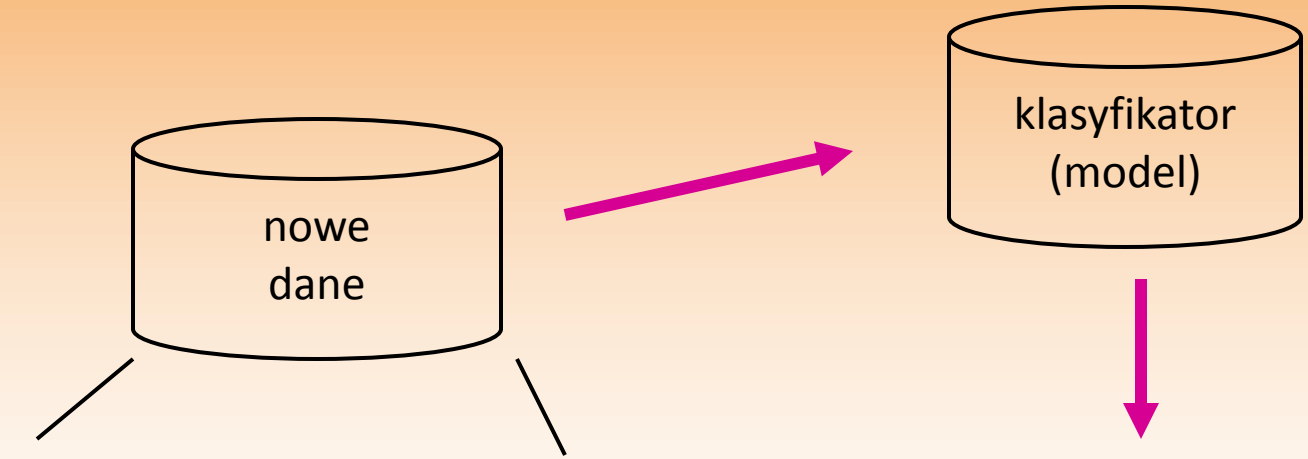


Wiek	Typ_sam	Ryzyko
27	Sports	High
34	Family	Low
66	Family	High
44	Sports	High

Ryzyko
High
Low
Low
High

Trafność= 3/4 = 75%

Klasyfikacja

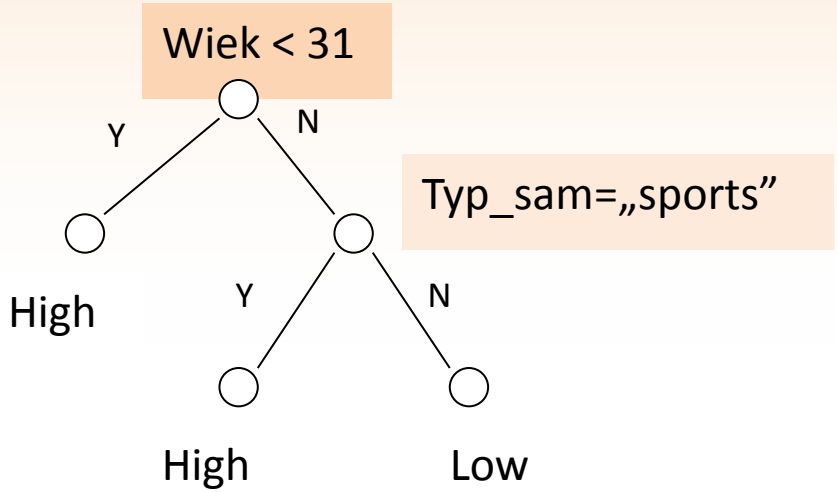


Wiek	Typ_sam	Ryzyko
27	Sports	
34	Minivan	
55	Family	
34	Sports	

Ryzyko
High
Low
Low
High

Klasyfikacja przez indukcje drzew decyzyjnych (1)

- Drzewo decyzyjne jest grafem o strukturze drzewiastej, gdzie
  - każdy wierzchołek wewnętrzny reprezentuje test na atrybucie (atrybutach),
  - każdy luk reprezentuje wynik testu,
  - każdy liść reprezentuje pojedynczą klasę lub rozkład wartości klas



Klasyfikacja przez indukcje drzew decyzyjnych (2)

- Drzewo decyzyjne rekurencyjnie dzieli zbiór treningowy na partycje do momentu, w którym każda partycja zawiera dane należące do jednej klasy, lub, gdy w ramach partycji dominują dane należące do jednej klasy
- Każdy wierzchołek wewnętrzny drzewa zawiera tzw. **punkt podziału** (ang. **split point),** którym jest test na atrybucie, który dzieli zbiór danych na partycje
- Drzewo decyzyjne jest konstruowane w dwóch krokach:
  - Krok 1: Konstrukcja drzewa decyzyjnego w oparciu o zbiór treningowy
  - Krok 2: Przycinanie drzewa w celu poprawy dokładności, interpretowalności i uniezależnienia się od efektu przetrenowania

Kryteria oceny podziału

- Indeks Giniego (Gini indeks) (algorytmy CART, SPRINT)
  - Wybieramy atrybut, który minimalizuje indeks Giniego
- Zysk informacyjny (algorytmy ID3, C4.5)
  - Wybieramy atrybut, który maksymalizuje redukcję entropii
- Indeks korelacji $\chi 2$ (algorytm CHAID)
  - Mierzymy korelację pomiędzy każdym atrybutem i każdą klasą (wartością atrybutu decyzyjnego)
  - Wybieramy atrybut o maksymalnej korelacji

Zysk infomacyjny (1)

- Algorytm ID3 - metoda wyboru punktu podziału SS wykorzystuje do wyboru atrybutu podziałowego miarę **zysku informacyjnego** (ang. information gain)
- Jako atrybut testowy (aktualny wierzchołek drzewa decyzyjnego) wybieramy atrybut o największym zysku informacyjnym (lub największej redukcji entropii)
- Atrybut testowy minimalizuje ilość informacji niezbędnej do klasyfikacji przykładów w partycjach uzyskanych w wyniku podziału

Zysk infomacyjny (2)

- Niech S oznacza zbiór s przykładów. Załóżmy, że atrybut decyzyjny posiada m różnych wartości definiujących m klas, C_i (dla i=1, ..., m)
- Niech s_i oznacza liczbę przykładów zbioru S należących do klasy C_i
- Oczekiwana ilość informacji niezbędna do zaklasyfikowania danego przykładu:

$$I(s_1, s_2, ..., s_m) = - \sum p_i \log_2(p_i)$$

Zysk infomacyjny (3)

- Niech s_{ij} oznacza liczbę przykładów z klasy C_i w partycji S_j. Entropię podziału zbioru S na partycje, według atrybutu A definiujemy następująco:

$$E(A_1, A_2, \ldots A_v) = \sum_{j=0}^{v} \frac{(s_{1j} + s_{2j} + \ldots + s_{mj})}{s} * I(s_{1j}, s_{2j}, \ldots, s_{mj})$$

- Im mniejsza wartość entropii, tym większa „czystość” podziału zbioru S na partycje

Zysk infomacyjny (4)

Zysk informacyjny, wynikający z podziału zbioru S na partycje według atrybutu A, definiujemy następująco:

$$\mathbf{Gain(A) = I(s_1, s_2, ..., s_m) - E(A)}$$

Gain(A) oznacza oczekiwaną redukcję entropii (nieuporządkowania) spowodowaną znajomością wartości atrybutu A

Przykład (1)

ID	wiek	dochód	student	status	kupi_komputer
1	<=30	wysoki	nie	kawaler	nie
2	<=30	wysoki	nie	żonaty	nie
3	31..40	wysoki	nie	kawaler	tak
4	>40	średni	nie	kawaler	tak
5	>40	niski	tak	kawaler	tak
6	>40	niski	tak	żonaty	nie
7	31..40	niski	tak	żonaty	tak
8	<=30	średni	nie	kawaler	nie
9	<=30	niski	tak	kawaler	tak
10	>40	średni	tak	kawaler	tak
11	<=30	średni	tak	żonaty	tak
12	31..40	średni	nie	żonaty	tak
13	31..40	wysoki	tak	kawaler	tak
14	>40	średni	nie	żonaty	nie

Przykład (2)

- Rozważmy przedstawiony zbiór treningowy opisujący klientów sklepu komputerowego
- Atrybut decyzyjny, kupi_komputer, posiada dwie wartości (tak, nie), stąd, wyróżniamy dwie klasy (m=2)

C_1 odpowiada wartości tak - $s_1 = 9$

C_2 odpowiada wartości nie - $s_2 = 5$

$I(s_1, s_2)=I(9, 5)= - 9/14\log_2 9/14 – 5/14\log_2 5/14=$ **0.94**

Przykład (3)

- Następnie, obliczamy entropię każdego deskryptora. Rozpocznijmy od atrybutu *wiek*

dla wiek =‘<=30’

$s_{11}=2 \quad s_{21}=3 \quad I(s_{11}, s_{21}) = 0.971$

dla wiek =’31..40’

$s_{12}=4 \quad s_{22}=0 \quad I(s_{12}, s_{22}) = 0$

dla wiek =‘>40’

$s_{13}=2 \quad s_{23}=3 \quad I(s_{13}, s_{23}) = 0.971$

Przykład (4)

- Entropia atrybutu wiek wynosi:

$$E(\text{wiek}) = 5/14 * I(s_{11}, s_{21}) + 4/14 * I(s_{12}, s_{22}) + 5/14 * I(s_{13}, s_{23}) = 0.694$$

- Zysk informacyjny wynikający z podziału zbioru S według atrybutu wiek wynosi:

$$\text{Gain}(\text{wiek}) = I(s_1, s_2) - E(\text{wiek}) = 0.246$$

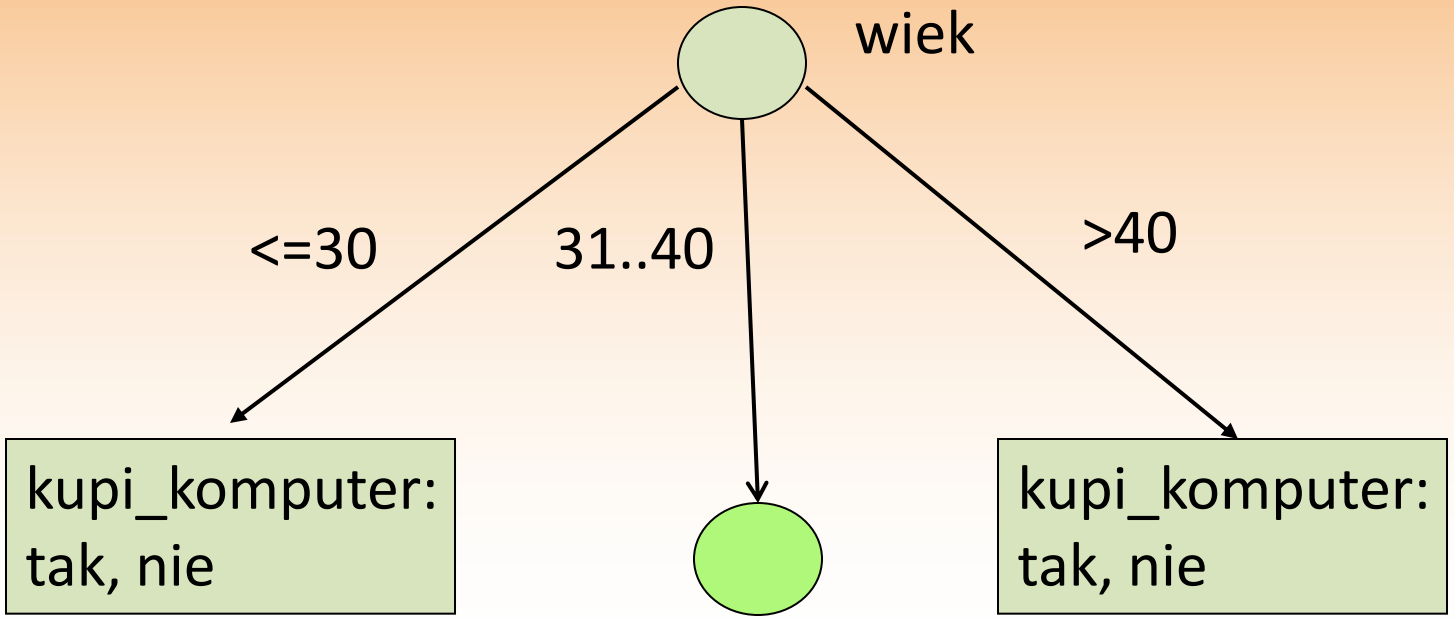
- Analogicznie obliczamy zysk informacyjny dla pozostałych atrybutów:

$\text{Gain}(\text{dochód}) = 0.029$,
$\text{Gain}(\text{student}) = 0.151$
$\text{Gain}(\text{status}) = 0.048$

Przykład (5)

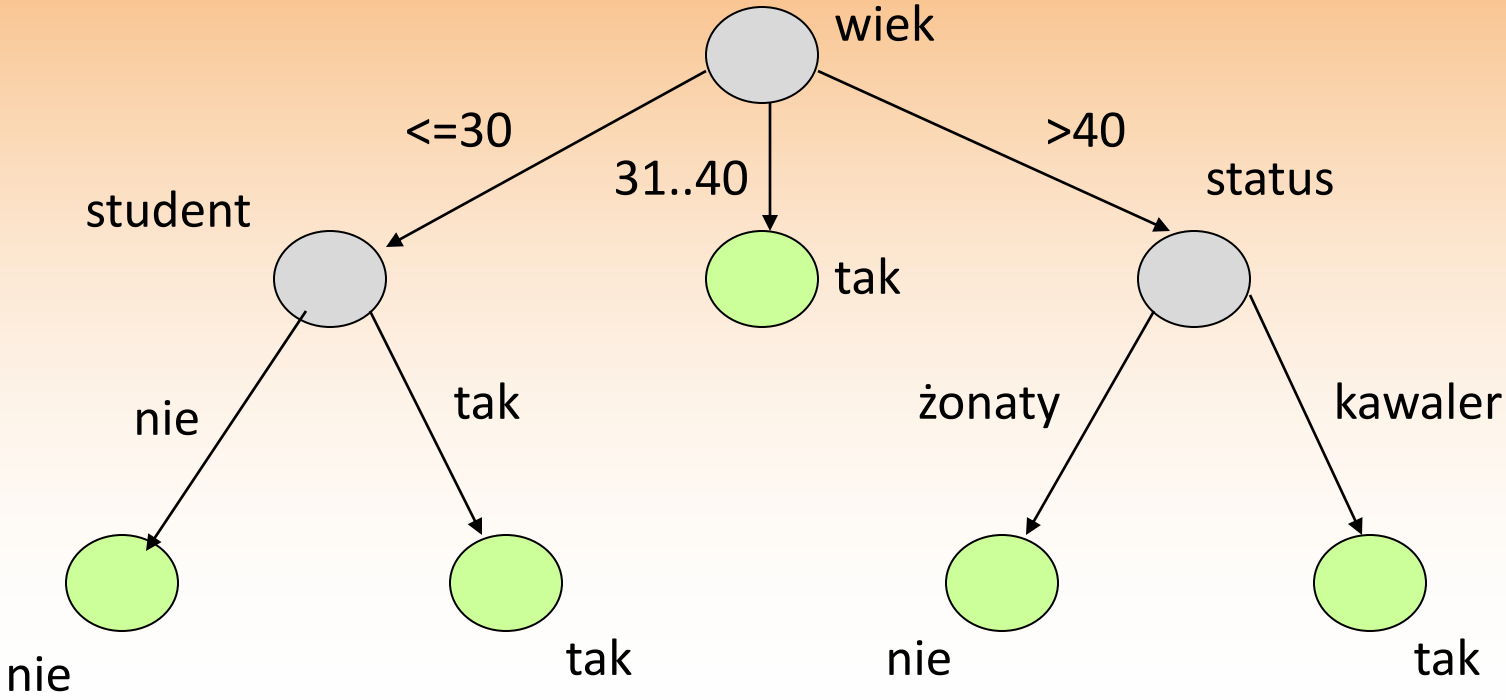
- Atrybut **wiek** daje największy zysk informacyjny spośród wszystkich deskryptorów, atrybut ten jest wybierany jako pierwszy atrybut testowy.
- Tworzymy wierzchołek drzewa o etykiecie **wiek**, oraz etykietowane łuki wychodzące, łączące wierzchołek **wiek** z wierzchołkami odpowiadającymi partycjom zbioru utworzonymi według atrybutu **wiek**

Przykład (6)



kupi_komputer: **tak**

Przykład (7)



Ostateczna postać drzewa decyzyjnego

Indeks Giniego (1)

- **Dany jest zbiór D** zawierający n rekordów oraz atrybut decyzyjny, który przyjmuje m różnych wartości, definiując m rozłącznych klas C_i, dla i = 1, …, m, s_i oznacza liczbę rekordów zbioru D należących do klasy C_i, dla i = 1, …, m.
- Nieuporządkowanie zbioru D, zgodnie z miarą indeksu Giniego, jest zdefiniowane następująco:

$$\text{gini}(D) = 1 - \sum_{j=1}^{m} P_j^2$$

gdzie P_j oznacza prawdopodobieństwo, że dany rekord zbioru D należy do klasy C_j; prawdopodobieństwo to estymujemy względną częstością występowania klasy C_j w D, to jest:

$$P_j = \frac{s_j}{|D|}$$

Indeks Giniego (2)

- Przykładowo: dwie klasy, Pos i Neg, oraz zbiór przykładów D zawierający p elementów należących do klasy Pos i n elementów należących do klasy Neg

$$P_{pos} = p/(p+n) \qquad P_{neg} = n/(n+p)$$

$$gini(D) = 1 - P_{pos}^2 - P_{neg}^2$$

- Załóżmy, że metodą wyboru punktu podziału został wybrany atrybut podziałowy A, który dzieli zbiór treningowy D na dwa podzbiory D_1 i D_2

Indeks Giniego (3)

- Indeks podziału Gini $gini^{A}_{split}(D_1, D_2)$ binarnego podziału zbioru treningowego D na podzbiory (partycje) D_1 i D_2, zgodnie z predykatem podziałowym zdefiniowanym dla atrybutu A, jest zdefiniowany w następujący sposób:

$$gini^{A}_{split}(D_1, D_2) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

Indeks Giniego (4)

- Indeks podziału Gini jest ważoną sumą indeksów Giniego podzbiorów D_1 i D_2 określających nieuporządkowanie zbiorów D_1 i D_2
- Indeks podziału Gini definiuje nieuporządkowanie zbioru treningowego D uzyskane w wyniku podziału na podzbiory na podzbiory D_1 i D_2
- Różnica wartości miar

$$\text{Gain}_{\text{Gini}}(A) = \text{gini}(D) - \text{gini}^{A}_{\text{split}}(D_1,\ D_2)$$

definiuje różnicę uporządkowania zbioru treningowego D przed i po podziale zbioru D na podzbiory D1 i D2 dla atrybutu podziałowego A

Indeks Giniego (5)

- $Gain_{Gini}(A)$ określa redukcję nieuporządkowania zbioru D w wyniku podziału zbioru D na podzbiory D_1 i D_2 dla atrybutu podziałowego A
- Maksymalną redukcje nieuporządkowania uzyskujemy, minimalizując wartość indeksu podziału Gini – stąd, „najlepszym” punktem podziału zbioru D jest punkt podziału, który charakteryzuje się najmniejszą wartością indeksu podziału Gini

Idea algorytmu

1. Dla każdego atrybutu, dla wszystkich możliwych punktów podziału, oblicz wartość indeksu podziału Gini – wybierz punkt podziału o najmniejszej wartości indeksu podziału Gini
2. Wybrany punkt podziału włącz do drzewa decyzyjnego
3. Punkt podziału dzieli zbiór D na dwie partycje D_1 i D_2
4. Powtórz procedurę obliczania indeksu podziału Gini dla partycji D_1 i D_2 – znalezione punkty podziału włącz do drzewa decyzyjnego
5. Powtarzaj procedurę dla kolejnych partycji aż do osiągnięcia warunku stopu

Przykład (1)

ID	Wiek	Typ_sam	Ryzyko
0	23	family	high
1	17	sport	high
2	43	sport	high
3	68	family	low
4	32	truck	low
5	20	family	high

Zbiór treningowy D

Lista wartości atrybutu Wiek oraz atrybutu Typ_sam

Wiek	ID	Ryzyko
17	1	high
20	5	high
23	0	high
32	4	low
43	2	high
68	3	low

Typ_sam	ID	Ryzyko
family	0	high
sport	1	high
sport	2	high
family	3	low
truck	4	low
family	5	high

Przykład (2)

- Możliwe punkty podziału dla atrybutu Wiek:

  $\text{Wiek} \leq 17, \text{Wiek} \leq 20, \text{Wiek} \leq 23, \text{Wiek} \leq 32, \text{Wiek} \leq 43$

- Obliczmy wartości indeksu podziału Gini dla poszczególnych punktów podziału

$\text{Wiek} \leq 17$

Liczba krotek	High	Low
Wiek<=17	1	0
Wiek >17	3	2

$$\text{gini}(\text{Wiek} <= 17) = 1\text{-}\ (1^2+0^2) = 0$$
$$\text{gini}(\text{Wiek} > 17) = 1\text{-}\ ((3/5)^2+(2/5)^2) = 0{,}73$$
$$\text{gini}_{\text{SPLIT}} = (1/6) * 0 + (5/6) * (0{,}73)^2 = 0{,}4$$

Przykład (3)

Wiek $\leq$ 20

Liczba krotek	High	Low
Wiek<=20	2	0
Wiek >20	2	2

gini(Wiek<=20) = 1- (1^2+0^2) = 0
gini(Wiek >20) = 1- $((1/2)^2+(1/2)^2)$ = 1/2
gini $_{SPLIT}$ = (2/6) * 0 + (4/6) * (1/8) = 1/3

Wiek $\leq$ 23

Liczba krotek	High	Low
Wiek <=23	3	0
Wiek >23	1	2

gini(Wiek $\leq$23) = 1- (1^2+0^2) = 0
gini(Wiek >23) = 1- $((1/3)^2+(2/3)^2)$ = 4/9
gini $_{SPLIT}$ = (3/6) * 0 + (3/6) * (4/9) = 2/9

Przykład (4)

Wiek $\leq$ 32

Liczba krotek	High	Low
Wiek <=32	3	1
Wiek >32	1	1

gini(Age$\leq$32) = 1- $((3/4)^2+(1/4)^2)$ = 3/8
gini(Age>32) = 1- $((1/2)^2+(1/2)^2)$ = 1/2
gini $_{SPLIT}$ = (4/6)*(3/8) + (2/6)*(1/2) = 7/24

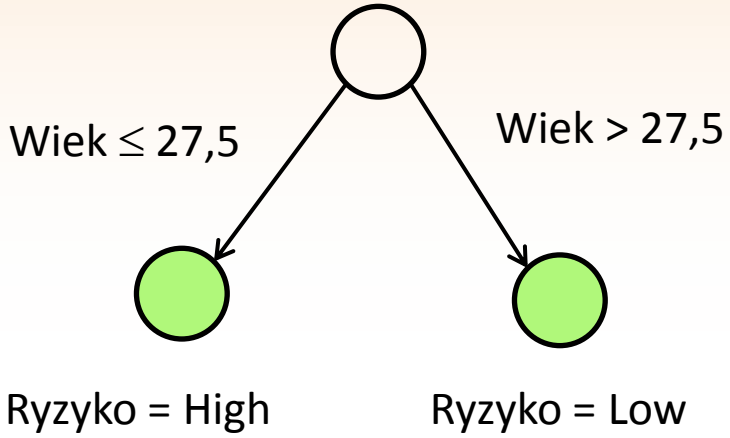
Wiek $\leq$ 43

Liczba krotek	High	Low
Wiek <=43	4	1
Wiek >43	0	1

gini(Age$\leq$43) = 1- $((4/5)^2+(1/5)^2)$ = 8/25
gini(Age>43) = 1- $((1/1)^2+(0/1)^2)$ = 0
gini $_{SPLIT}$ = (5/6)*(8/25) + (1/6)*(0) = 4/15

Przykład (5)

- Obliczamy następnie wartości indeksu podziału $gini_{split}$ dla atrybutu Typ_sam
- Najmniejszą wartość indeksu podziału $gini_{SPLIT}$ posiada punkt podziału Wiek ≤ 23, stąd, tworzymy wierzchołek drzewa decyzyjnego dla punktu podziału Wiek $= (23+32) / 2 = 27.5$



Przykład (6)

Listę wartości atrybutów dzielimy w punkcie podziału:
Listy wartości atrybutów dla Wiek ≤27.5:

Wiek	ID	Ryzyko
17	1	high
20	5	high
23	0	high

Typ_sam	ID	Ryzyko
family	0	high
sport	1	high
family	5	high

Listy wartości atrybutów dla Wiek > 27.5:

Wiek	ID	Ryzyko
32	4	low
43	2	high
68	3	low

Typ_sam	ID	Ryzyko
sport	2	high
family	3	low
truck	4	low

Przykład (7)

Ocena punktów podziału dla atrybutu kategorycznego

Musimy dokonać oceny wszystkich punktów podziału atrybutu kategorycznego ($2^N - 2$ kombinacji), gdzie N oznacza liczbę wartości atrybutu kategorycznego

Liczba krotek	High	Low
Typ_sam= {sport}	1	0
Typ_sam ={family]	0	1
Typ_sam= {truck}	0	1

$gini(Typ\_sam \in \{sport\}) = 1 - 1^2 - 0^2 = 0$
$gini(Typ\_sam \in \{family\}) = 1 - 0^2 - 1^2 = 0$
$gini(Typ\_sam \in \{truck\}) = 1 - 0^2 - 1^2 = 0$

Przykład (8)

$gini(Typ\_sam \in \{ sport, family \}) = 1 - (1/2)^2 - (1/2)^2 = 1/2$

$gini(Typ\_sam \in \{ sport, truck \}) = 1/2$

$gini(Typ\_sam \in \{ family, truck \}) = 1 - 0^2 - 1^2 = 0$

$gini_{SPLIT}(Typ\_sam \in \{ sport \}) = (1/3) * 0 + (2/3) * 0 = 0$

$gini_{SPLIT}(Typ\_sam \in \{ family \}) = (1/3) * 0 + (2/3)*(1/2) = 1/3$

$gini_{SPLIT}(Typ\_sam \in \{ truck \}) = (1/3) * 0 + (2/3)*(1/2) = 1/3$

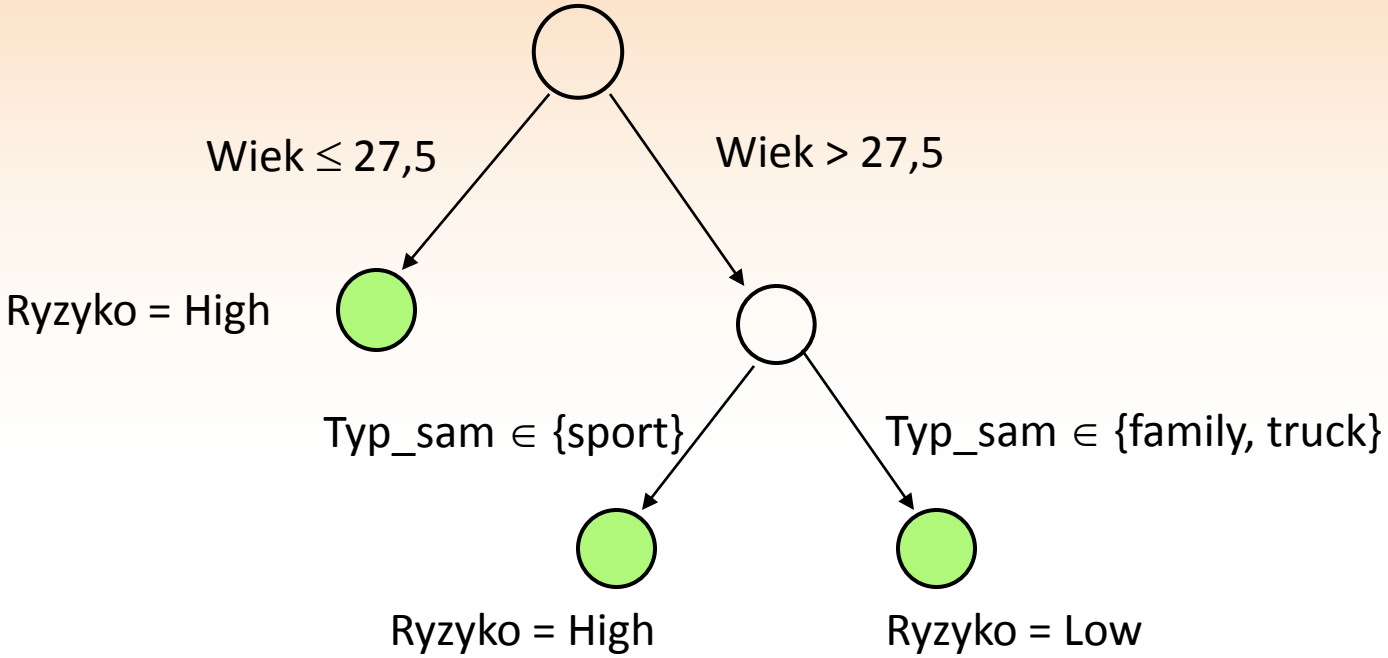
$gini_{SPLIT}(Typ\_sam \in \{ sport, family\}) = (2/3)*(1/2)+(1/3)*0= 1/3$

$gini_{SPLIT}(Typ\_sam \in \{ sport, truck\}) = (2/3)*(1/2)+(1/3)*0= 1/3$

$gini_{SPLIT}(Typ\_sam \in \{ family, truck \}) = (2/3)*0+(1/3)*0=0$

Przykład (9)

Najmniejszą wartość indeksu podziału $gini_{SPLIT}$ posiada punkt podziału Typ_sam $\in$ {sport}. Tworzymy wierzchołek w drzewie decyzyjnym dla tego punktu podziału. Drzewo decyzyjne po wprowadzeniu wierzchołka ma postać:



Błędy klasyfikatorów (1)

- Błędy popełniane przez modele klasyfikacyjne:
  - błędy treningowe - błędem treningowym klasyfikatora nazywamy stosunek niepoprawnie zaklasyfikowanych rekordów zbioru treningowego do łącznej liczby rekordów tego zbioru
  - błędy testowe - błędem testowym klasyfikatora nazywamy stosunek niepoprawnie zaklasyfikowanych rekordów zbioru testowego do łącznej liczby rekordów tego zbioru
  - błędy generalizacji (lub uogólnienia) - błędem generalizacji klasyfikatora nazywamy oczekiwany błąd klasyfikacji na zbiorze nowych rekordów, dla których wartość atrybutu decyzyjnego nie jest znana

Błędy klasyfikatorów (2)

- Cel algorytmów konstrukcji drzew decyzyjnych – konstrukcja drzew decyzyjnych maksymalnie wiernie odzwierciedlających strukturę zbioru treningowego (minimalizacja błędu treningowego)
- Drzewo decyzyjne wiernie odzwierciedlające strukturę zbioru treningowego – drzewo silnie *dopasowane* do danych treningowych

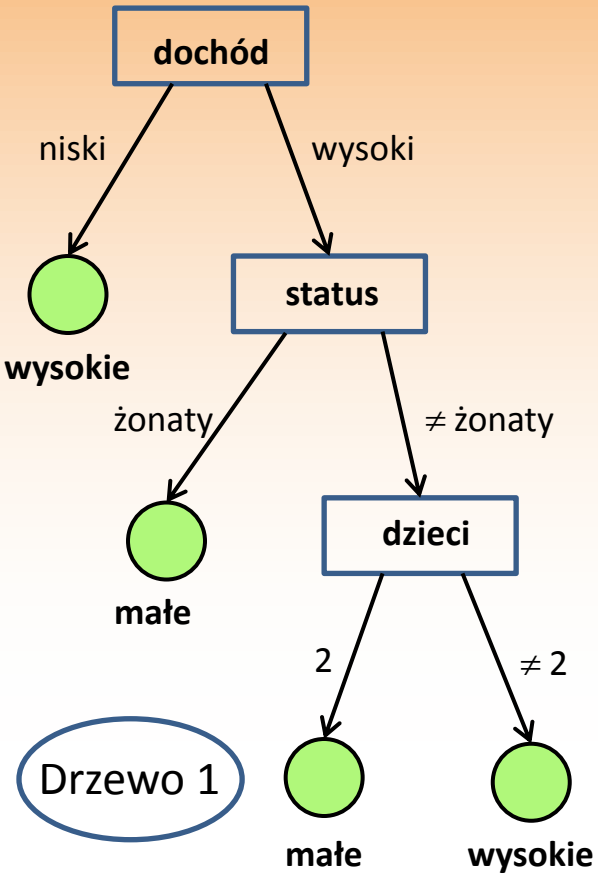
  minimalizacja błędu treningowego $\rightarrow$ minimalizacja błędu testowego

  minimalizacja błędu testowego $\rightarrow$ minimalizacja błędu generalizacji

- Drzewo decyzyjne o zerowym błędzie treningowym może mieć niezerowy błąd testowy, i drzewo decyzyjne o niezerowym błędzie treningowym może mieć zerowy błąd testowy

Przykład (1)

id	wiek	status	dochód	dzieci	ryzyko
1	25	kawaler	niski	0	wysokie
2	28	żonaty	niski	1	wysokie
3	29	kawaler	wysoki	0	małe
4	31	kawaler	niski	0	wysokie
5	35	żonaty	średni	1	małe
6	38	rozwodnik	wysoki	2	małe
7	38	rozwodnik	niski	2	wysokie
8	39	rozwodnik	wysoki	0	wysokie
9	41	żonaty	średni	1	małe
10	42	rozwodnik	średni	4	wysokie
11	45	żonaty	średni	2	małe
12	48	żonaty	średni	1	małe
13	56	żonaty	wysoki	2	małe
14	56	rozwodnik	wysoki	2	wysokie



Przykład (2)

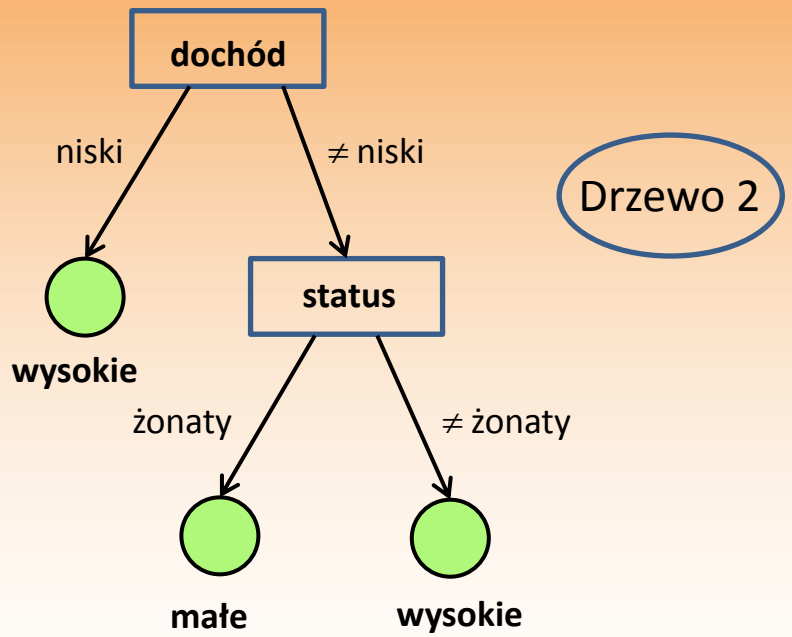
id	wiek	status	dochód	dzieci	ryzyko
15	27	żonaty	niski	1	wysokie
16	35	żonaty	średni	2	małe
17	40	rozwodnik	średni	2	wysokie
18	48	kawaler	niski	0	wysokie
19	55	rozwodnik	średni	2	wysokie
20	44	żonaty	wysoki	1	małe

Zbiór danych testowych

Błąd treningowy Drzewa 1 wynosi 0%

Błąd klasyfikacji Drzewa 1 na danych testowych (błąd testowy) wynosi 30%

Przykład (3)



Błąd treningowy Drzewa 2 wynosi 21,4 %

Błąd klasyfikacji Drzewa 2 na danych testowych (błąd testowy) wynosi 0 %

Modyfikację drzewa decyzyjnego, polegającą na zastąpieniu poddrzewa liściem drzewa, nazywamy *upraszczaniem* lub *przycinaniem* drzewa decyzyjnego

Naiwny klasyfikator Bayesa

- Naiwny klasyfikator Bayesa jest klasyfikatorem statystycznym - oparty na twierdzeniu Bayesa
- Celem klasyfikatora jest predykcja prawdopodobieństwa, że dany obiekt (przykład) należy do określonej klasy decyzyjnej
- Niech X oznacza przykład, którego klasa nie jest znana - każdy przykład jest reprezentowany w postaci n-wymiarowego wektora, X=(x1, x2, ..., xn)
- Problem klasyfikacji przykładu X można sformułować następująco:

  wyznacz prawdopodobieństwo a posteriori $P(C = C_i \mid X)$ klasy C_i przy znajomości rekordu X (tj. prawdopodobieństwo, że przykład X należy do klasy C_i), a następnie przydziel rekord X do klasy o największym prawdopodobieństwie a posteriori $P(C = C_i \mid X)$

Twierdzenie Bayesa

- W jaki sposób oszacować prawdopodobieństwo a-posteriori $P(C = C_i|X)$?
- Twierdzenie Bayesa:

$$P(C = C_i\,|X) = \frac{P(X \mid C = C_i)\,P(C = C_i)}{P(X)}$$

gdzie $P(C = C_i)$ oznacza prawdopodobieństwo a piori wystąpienia klasy C_i (tj. prawdopodobieństwo, że dowolny przykład należy do klasy C_i), $P(X|C = C_i)$ oznacza prawdopodobieństwo a-posteriori, że X należy do klasy C_i, i $P(X)$ oznacza prawdopodobieństwo a priori wystąpienia przykładu X

Założenie o warunkowej niezależności atrybutów

- W jaki sposób obliczyć $P(X|C = C_i)$?
- Dla dużych zbiorów danych, o dużej liczbie deskryptorów, obliczenie $P(X|C = C_i)$ będzie bardzo kosztowne
- Rozwiązanie - przyjmujemy założenie o warunkowej niezależności atrybutów
- Założenie o warunkowej niezależności atrybutów prowadzi do następującej formuły:

$$P(X \mid C = C_i) = \prod_{i=1}^{n} P(A_i = x_i \mid C = C_i)$$

Szacowanie prawdopodobieństw warunkowych

id	wiek	status	dochód	dzieci	ryzyko
1	25	kawaler	1600	0	wysokie
2	35	żonaty	3100	1	małe
3	38	rozwodnik	6700	2	małe
4	45	żonaty	3300	2	małe
5	28	żonaty	1800	1	wysokie
6	39	rozwodnik	6500	0	wysokie
7	31	kawaler	1500	0	wysokie
8	56	żonaty	7500	2	małe
9	48	żonaty	3200	1	małe
10	38	rozwodnik	1300	2	wysokie
11	29	kawaler	6700	0	małe
12	42	rozwodnik	3500	4	wysokie
13	41	żonaty	3700	1	małe
14	56	rozwodnik	7700	2	małe

Przykładowy zbiór danych treningowych

Problem „częstości zero”

- Po wprowadzeniu korekty prawdopodobieństwo warunkowe $P(A_j = x_j \mid C = C_i)$ estymujemy wzorem:

$$P(A_j = x_j \mid C = C_i) = \frac{s_{ij} + \lambda}{s_i + \lambda \cdot m_j}$$

gdzie m_j oznacza liczbę różnych wartości atrybutu Aj , a λ współczynnik skalujący (współczynnik korekcji), który najczęściej przyjmuje wartość $\lambda = 1/n$.
Jeżeli przyjmiemy współczynnik $\lambda = 1$, to otrzymamy tak zwany *estymator Laplace'a* prawdopodobieństwa warunkowego

Klasyfikator najbliższego sąsiedztwa

- Klasyfikator najbliższego sąsiedztwa należy do grupy klasyfikatorów opartych na analizie przypadku (ang. *instancje-based classification methods*)
- Klasyfikator nie konstruuje modelu klasyfikacyjnego (klasyfikatora) - proces klasyfikacji jest realizowany on-line, gdy zachodzi potrzeba dokonania klasyfikacji nowego przypadku
- Klasyfikacja metod:
  - Leniwe metody uczące
  - Gorliwe metody uczące
- Każdy przykład ze zbioru treningowego jest opisany n-wymiarowym wektorem reprezentującym punkt w przestrzeni n-wielowymiarowej nazywanej przestrzenią wzorców (pattern space)

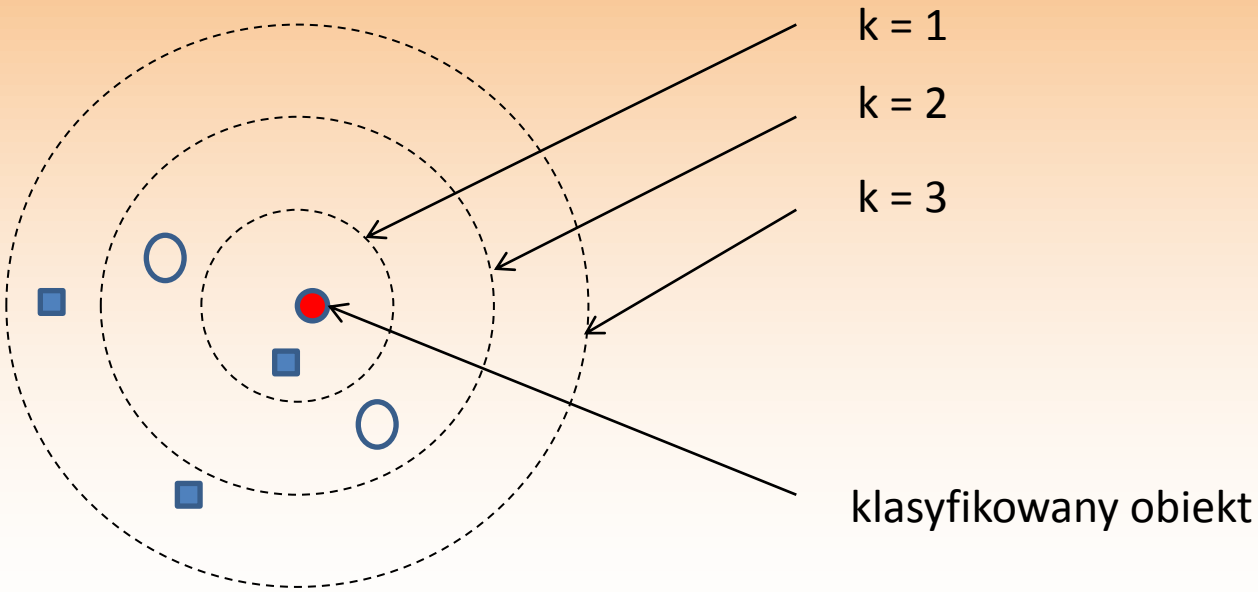
Klasyfikator 1NN

- Klasyfikacja nowego przypadku X – poszukujemy punktu w przestrzeni wzorców, który jest „najbliższy” nowemu przypadkowi
- Przypadek X klasyfikujemy jako należący do klasy, do której należy „najbliższy” punkt w przestrzeni wzorców
- Wada metody 1NN: metoda jest bardzo czuła na punkty osobliwe i szum w danych treningowych

Klasyfikator kNN

- Rozwiązanie problemu 1NN: zastosowanie strategii k-najbliższych sąsiadów
- Klasyfikacja nowego przypadku X – poszukujemy k najbliższych punktów w przestrzeni wzorców, tj. k najbliższych sąsiadów
- Przypadek X klasyfikujemy jako należący do klasy, która dominuje w zbiorze k najbliższych sąsiadów
- Do znalezienia k najbliższych sąsiadów wykorzystujemy indeksy wielowymiarowe (np. R-drzewa)

Wybór wartości parametru k (1)



Wybór wartości parametru k (2)

- Określenie klasy decyzyjnej klasyfikowanego obiektu na podstawie listy najbliższych k sąsiadów
  - głosowanie większościowe
  - waga głosu proporcjonalna do odległości od klasyfikowanego obiektu (waga głosu w, $w = 1/d^2$)
- Adaptacyjny dobór wielkości parametru k na podstawie wyników analizy trafności klasyfikacji (lub błędu klasyfikacji)

Ocena jakości klasyfikatora

- Ocenę jakości modelu klasyfikacyjnego przeprowadza się w odniesieniu do zbioru rekordów testowych
- Dla każdego rekordu testowego jest znana etykieta klasy tego rekordu - rekordy testowe są poddawane klasyfikacji za pomocą klasyfikatora skonstruowanego na podstawie zbioru rekordów treningowych a następnie etykiety klas przypisane tym rekordom przez klasyfikator są porównywane z rzeczywistymi etykietami klas rekordów testowych.
- Zliczana jest liczba rekordów testowych poprawnie i niepoprawnie zaklasyfikowanych przez klasyfikator

Macierz pomyłek (1)

- Macierz pomyłek jest macierzą kwadratową m $\times$ m, gdzie m oznacza liczbę etykiet klas, w której wiersze odpowiadają rzeczywistym etykietom klas rekordów testowych, a kolumny – etykietom klas przypisywanym rekordom testowym przez klasyfikator
- Element f_{ij} macierzy pomyłek oznacza liczbę rekordów testowych z klasy C_i przypisanych błędnie przez klasyfikator do klasy C_j
- Macierz pomyłek dla klasyfikacji binarnej

Klasa rzeczywista	Klasa przewidywana	
	C_1	C_2
C_1	f_{11}	f_{12}
C_2	f_{21}	f_{22}

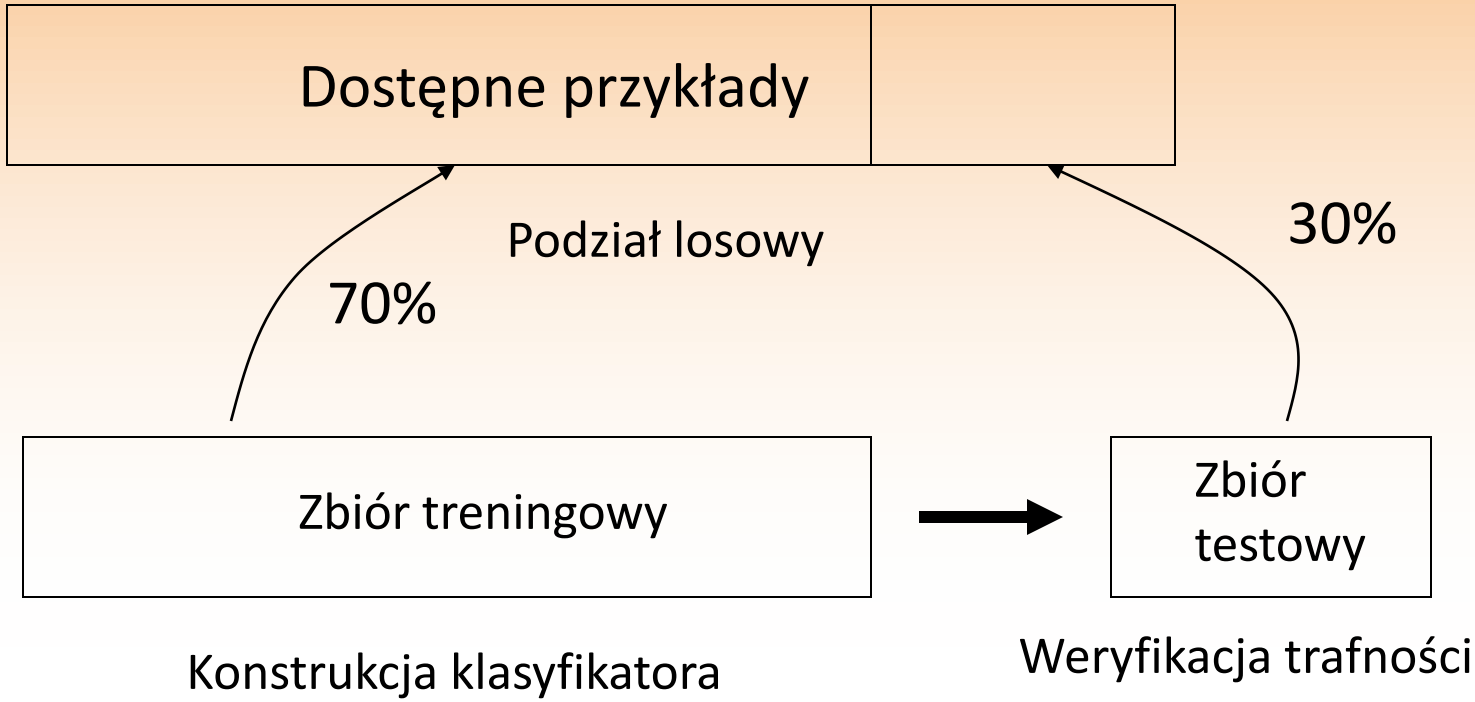
Macierz pomyłek (2)

- Elementy macierzy pomyłek dla klasyfikacji binarnej oznacza się:

$$TP = f_{11} \qquad FN = f_{12} \qquad FP = f_{21} \qquad TN = f_{22}$$

Klasa rzeczywista	Klasa przewidywana		liczba wystąpień
	C_1	C_2	
C_1	f_{11}	f_{12}	P
C_2	f_{21}	f_{22}	N

Metoda wydzielania



K-krotna walidacja krzyżowa