Sztuczna inteligencja

WYKŁAD 3

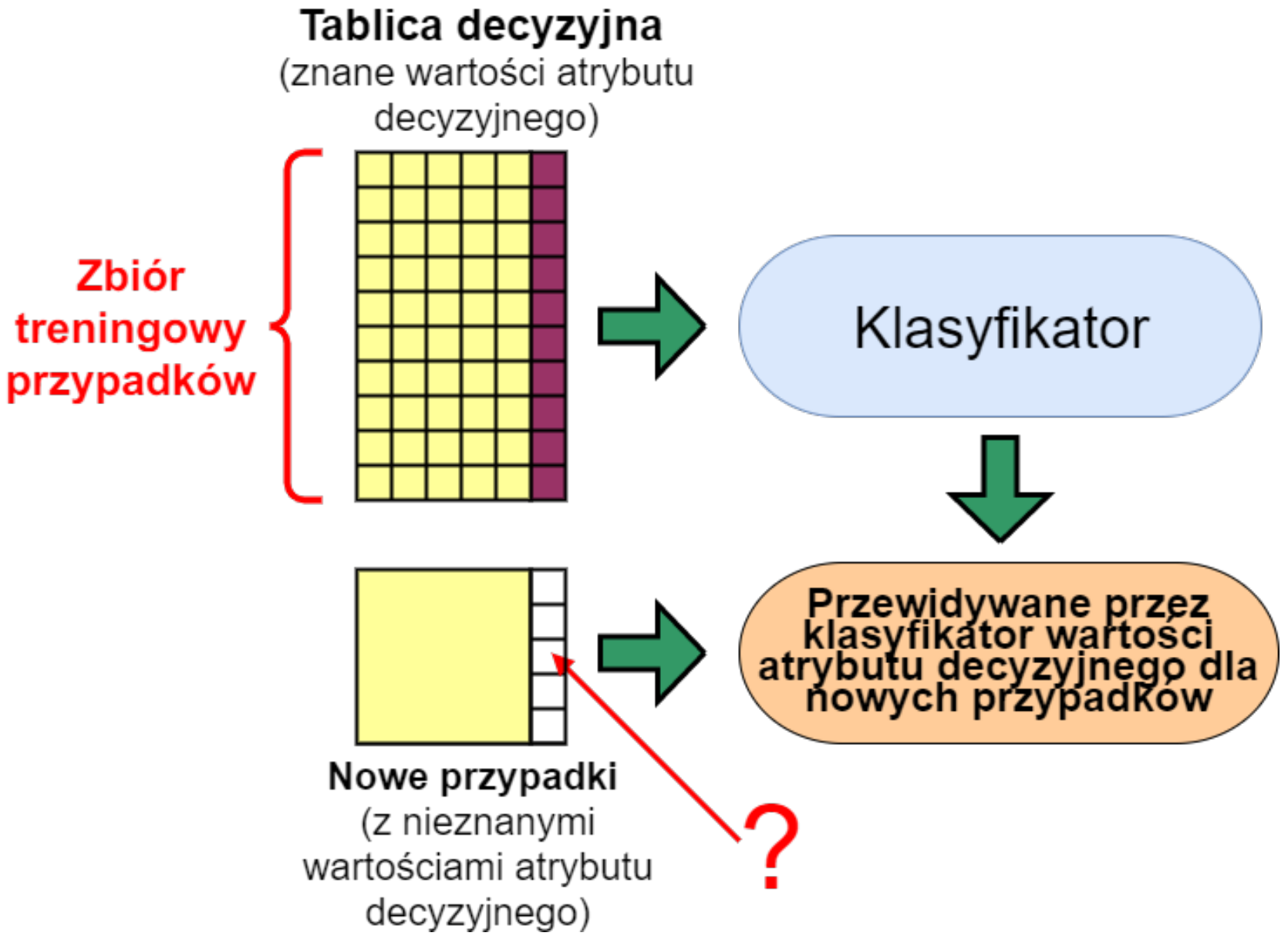
Tabela decyzyjna – przykład

Atrybuty warunkowe (opisujące przypadki) **Atrybuty decyzyjny**

Klient	Oszczędności	Majątek	Dochód (w 1000$)	Ryzyko kredytowe
1	średnie	duży	75	małe ryzyko
2	małe	mały	50	duże ryzyko
3	duże	średni	25	duże ryzyko
4	średnie	średni	50	małe ryzyko
5	małe	średni	100	małe ryzyko
6	duże	duży	25	małe ryzyko
7	małe	mały	25	duże ryzyko
8	średnie	średni	75	małe ryzyko

Źródło: D.T. Larose: „Odkrywanie wiedzy z danych”. PWN, Warszawa, 2006.

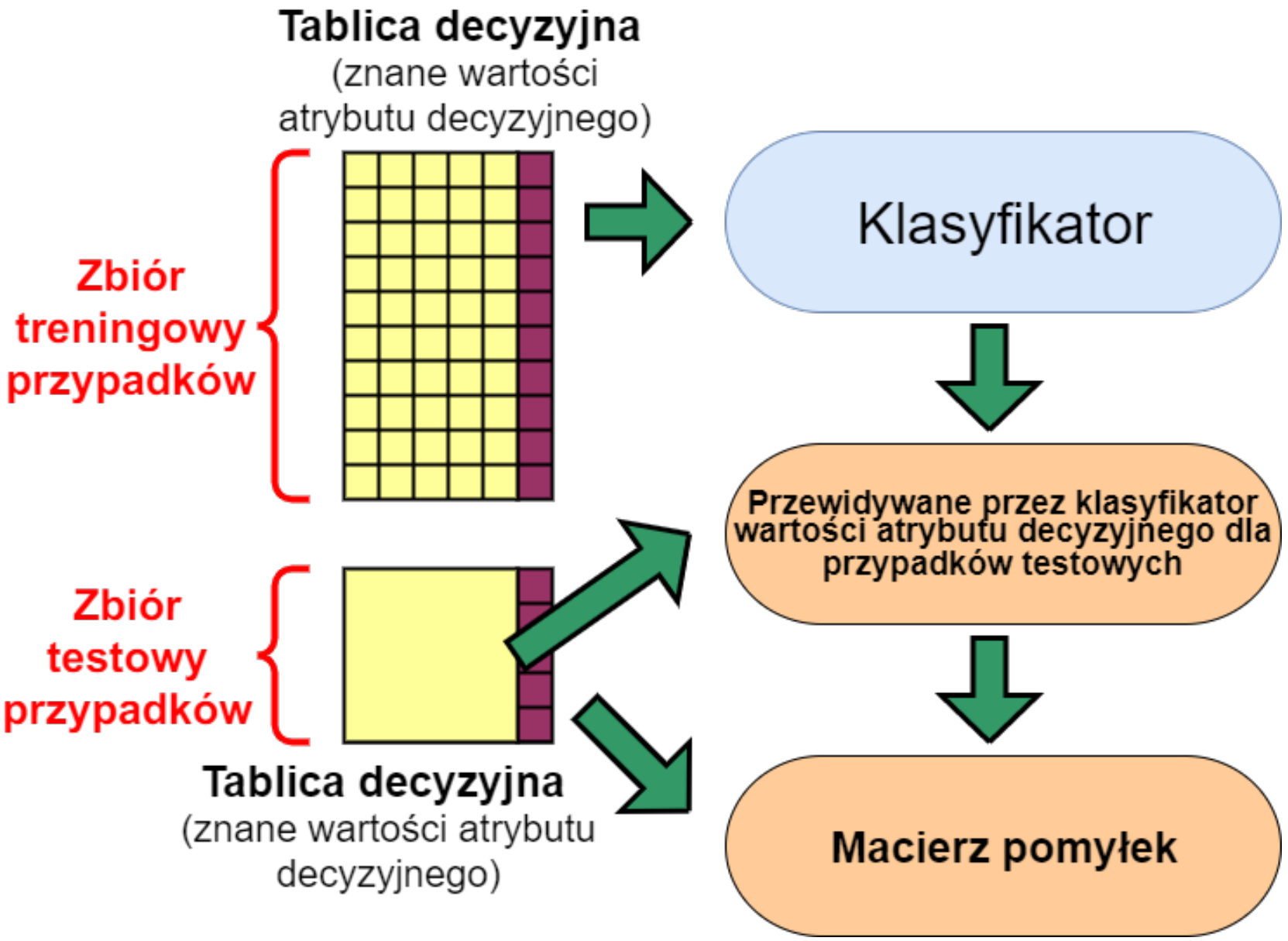
Problem klasyfikacji



Klasyfikatory

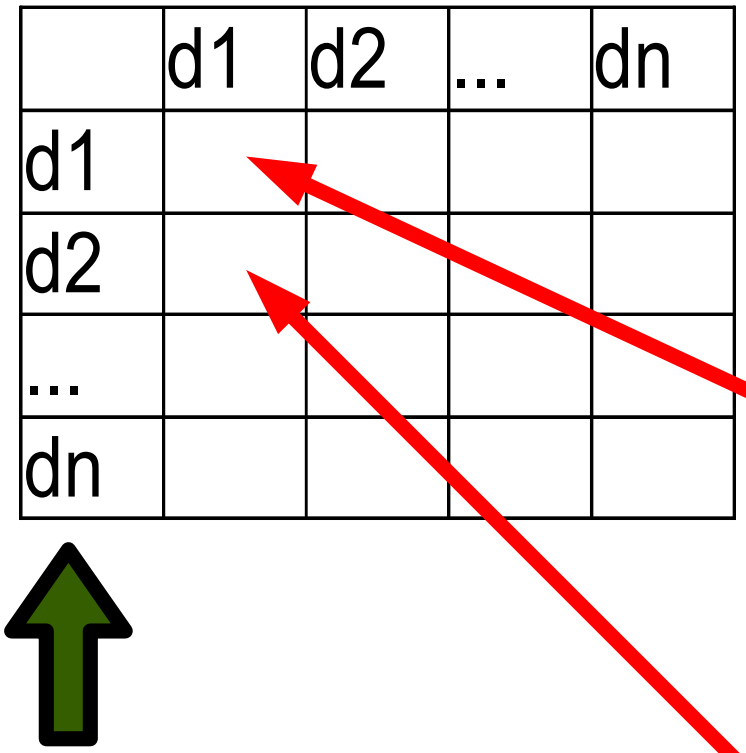
- Klasyfikatory oparte o model, np.:
  - model w postaci sieci neuronowej
  - model w postaci drzewa decyzyjnego
  - model w postaci zbioru reguł (IF-THEN)
  - model w postaci sieci Bayesa
- Klasyfikatory leniwe, np.:
  - k-NN (k najbliższych sąsiadów)

Ocena jakości klasyfikatora



Ocena jakości klasyfikatora

Macierz pomyłek (konfuzji, rozrzutu)



Przewidywane przez klasyfikator wartości atrybutu decyzyjnego

Aktualne wartości atrybutu decyzyjnego

Liczba przypadków z przypisaną aktualną decyzją d1 oraz przewidywaną decyzją d1

Liczba przypadków z przypisaną aktualną decyzją d2 oraz przewidywaną decyzją d1

Ocena jakości klasyfikatora

- Strategie:
  - Niezależne zbiory: treningowy i testowy.
  - Losowy podział zbioru wszystkich przypadków na zbiory: treningowy i testowy w ustalonej proporcji.
  - Kroswalidacja *k*-krotna
    - kroswalidacja stratyfikowana *k*-krotna (zachowane są oryginalne proporcje pomiędzy klasami decyzyjnymi).
  - *Leave one out* (szczególny przypadek kroswalidacji *k*-krotnej).

Ocena jakości klasyfikatora

- Kroswalidacja *k*-krotna

```
podzbiory[1..k]<-losowy podział zbioru przypadków na
    k podzbiorów

Dla każdego i od 1 do k:
    zbiór treningowy ← podzbiory z wyjątkiem i-tego
    zbiór testowy ← podzbiór i-ty
    uczenie klasyfikatora na zbiorze treningowym
    testowanie klasyfikatora na zbiorze testowym
```

Ocena jakości klasyfikatora

- *Leave one out* (*n* - liczba przypadków)

```
Dla każdego i od 1 do n:
    zbiór treningowy ← zbiór przypadków z wyjątkiem i-tego
    zbiór testowy ← i-ty przypadek
    uczenie klasyfikatora na zbiorze treningowym
    testowanie klasyfikatora na zbiorze testowym
```

Ocena jakości klasyfikatora

Klasa przewidywana → Klasa aktualna ↓	**Pozytywna**	**Negatywna**
Pozytywna	**TP**	**FN**
Negatywna	**FP**	**TN**

- TP (true positive) – liczba przypadków prawdziwie pozytywnych
- FP (false positive) – liczba przypadków fałszywie pozytywnych
- TN (true negative) – liczba przypadków prawdziwie negatywnych
- FN (false negative) – liczba przypadków fałszywie negatywnych

Ocena jakości klasyfikatora

- Dokładność (*accuracy*): $$accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$
- Precyzja (*precision*): $$precision = \frac{TP}{TP + FP}$$
- Zwrot (*recall*) $$recall = \frac{TP}{TP + FN}$$
- F-miara (*F-masure*) $$F = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

Drzewa decyzyjne

- Metody uczenia się drzew decyzyjnych to najbardziej znane i najczęściej stosowane w praktyce algorytmy indukcji symbolicznej reprezentacji wiedzy z przykładów.
- Struktura drzewa jest dość czytelna dla człowieka.
- Drzewo decyzyjne składa się z korzenia, z którego wychodzą co najmniej dwie gałęzie do węzłów leżących na niższym poziomie.
- Z każdym węzłem związany jest test sprawdzający wartości atrybutu opisującego przykłady.

Drzewa decyzyjne

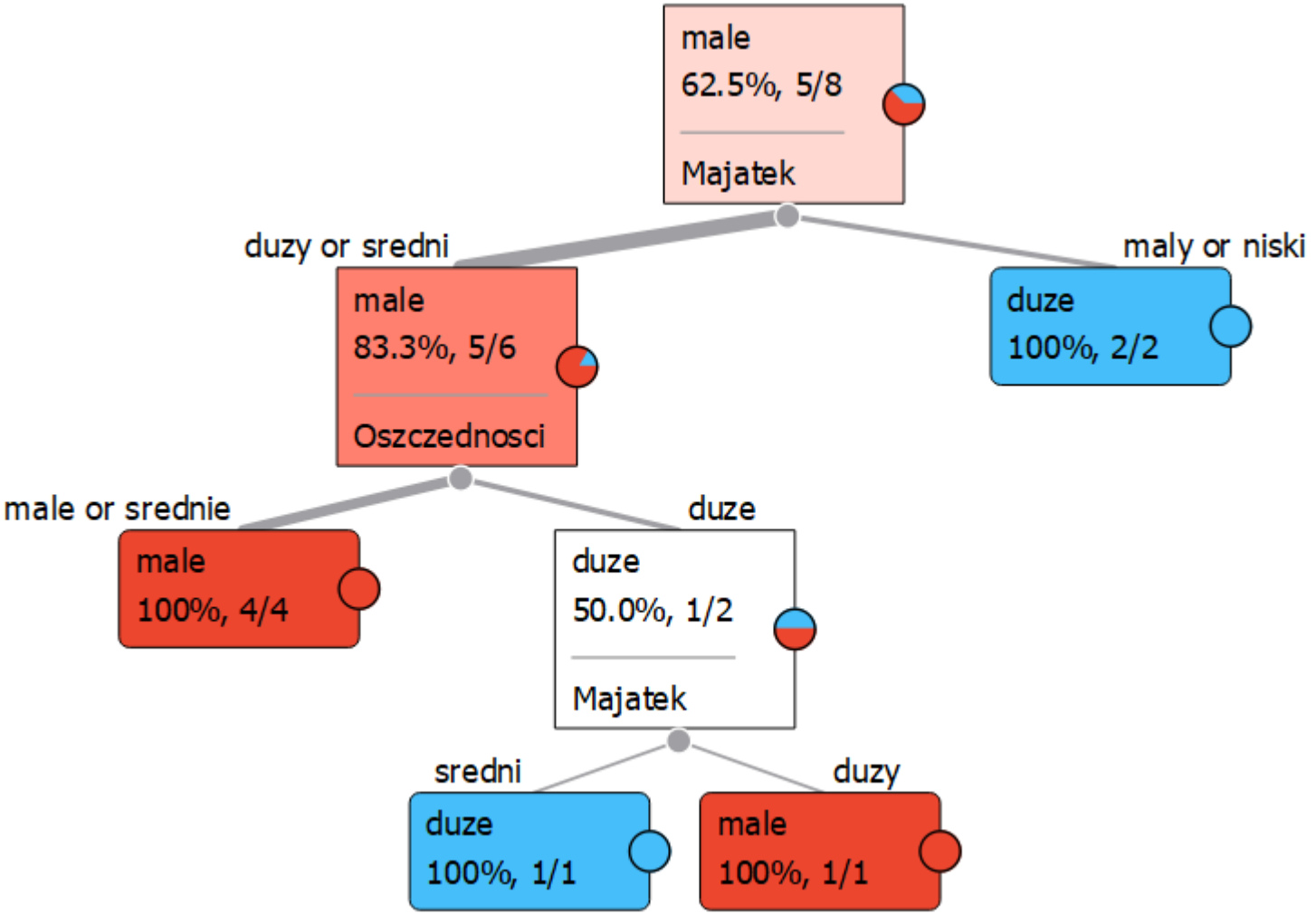
- Dla każdego z możliwych wyników testu odpowiadająca mu gałąź prowadzi do węzła na niższym poziomie drzewa.
- Węzłom, z których nie wychodzą żadne gałęzie (nazywanym liśćmi) przypisane są odpowiednie klasy decyzyjne.
- Ścieżki prowadzące od korzenia do liścia drzewa reprezentują koniunkcję pewnych testów zdefiniowanych na wartościach atrybutów opisujących przykłady uczące.

Drzewa decyzyjne

- Drzewo decyzyjne może więc posłużyć do określenia zbioru reguł określających przydział przykładów do klas decyzyjnych.
- Każda ścieżka drzewa od korzenia do liścia odpowiada jednej regule.

Drzewa decyzyjne

- Przykład:



Konstruowanie drzew decyzyjnych

- Większość algorytmów konstrukcji (uczenia się) drzew decyzyjnych oparta jest na schemacie zstępującego konstruowania drzewa (TDIDT – *Top Down Induction of Decision Trees*).
- Przykładowe algorytmy:
  - CART
  - ID3
  - C4.5
- Generalnie różnice pomiędzy poszczególnymi algorytmami dotyczą wyboru optymalnego podziału.

Algorytm C4.5

- Został zaproponowany przez Quinlana.
- Przy wyborze optymalnego podziału algorytm wykorzystuje pojęcie zysku informacji.
- Algorytm umożliwia generowanie drzewa decyzyjnego dla systemu z brakującymi wartościami atrybutów.

Algorytm C4.5

- Testy dla atrybutów symbolicznych
  - dla każdej wartości atrybutu symbolicznego tworzona jest osobna gałąź
- Testy dla atrybutów numerycznych ciągłych
  - stosowany jest test binarny:
    - wartość atrybutu mniejsza lub równa od wartości progowej
    - wartość atrybutu większa od wartości progowej

Entropia zbioru treningowego

$$E(S) = -\sum_{i=1}^{k} p_i \log p_i$$

k – liczba klas decyzyjnych
n – liczba wszystkich przykładów uczących (treningowych)
n_i – liczba przykładów uczących (treningowych) należących do i-tej klasy

$$p_i = \frac{n_i}{n}$$

$$i = 1, 2 \ldots, k$$

Zysk (przyrost) informacji

$$IG(S,a) = E(S) - \sum_{v \in V_a} \frac{n_v}{n} E(S_v)$$

a – cecha (atrybut)
V – zbiór wartości cechy a

n_V – liczba przykładów uczących (treningowych),

dla których cecha a ma wartość v
S_V – podzbiór zbioru przykładów uczących (treningowych), dla których cecha a ma wartość v

Możliwe podziały dla C4.5 – przykład

Podział	Poddrzewa		
1	oszczędności = małe	oszczędności = średnie	oszczędności = duże
2	majątek = mały	majątek = średni	majątek = duży
3	dochód $\leq$ 25 000$	dochód > 25 000$	
4	dochód $\leq$ 50 000$	dochód > 50 000$	
5	dochód $\leq$ 75 000$	dochód > 75 000$	

Źródło: D.T. Larose: „Odkrywanie wiedzy z danych”. PWN, Warszawa, 2006.

Zjawisko przeuczenia

- Drzewo decyzyjne jest przeuczone (nadmiernie dopasowane) do danego zbioru uczącego, jeśli istnieje inne drzewo o większym błędzie na tym zbiorze, które mimo to ma mniejszy błąd na całym rozkładzie przypadków (tj. obejmującego także przykłady, które nie wystąpiły z zbiorze uczącym).
- Drzewo przeuczone odzwierciedla przypadkowe przekłamania w danych lub zbyt szczegółowe regularności, nieistotne dla klasyfikacji przypadków.

Upraszczanie (przycinanie drzewa)

- Usuwane są pewne fragmenty drzewa (tj. podrzewa) o niewielkim znaczeniu dla klasyfikacji.
- W wyniku lokalnego usunięcia podrzewa z korzeniem tego podrzewa może być związany zbiór przykładów uczących należących do różnych klas decyzyjnych. Zmieniając taki węzeł na liść decyzyjny przypisuje mu się etykietę większościowej klasy decyzyjnej w zbiorze przykładów związanym z tym węzłem.

Upraszczanie (przycinanie drzewa)

- Metody upraszczania:
  - upraszczanie wstępne
  - upraszczanie w pełni zbudowanego drzewa

Algorytm k-NN

- *k*-NN (*k* nearest neighbors) - algorytm *k* najbliższych sąsiadów jest algorytmem używanym do klasyfikacji nowych obiektów (przypadków).
- Nowy obiekt porównywany jest z *k* najbliższymi sąsiadami ze zbioru uczącego.
- Nowemu obiektowi przypisywana jest większościowa klasa sąsiadów ze zbioru uczącego.

Algorytm k-NN