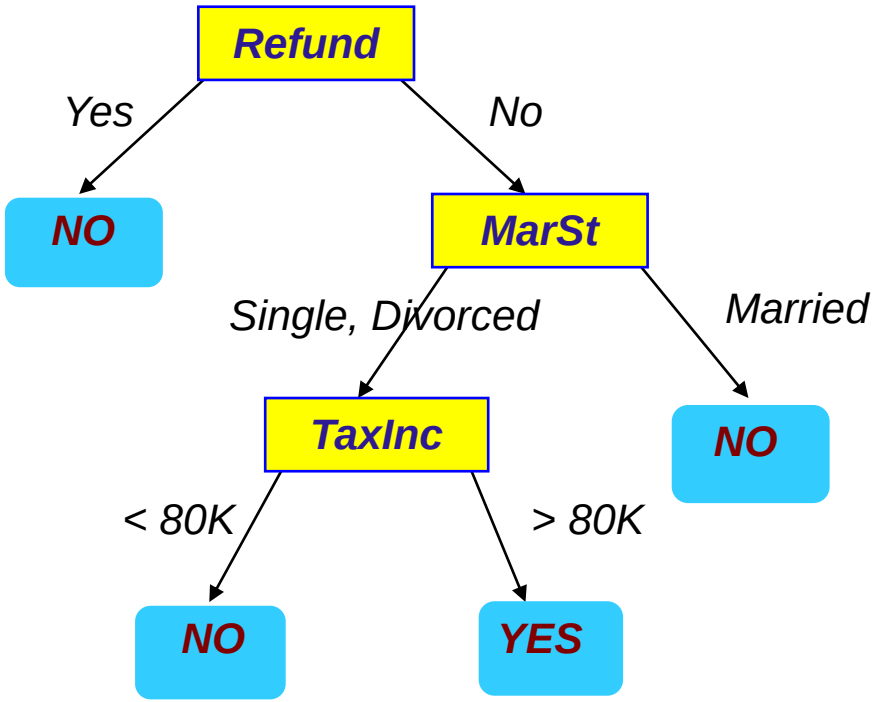


Drzewa klasyfikacyjne

Agnieszka Nowak – Brzezińska

Zadania sztucznej inteligencji

- Klasyfikacja, predykcja, przewidywanie – polega ona na znajdowaniu odwzorowania danych w zbiór predefiniowanych klas. Na podstawie zawartości bazy danych budowany jest model (np. drzewo decyzyjne, reguły logiczne), który służy do klasyfikowania nowych obiektów w bazie danych lub głębszego zrozumienia istniejącego podziału obiektów na predefiniowane klasy.
- Ogromne zastosowanie znalazła w systemach medycznych, przykładowo, w bazie danych medycznych znalezione mogą być reguły klasyfikujące poszczególne schorzenia, a następnie przy pomocy znalezionych reguł automatycznie może być przeprowadzone diagnozowanie kolejnych pacjentów.
- Klasyfikacja jest metodą eksploracji danych z nadzorem (z nauczycielem). Proces klasyfikacji składa się z kilku etapów – budowania modelu, po czym następuje faza testowania oraz predykcji nieznanych wartości.

Klasyfikacja danych jest dwu-etapowym procesem:

Etap 1:

- budowa modelu (klasyfikatora) opisującego predefiniowany zbiór klas danych lub zbiór pojęć

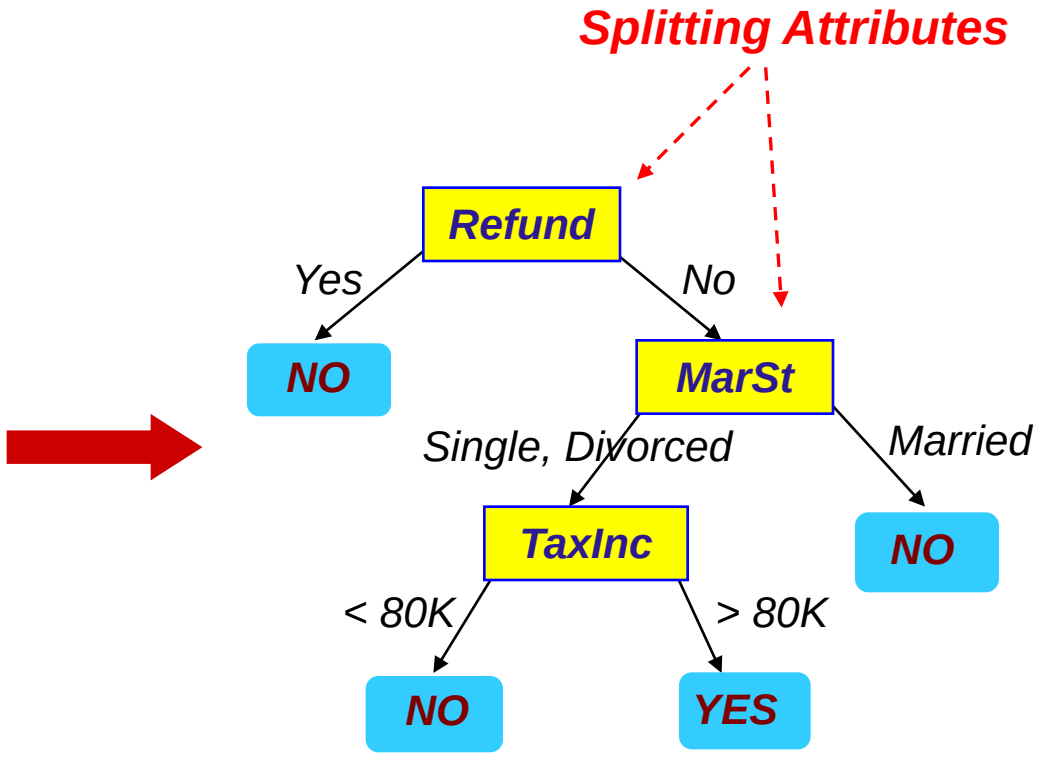
Etap 2:

- zastosowanie opracowanego modelu do klasyfikacji nowych danych

Przykład

Tid	Refund	Marital Status	Taxable Income	Cheat
	categorical	*categorical*	*continuous*	*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

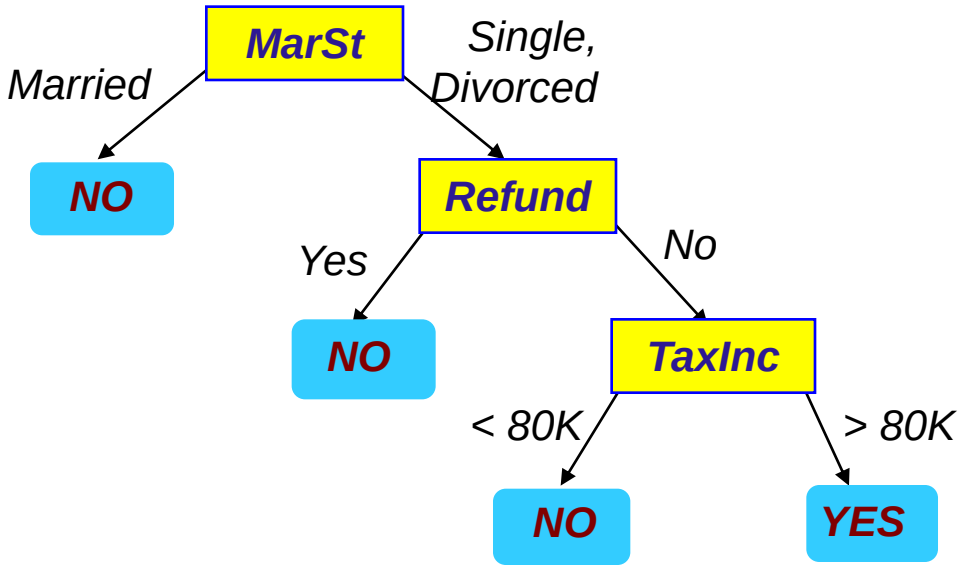
Dane treningowe



Model: Decision Tree

Inne drzewo decyzyjne

	categorical	categorical	continuous	class
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**



Drzew różnych dla tego samego zbioru danych może być wiele!

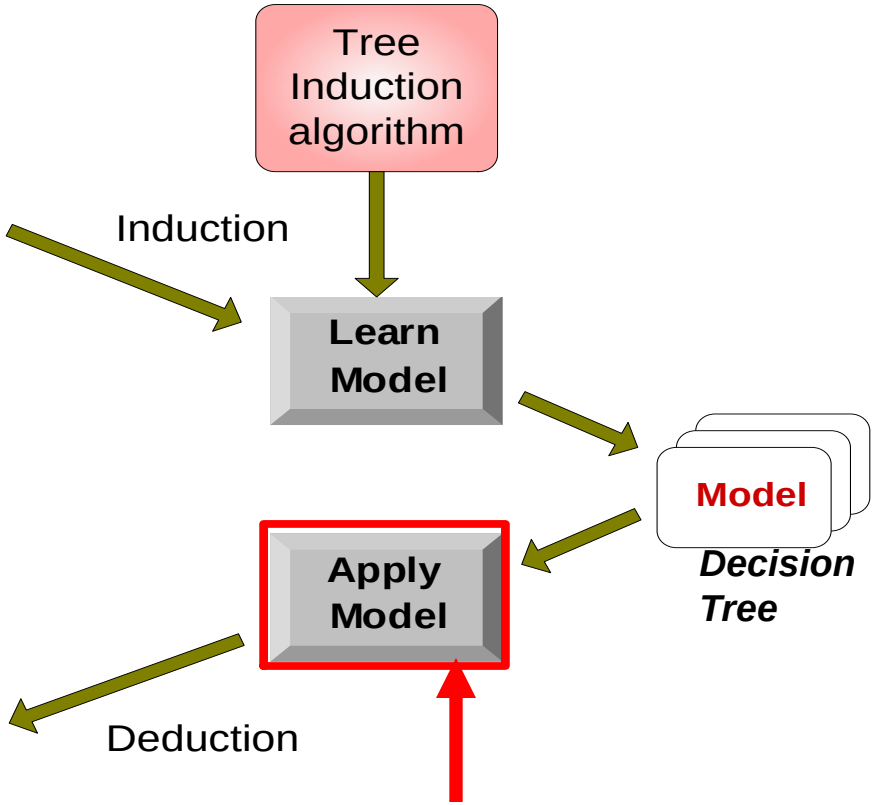
Zadanie drzew klasyfikacyjnych

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

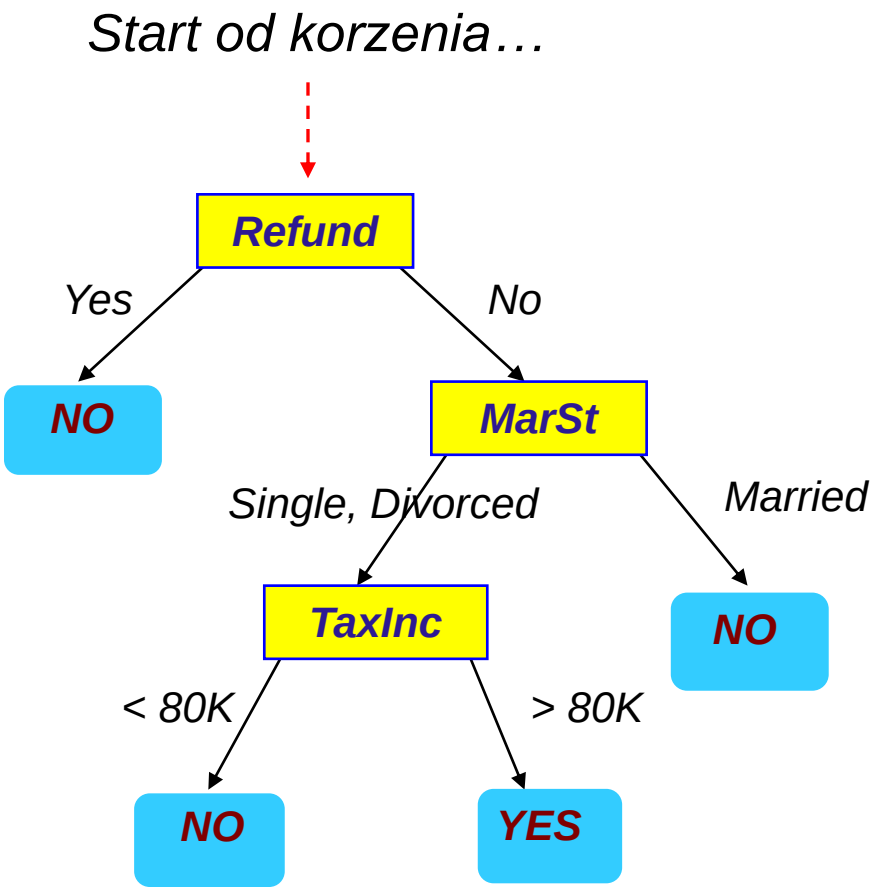
Training Set

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



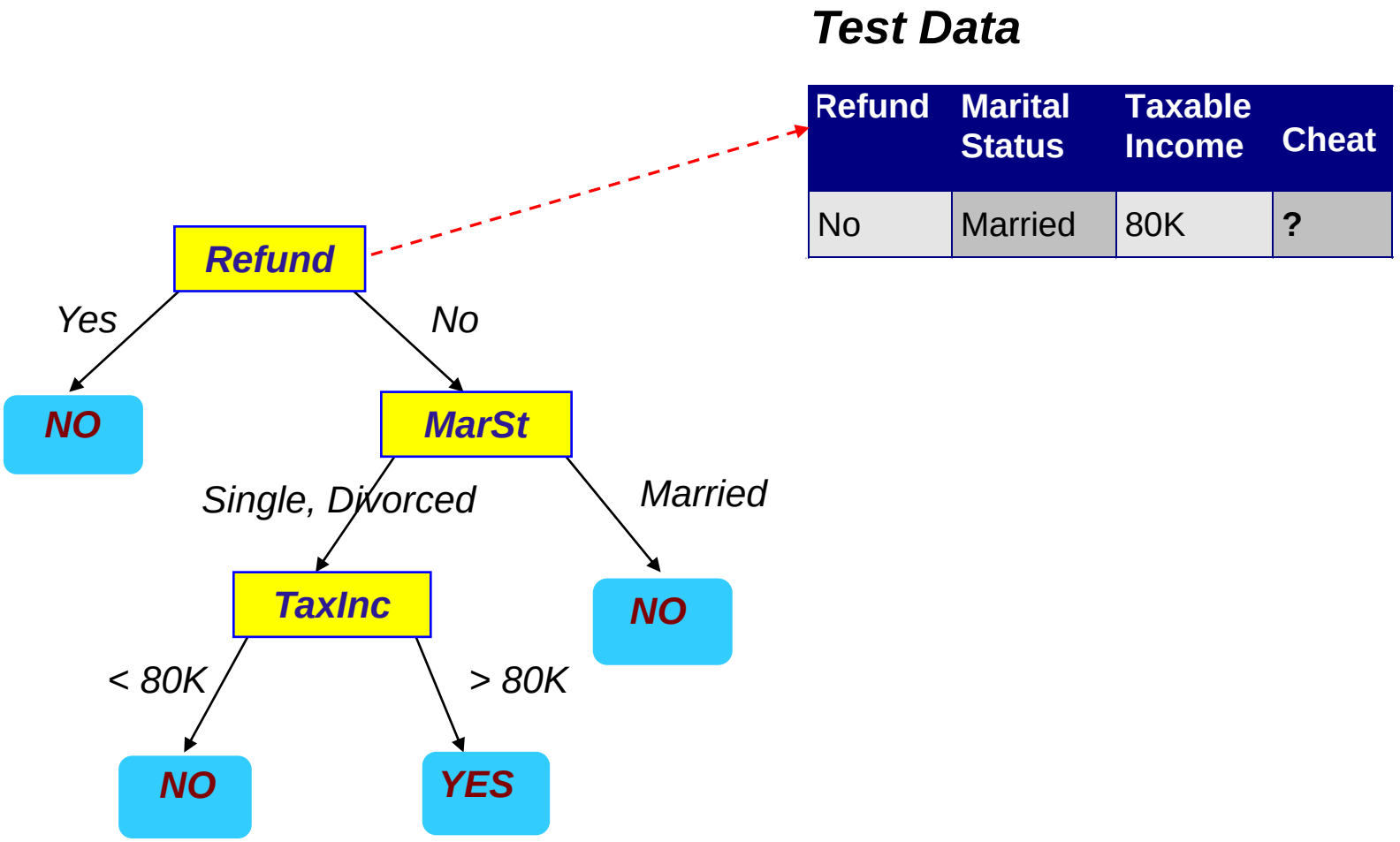
Zastosowanie modelu do danych…



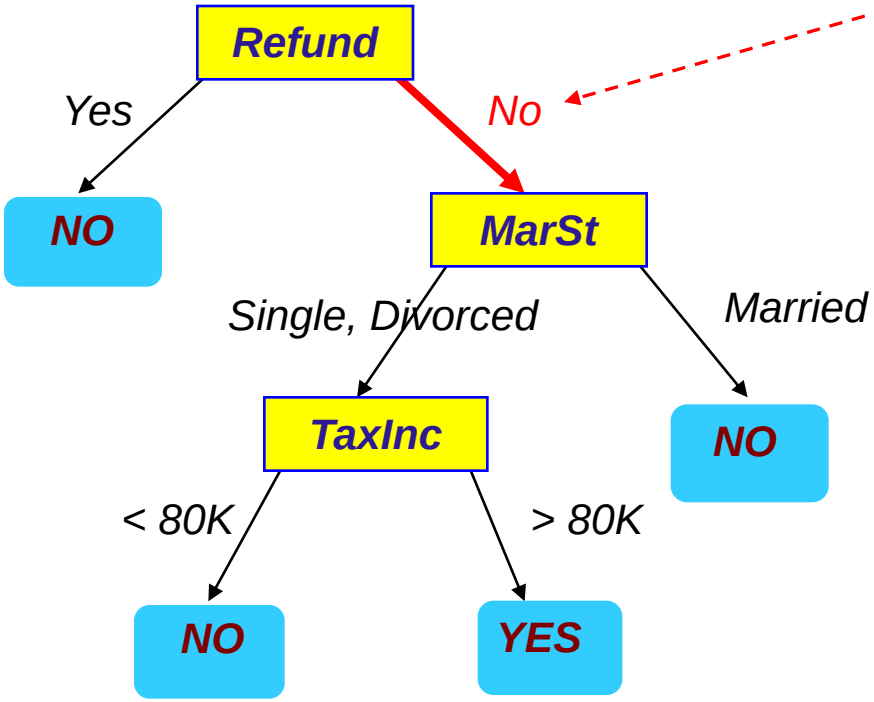
Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	**?**

Zastosowanie modelu do danych…



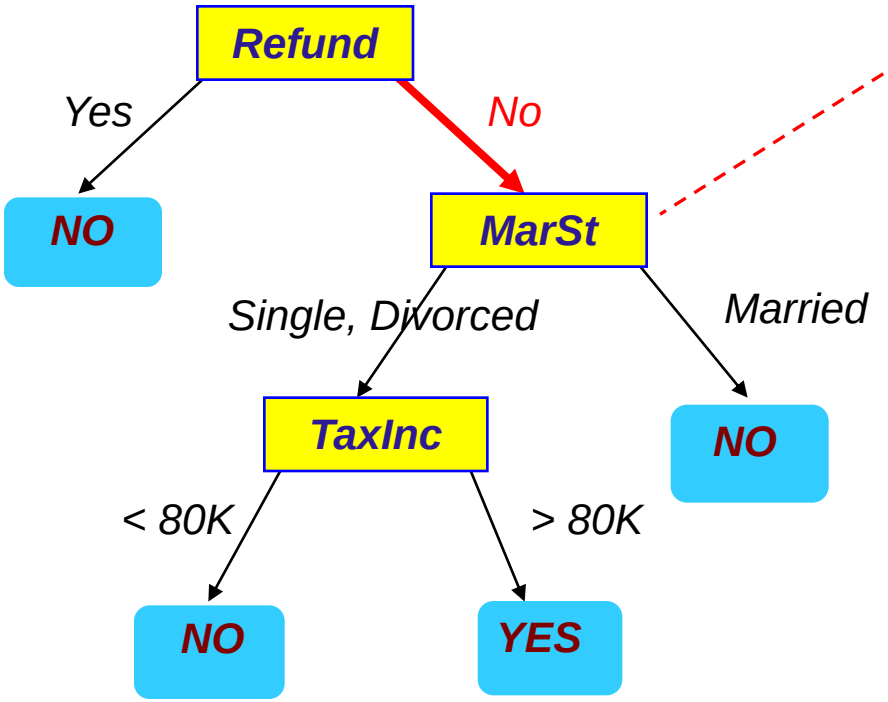
Zastosowanie modelu do danych…



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

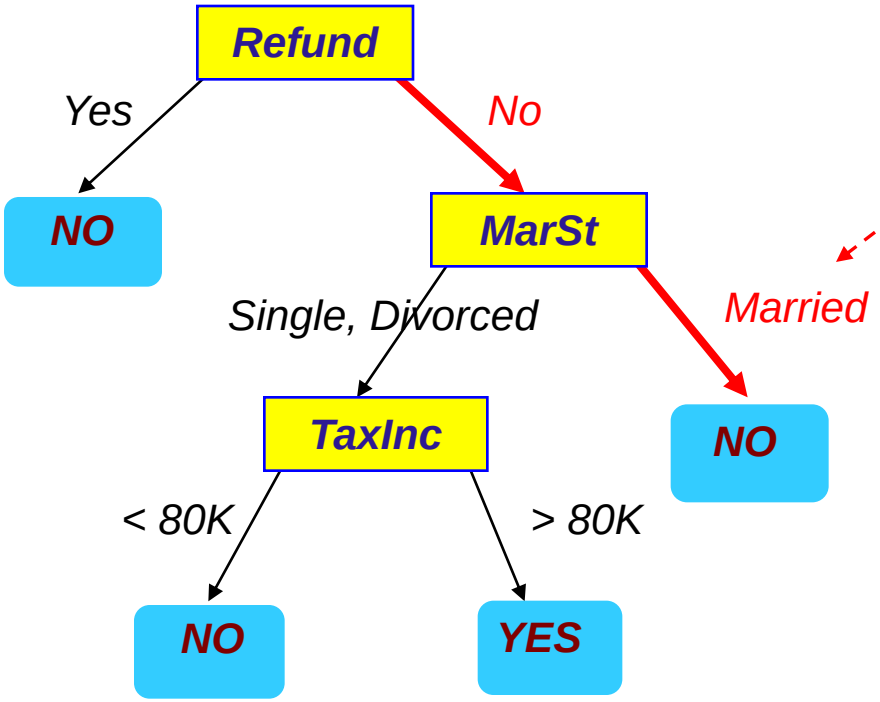
Zastosowanie modelu do danych…



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

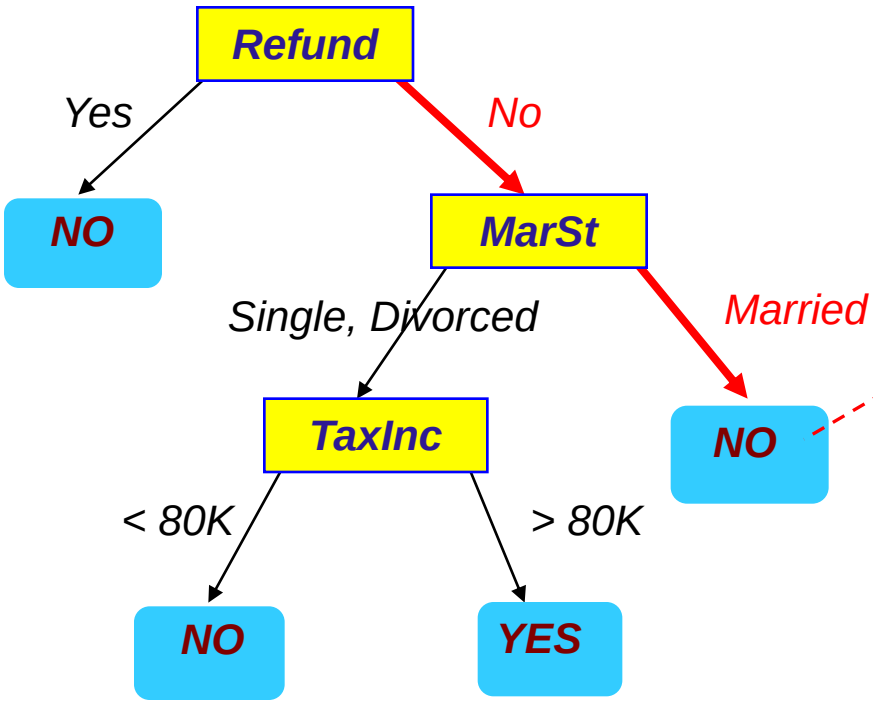
Zastosowanie modelu do danych…



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

Zastosowanie modelu do danych…



Test Data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

"No"

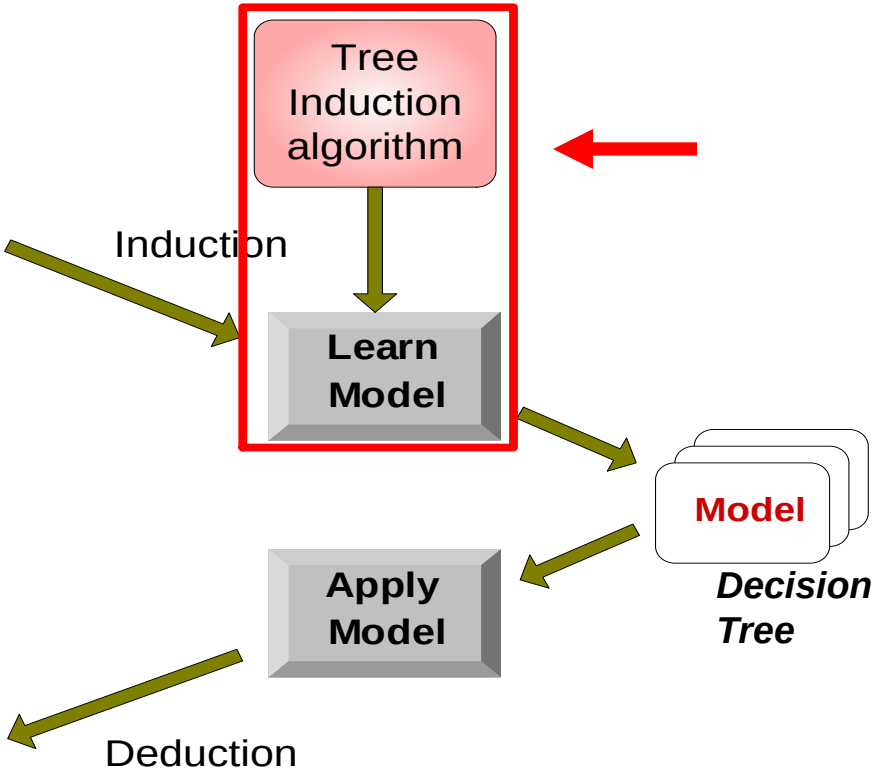
Zadanie klasyfikacji DD

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

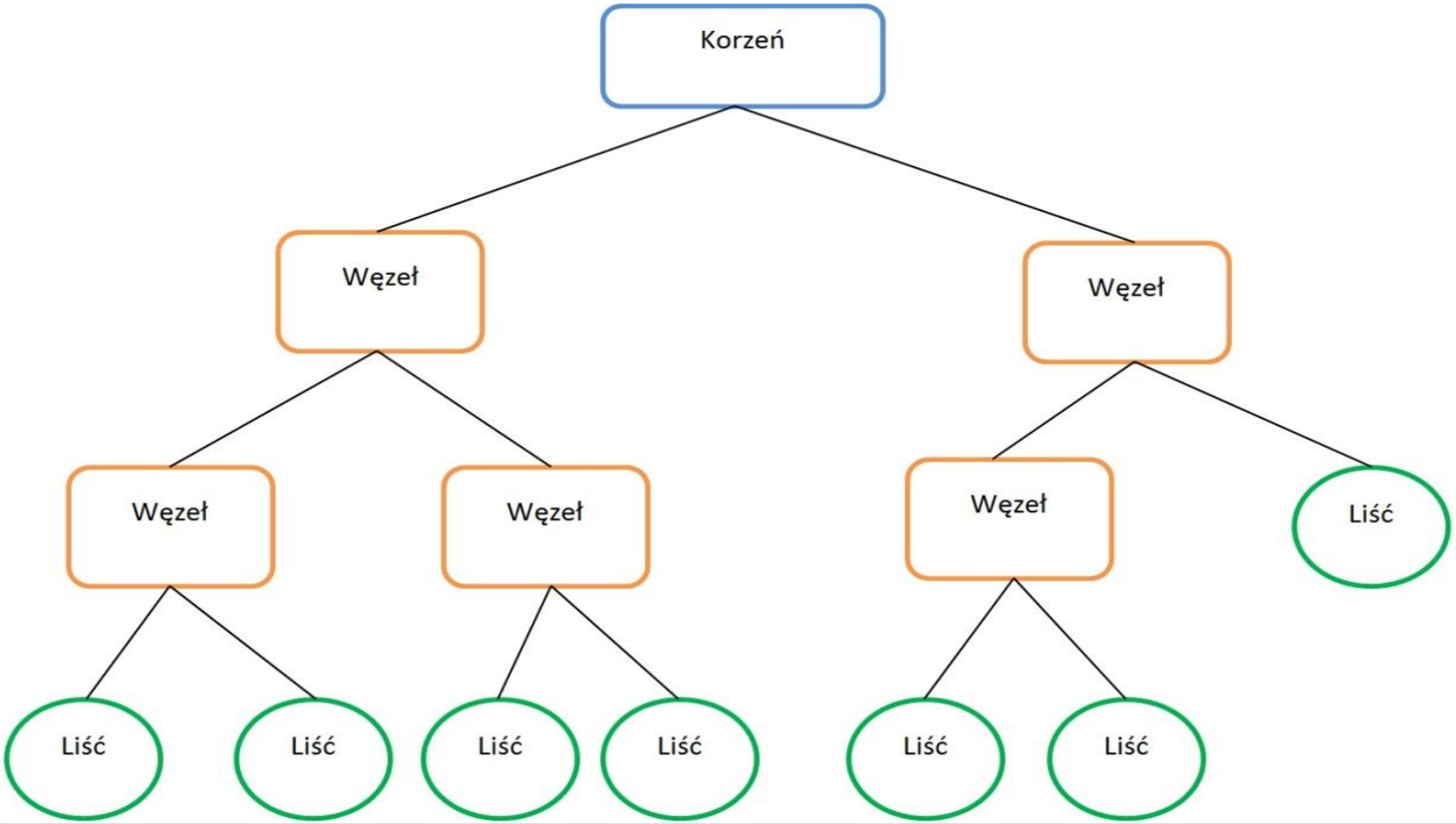


- W informatyce pod pojęciem drzewa rozumiemy spójny skierowany graf acykliczny, z którym związane jest następujące nazewnictwo:
- krawędzie grafu nazywane są gałęziami;
- wierzchołki, z których wychodzi co najmniej jedna krawędź to węzły, pozostałe zaś określane są mianem liści. Przyjmuje sie, ze wierzchołek ma co najwyżej jedną krawędź wchodzącą, a liczba krawędzi wychodzących jest równa 0 albo większa od 1;
- węzeł lub liść do którego nie prowadza żadne krawędzie to korzeń.

Definicja

- Drzewo decyzyjne jest drzewem, w którym węzły odpowiadają testom przeprowadzanym na wartościach atrybutów reguł, gałęzie są możliwymi wynikami takich testów zaś liście reprezentują część decyzyjną.
- Drzewa decyzyjne jako forma reprezentacji zbioru reguł zdobyły wśród programistów SI dużą popularność przede wszystkim ze względu na swą czytelność oraz znaczne zmniejszenie wymaganych do przechowania bazy, zasobów pamięci. Fakt ten zilustrowany jest tablica poniżej zawierającą przykładowa bazę reguł, oraz rysunkiem przedstawiającym odpowiadające jej drzewo decyzyjne.

Drzewo decyzyjne - schemat



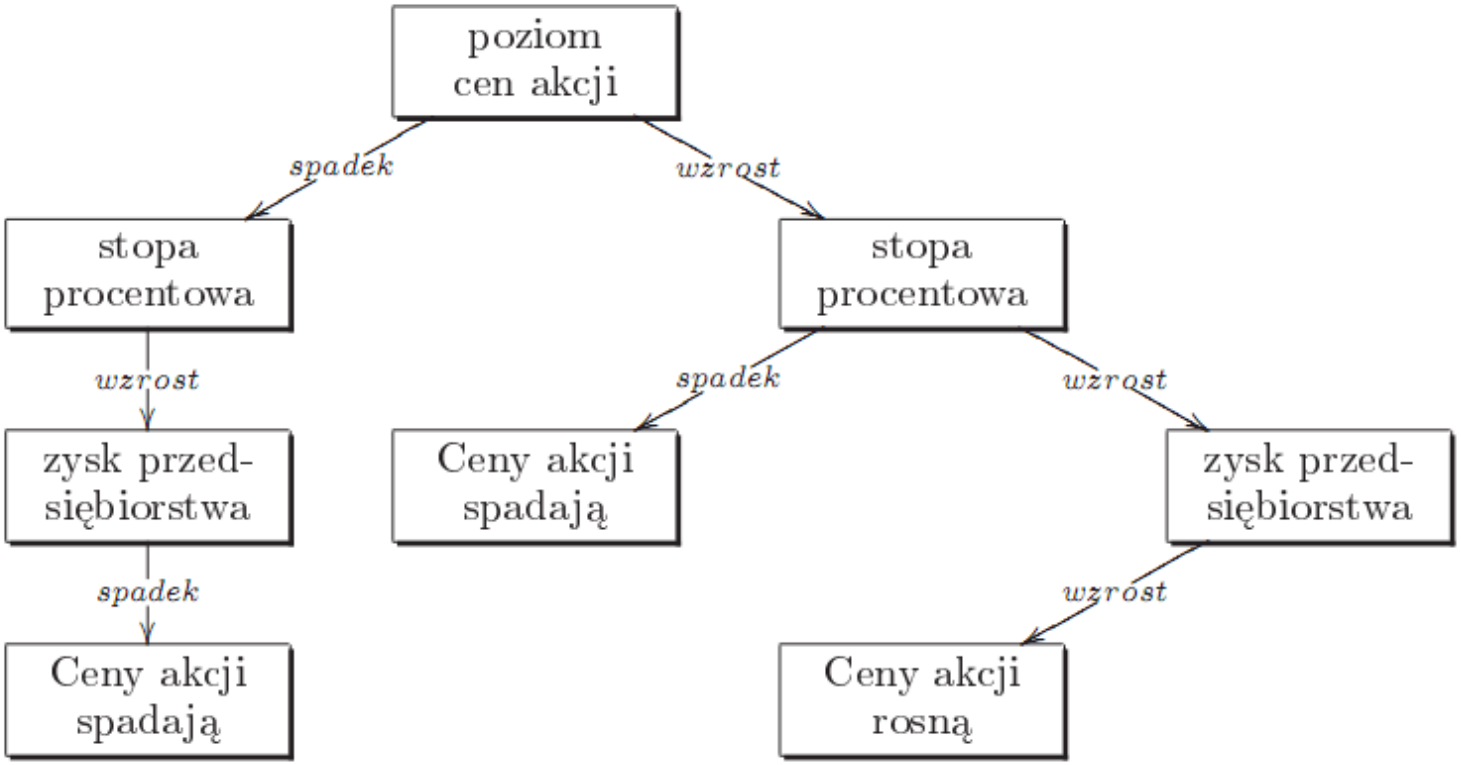
Definicje

- Drzewo to graf bez cykli (pętli), w którym istnieje tylko jedna ścieżka między dwoma różnymi węzłami.
- Czyli, to graf spójny bez cykli.
- To drzewo reprezentujące pewien proces podziału zbioru obiektów na jednorodne klasy. Jego wewnętrzne węzły opisują sposób dokonania tego podziału, a liście odpowiadają klasom, do których należą obiekty. Z kolei krawędzie drzewa reprezentują wartości cech, na podstawie których dokonano podziału.

Drzewo klasyfikacyjne

- Składa się z korzenia, z którego wychodzą co najmniej dwie krawędzie do węzłów leżących na niższym poziomie.
- Z każdym węzłem związane jest pytanie o wartości cech, jeśli np. pewien obiekt posiada te wartości, przenosi się go w dół odpowiednią krawędzią.
- Węzły, z których nie wychodzą już żadne krawędzie, to liście, które reprezentują klasy.

Drzewo to reguły decyzyjne



W przypadku spadku cen akcji:

"jeżeli stopa procentowa rośnie i zyski przedsiębiorstw spadają to ceny akcji spadają"

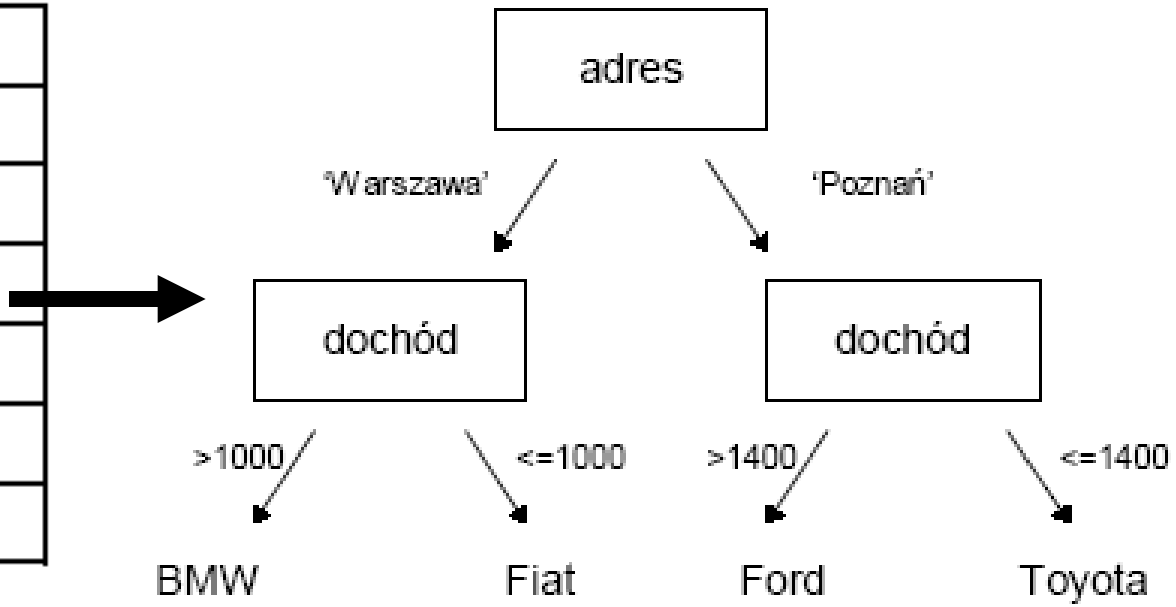
w przypadku wzrostu cen akcji:

"jeżeli stopa procentowa spada, lub jeśli stopa procentowa rośnie ale jednocześnie rosną zyski przedsiębiorstw rosną to ceny akcji rosną."

Drzewa decyzyjne

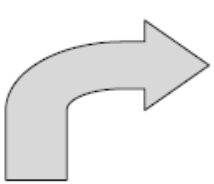
Najpowszechniejsza forma reprezentowania wiedzy przez dostępne oprogramowanie. Węzły są opisywane przez atrybuty eksplorowanej relacji, krawędzie opisują możliwe wartości dla atrybutu. Klasyfikacja odbywa się poprzez przeglądanie drzewa od korzenia do liści przez krawędzie opisane wartościami atrybutów.

Adres	Dochód	Samochód
Warszawa	4000	BMW
Poznań	2900	Ford
Poznań	1400	Toyota
Warszawa	1000	Fiat
Poznań	1600	Ford
Poznań	3500	Ford



Drzewa decyzyjne

Przykład: automatyczny podział kierowców na powodujących i nie powodujących wypadki drogowe Dwie klasy: TAK i NIE



Zbiór uczący

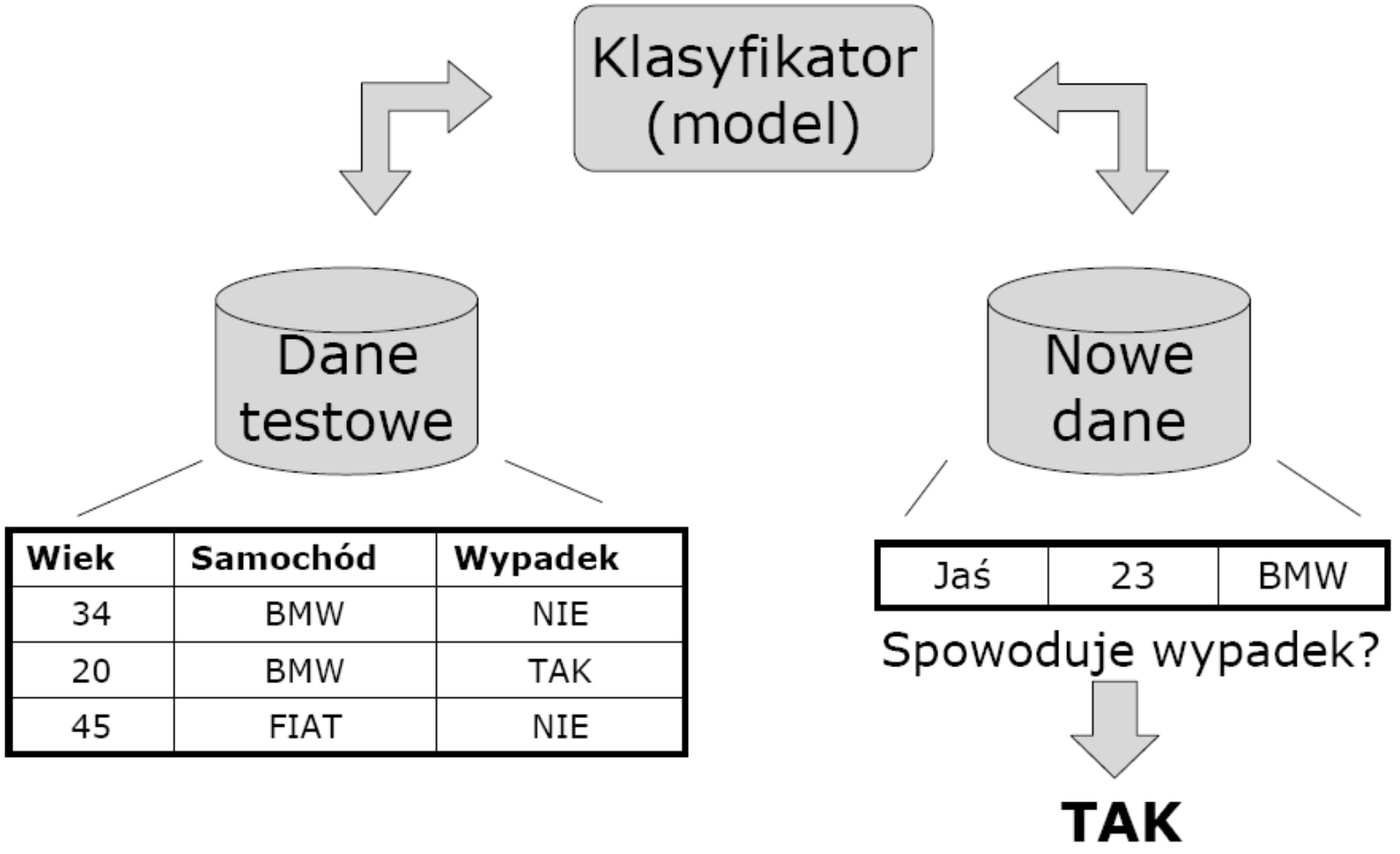
Wiek	Samochód	Wypadek
24	BMW	TAK
22	BMW	TAK
45	FIAT	NIE
20	FIAT	NIE
50	BMW	NIE

Algorytm uczenia się

Klasyfikator (model)

IF marka = BMW
AND wiek_kier < 25
THEN wypadek = TAK

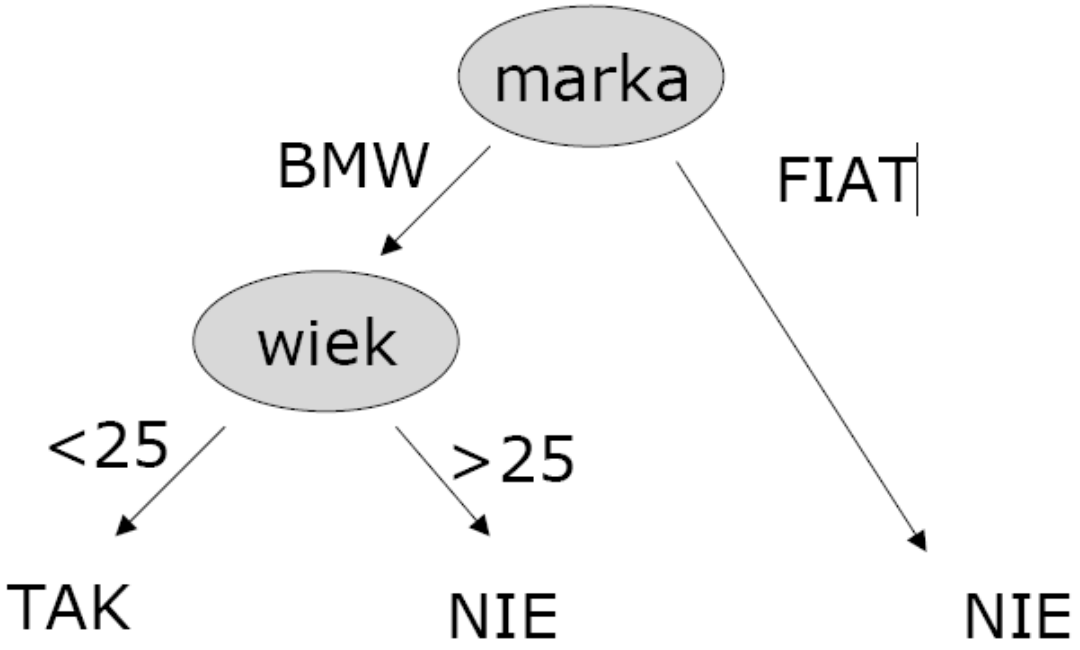
Drzewa decyzyjne




```
IF marka = BMW AND wiek_kier < 25 THEN wypadek = TAK
```

Drzewa decyzyjne

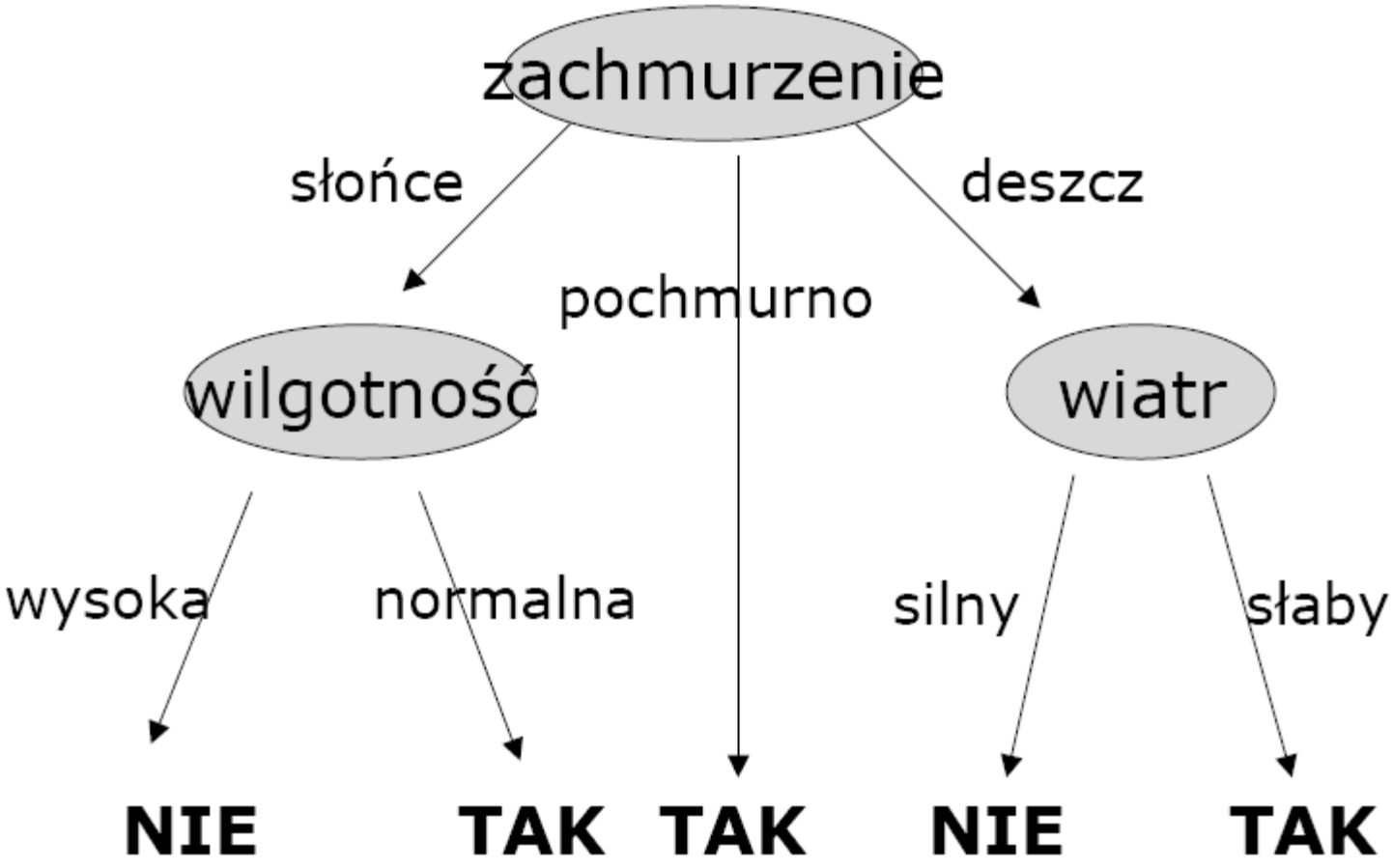
- Drzewo decyzyjne jest formą opisu wiedzy klasyfikującej
- Węzłom drzewa odpowiadają atrybuty eksplorowanej relacji
- Krawędzie opisują wartości atrybutów
- Liśćmi drzewa są wartości atrybutu klasyfikacyjnego



Drzewa decyzyjne - przykład

zachmurzenie	temperatura	wilgotność	wiatr	decyzja
słońce	gorąco	wysoka	słaby	nie
słońce	gorąco	wysoka	silny	nie
pochmurno	gorąco	wysoka	słaby	tak
deszcz	średnio	wysoka	słaby	tak
deszcz	chłodno	normalna	słaby	tak
deszcz	chłodno	normalna	silny	nie
pochmurno	chłodno	normalna	silny	tak
słońce	średnio	wysoka	słaby	nie
słońce	chłodno	normalna	słaby	tak
deszcz	średnio	normalna	słaby	tak
słońce	średnio	normalna	silny	tak
pochmurno	średnio	wysoka	silny	tak
pochmurno	gorąco	normalna	słaby	tak
deszcz	średnio	wysoka	silny	nie

Drzewa decyzyjne dla przykładu



Drzewa decyzyjne – indukcja drzewa

Problemy:

- Od jakiego atrybutu zacząć? Jakie atrybuty umieszczać w węzłach?
- Kiedy zaprzestać rozbudowywać drzewo?
- Co z atrybutami o zbyt dużej liczbie wartości lub atrybutami ilościowymi?
- Jak uwzględniać dane sprzeczne, brakujące lub błędne?

Estymacja błędu klasyfikowania

n – liczba wszystkich klasyfikowanych obiektów

b – liczba błędnie sklasyfikowanych obiektów

Błąd klasyfikowania:

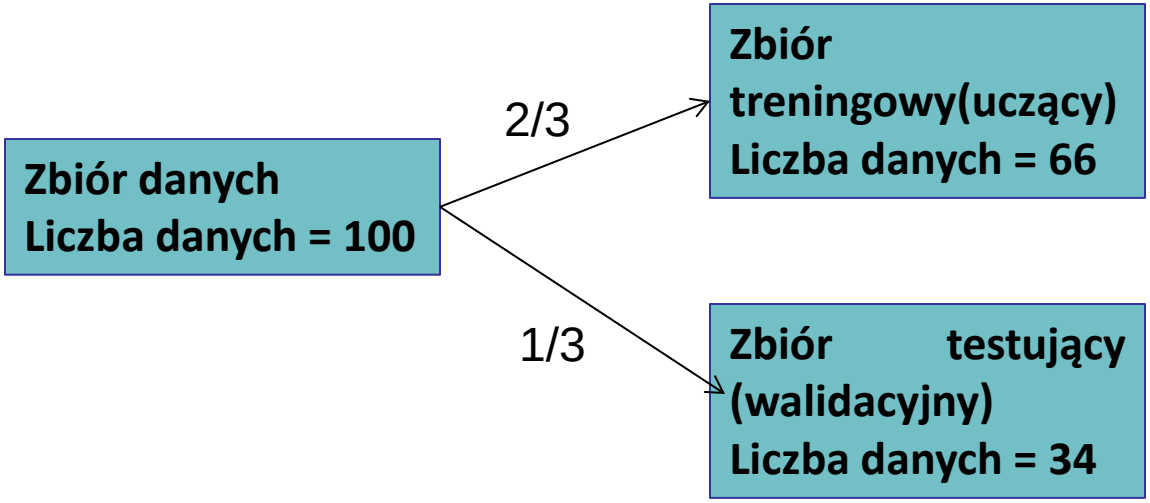
$$e = \frac{b}{n}$$

Dokładność klasyfikowania:

$$1 - e$$

Trening i testowanie

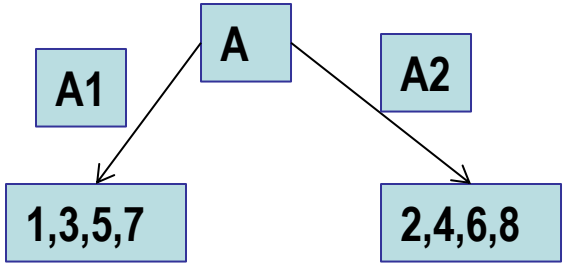
- Zbiór dostępnych krotek(przykładów, obserwacji, próbek) dzielimy na dwa zbiory: zbiór treningowy i zbiór testowy
- Model klasyfikacyjny (klasyfikator)jest budowany dwu-etapowo:
  - Uczenie (trening) –klasyfikator jest budowany w oparciu o zbiór treningowy danych
  - Testowanie –dokładność (jakość)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



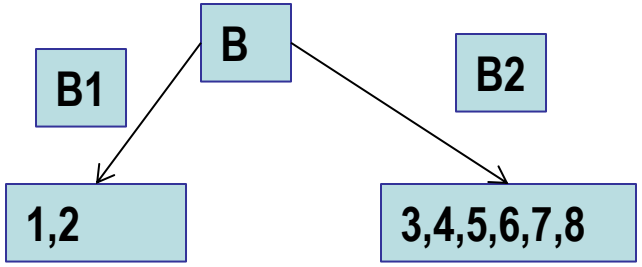
Podział obiektów ze względu na wartości atrybutów

- Każdy atrybut dzieli obiekty w systemie na grupy.
- Grup jest tyle ile wartości atrybutu, który analizujemy.
- W grupie dla danej wartości są tylko obiekty opisane tą wartością atrybutu.

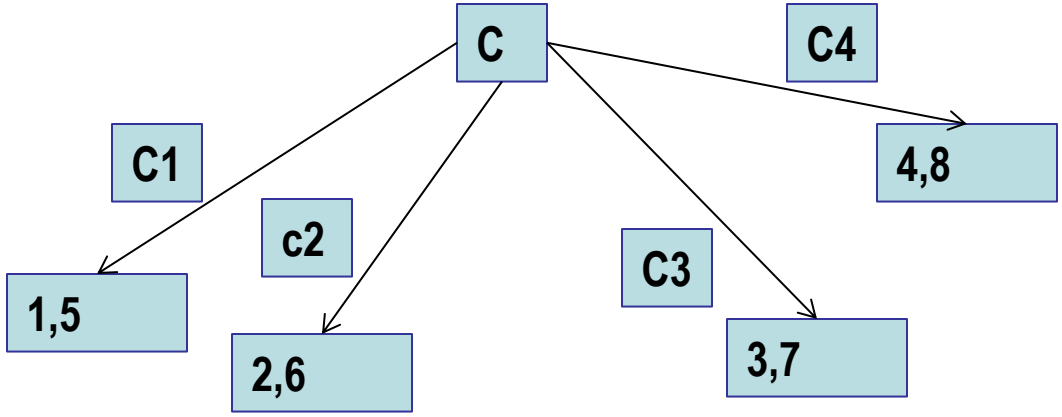
	A	B	C
1	**A1**	B1	C1
2	**A2**	B1	C2
3	**A1**	B2	C3
4	**A2**	B2	C4
5	**A1**	B2	C1
6	**A2**	B2	C2
7	**A1**	B2	C3
8	**A2**	B2	C4



	A	B	C
1	**A1**	B1	C1
2	**A2**	B1	C2
3	**A1**	B2	C3
4	**A2**	B2	C4
5	**A1**	B2	C1
6	**A2**	B2	C2
7	**A1**	B2	C3
8	**A2**	B2	C4



	A	B	C
1	**A1**	B1	C1
2	**A2**	B1	C2
3	**A1**	B2	C3
4	**A2**	B2	C4
5	**A1**	B2	C1
6	**A2**	B2	C2
7	**A1**	B2	C3
8	**A2**	B2	C4



Podział cech (atrybutów)

- **Ilościowe** – numeryczne np. liczba języków obcych, waga w kg, wzrost w cm, wiek w latach etc.
- **Jakościowe** (**nominalne**) to cechy nienumeryczne, nie da się z nich policzyć wartości średniej, ale da się np. powiedzieć która wartość tej cechy występuje najczęściej, np. waga w kategoriach {niska, średnia, wysoka}, kolor oczu, wykształcenie, płeć z wartościami {kobieta, mężczyzna}

Algorytm generowania drzewa decyzyjnego

Algorytm „top-down” dla zbioru uczącego S

1. Wybierz najlepszy atrybut A

2. Rozbuduj drzewo poprzez dodanie do węzła

nowych gałęzi odpowiadającym poszczególnym wartościom atrybutu A

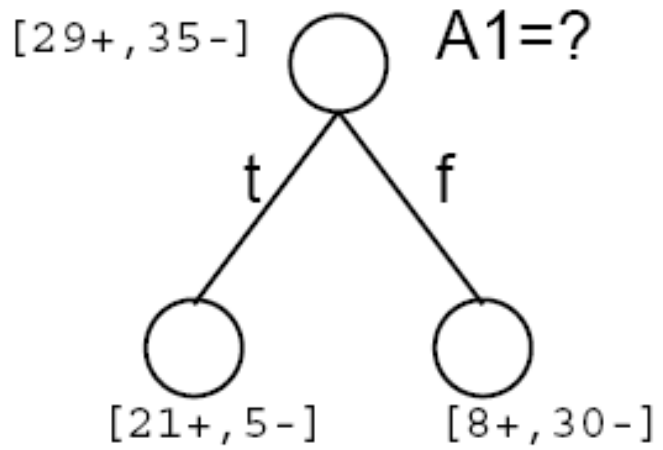
3. Podziel zbiór S na podzbiory S1, ... , Sn

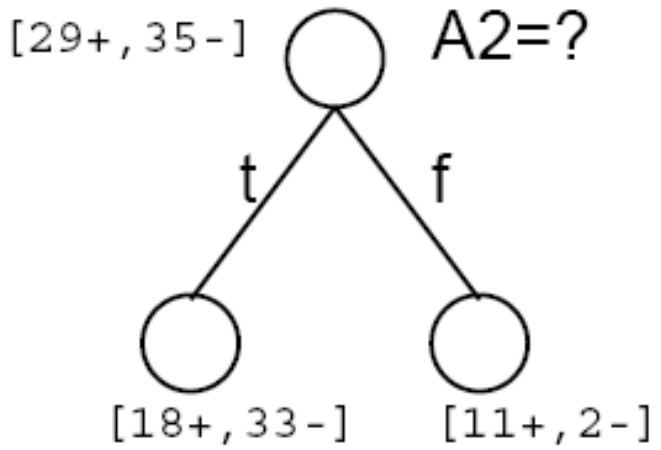
zgodnie z wartościami atrybutu A i przydziel do odpowiednich gałęzi

4. Jeśli wszystkie przykłady Si należą do tej samej klasy C zakończ gałąź liściem wskazującym C; w przeciwnym razie rekurencyjnie buduj drzewo Ti dla Si (tzn. powtórz kroki 1-4)

Wybór atrybutu do budowania drzewa

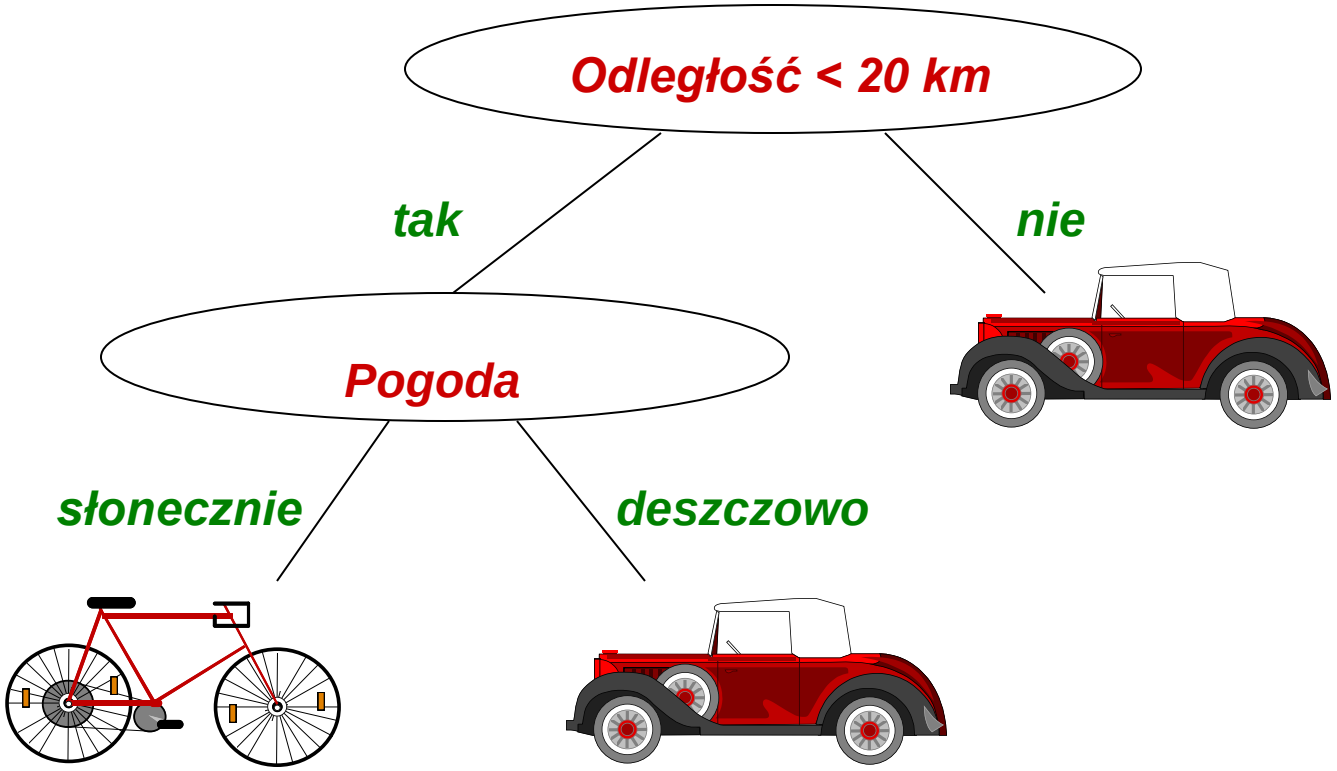
Rożne atrybuty dają różne rozkłady decyzji w gałęziach





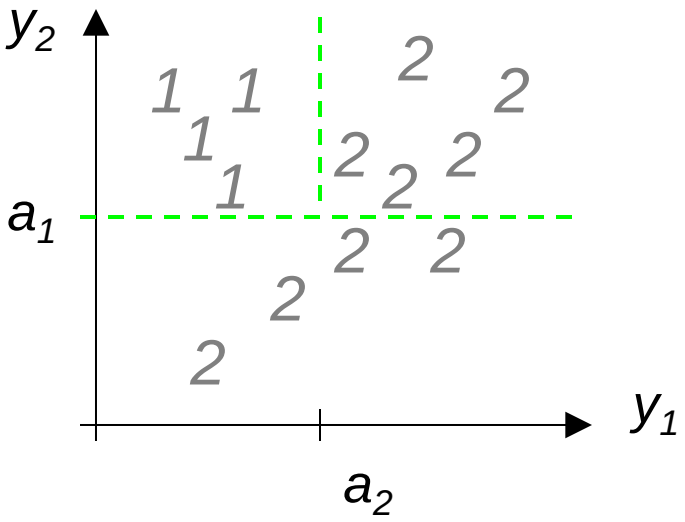
Funkcja CHOOSE-ATTRIBUTE wybiera najlepszy z nich

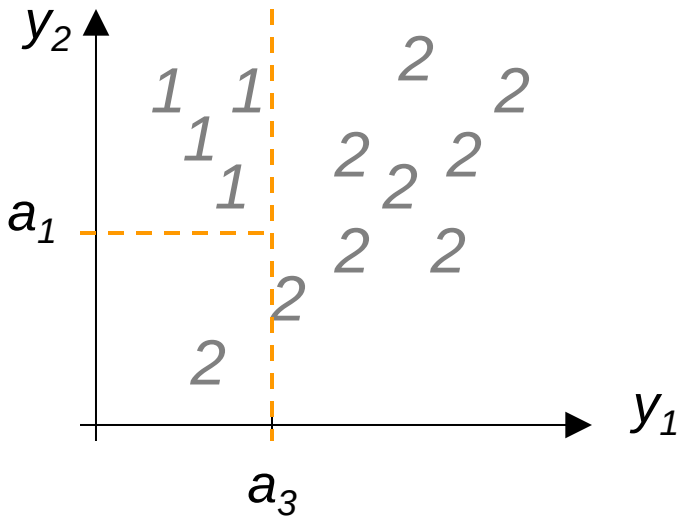
Drzewo decyzyjne

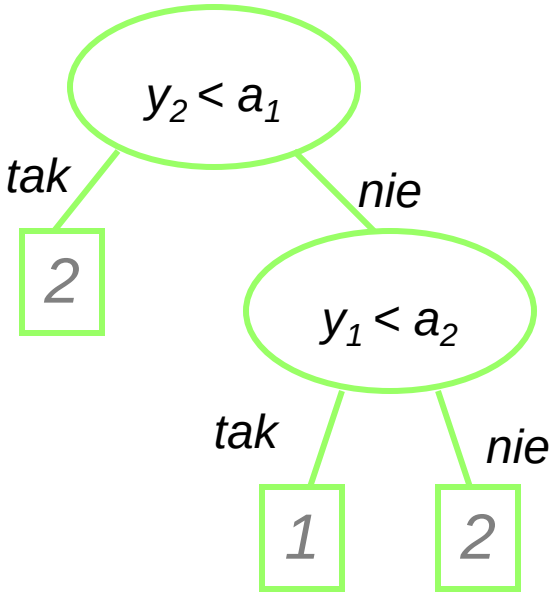


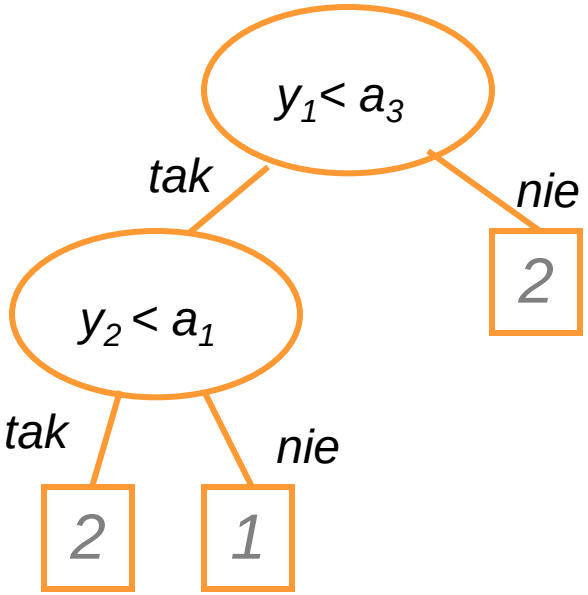
Pojęcia: korzeń drzewa, węzeł wewnętrzny, węzeł końcowy (liść), gałąź, ścieżka.

Konstrukcja drzewa decyzyjnego









Konstrukcja drzew decyzyjnych

Jeden zbiór danych $\rightarrow$ wiele możliwych drzew

Czym należy się kierować wybierając (konstruując) drzewo?

Kryteria optymalizacji

Globalne

- *średnie prawdopodobieństwo błędu*
- *średnia długość ścieżki*
- *liczba węzłów drzewa*

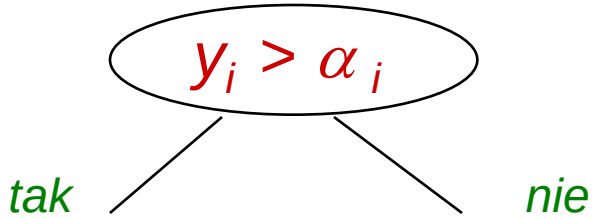
Lokalne

- *stopień zróżnicowania danych*
- *przyrost informacji*
- *współczynnik przyrostu informacji*

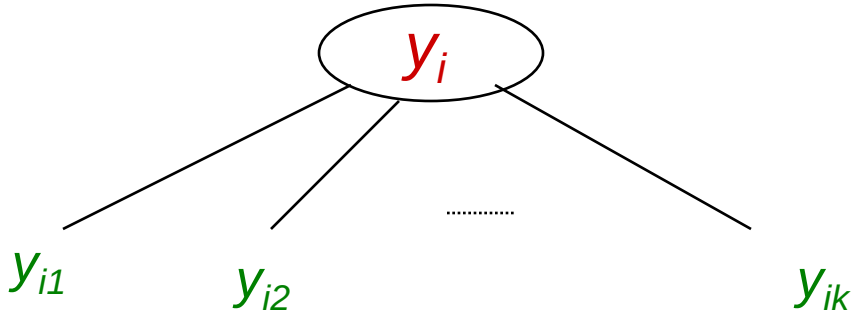
i inne

Podział węzła - przykłady

1. Cecha porównana z wartością progową (typowe dla atrybutów ciągłych).



2. Uwzględnione wszystkie możliwe wartości danego atrybutu (typowe dla atrybutów nominalnych).



Podział węzła

Najczęściej reguły decyzyjne budowane są na podstawie pojedynczych cech źródłowych. Prowadzi to do dzielenia przestrzeni cech hiperłaszczyznami prostopadłymi do osi cech.

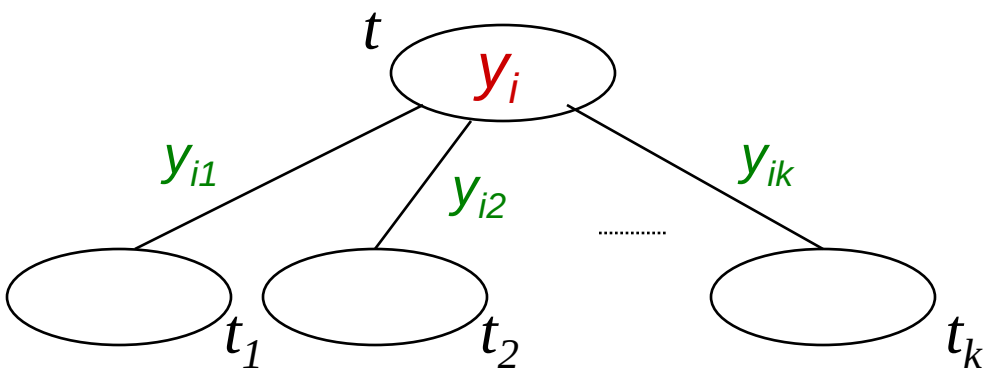
Wybierając cechę można się kierować jedną ze znanych miar, np. przyrostem informacji, wskaźnikiem przyrostu informacji, wskaźnikiem zróżnicowania danych itd.

Podział węzła w przypadku atrybutów nominalnych

1. Dla każdego atrybutu y_i oblicz wartość wybranej miary.

2. Wybierz atrybut optymalny w sensie powyższej miary.

3. Od danego węzła utwórz tyle gałęzi, ile różnych wartości przyjmuje atrybut y_i.



Entropia

Entropia w ramach teorii informacji jest definiowana jako średnia ilość informacji, przypadająca na znak symbolizujący zajście zdarzenia z pewnego zbioru. Zdarzenia w tym zbiorze mają przypisane prawdopodobieństwa wystąpienia.

$$H(x) = \sum_{i=1}^{n} p(i) \log_r \frac{1}{p(i)} = -\sum_{i=1}^{n} p(i) \log_r p(i)$$

gdzie *p(i) - prawdopodobieństwo zajścia zdarzenia i.*

Logarytm

Logarytm przy podstawie „**a**” z liczby „**b**” , zapisywany „$\log_a b$” to taka liczba „**c**”, że podstawa „**a**” podniesiona do potęgi „**c**” daje logarytmowaną liczbę „**b**”.
Symbolicznie:

$$\log_a b = c \leftrightarrow a^c = b$$

gdzie $a>0, a\neq 1$ oraz $b > 0$.

Na przykład $\log_2 8 = 3$, ponieważ $2^3 = 8$.

Entropia

Entropię można interpretować jako niepewność wystąpienia danego zdarzenia elementarnego w następnej chwili. Jeżeli następujące zdarzenie występuje z prawdopodobieństwem równym 1, to z prostego podstawienia wynika, że entropia wynosi 0, gdyż z góry wiadomo co się stanie - nie ma niepewności.

Własności entropii:

- jest nieujemna
- jest maksymalna, gdy prawdopodobieństwa zajść zdarzeń są takie same
- jest równa 0, gdy stany systemu przyjmują wartości 0 albo 1

własność superpozycji - gdy dwa systemy są niezależne to entropia sumy systemów równa się sumie entropii.

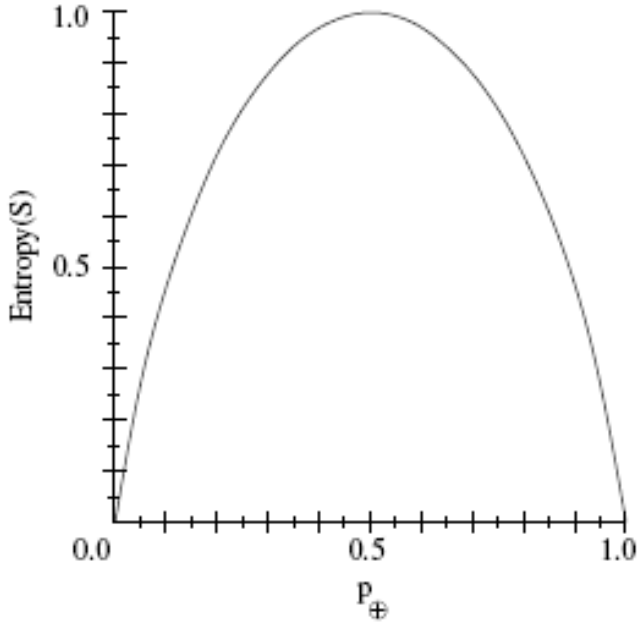
Entropia – dwie decyzje

Dane są dwie decyzje: pozytywna ($\oplus$) i negatywna ($\ominus$)

$p_\oplus = \frac{|S_\oplus|}{|S|}$ — proporcja obiektów z decyzją pozytywną w zbiorze S

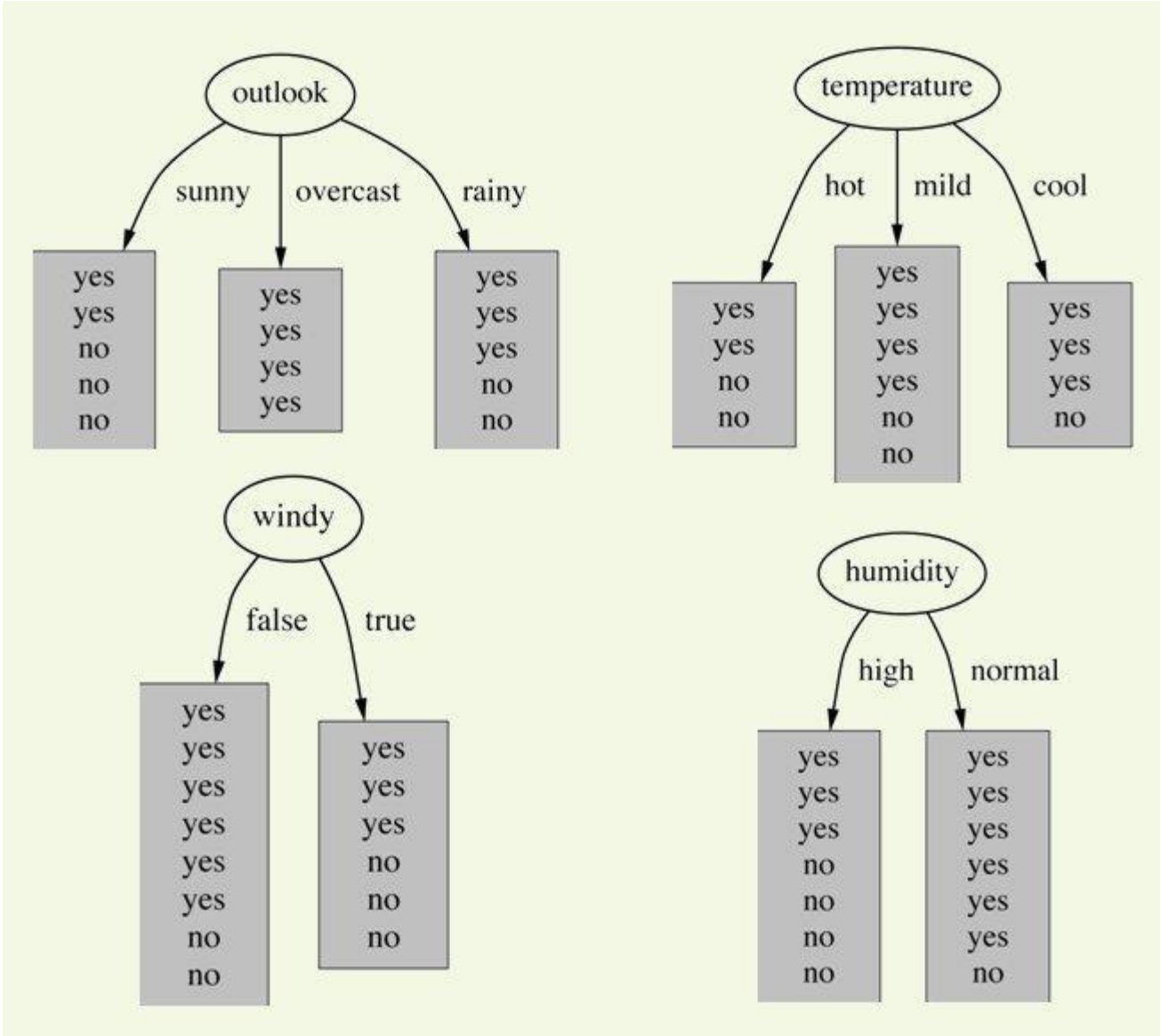
$p_\ominus = \frac{|S_\ominus|}{|S|}$ — proporcja obiektów z decyzją negatywną w zbiorze S

$$Entropy(S) = -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$



Outlook	Tempreature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P

Przykład: który atrybut wybrać?



Ile informacji zawiera dany podział ?

Średnia liczba bitów do zakodowania dowolnego wektora wynosi:

Outlook = sunny

info([2,3]) = entropy(2/5,3/5) = -2 / 5log(2 / 5) - 3/ 5log(3/ 5) = 0.971bits

Outlook = overcast

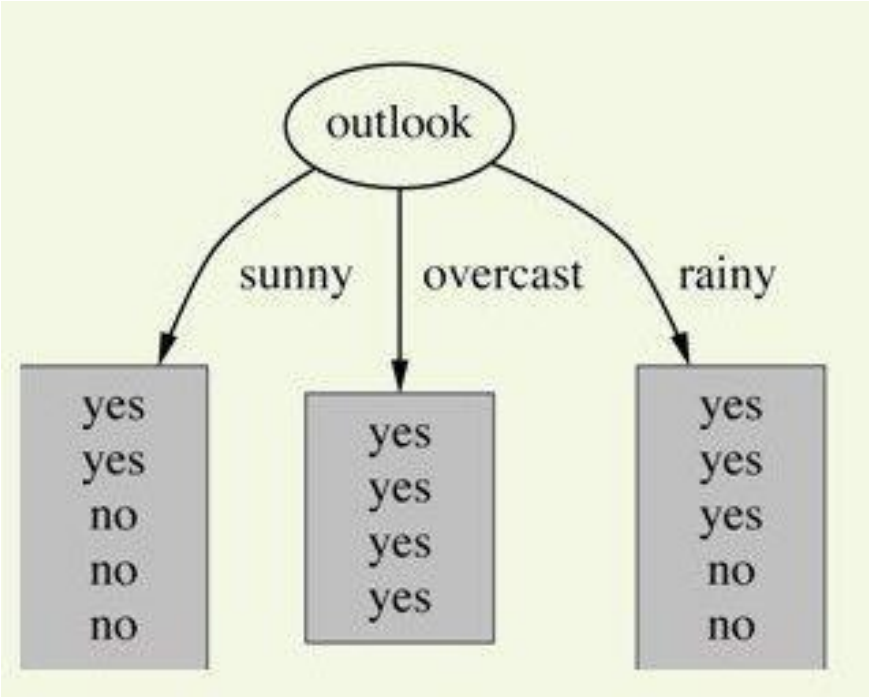
info([4,0]) = entropy(1,0) = -1log(1) - 0log(0) = 0 bits

Outlook = rainy

info([3,2]) = entropy(3/5,2/5) = -3/5log(3/ 5) - 2 /5log(2/ 5) = 0.971bits

Wartość dla atrybutu:

info([3,2],[4,0],[3,2]) = (5/14)´0.971+ (4 /14)´0 + (5/14)´0.971= 0.693 bits

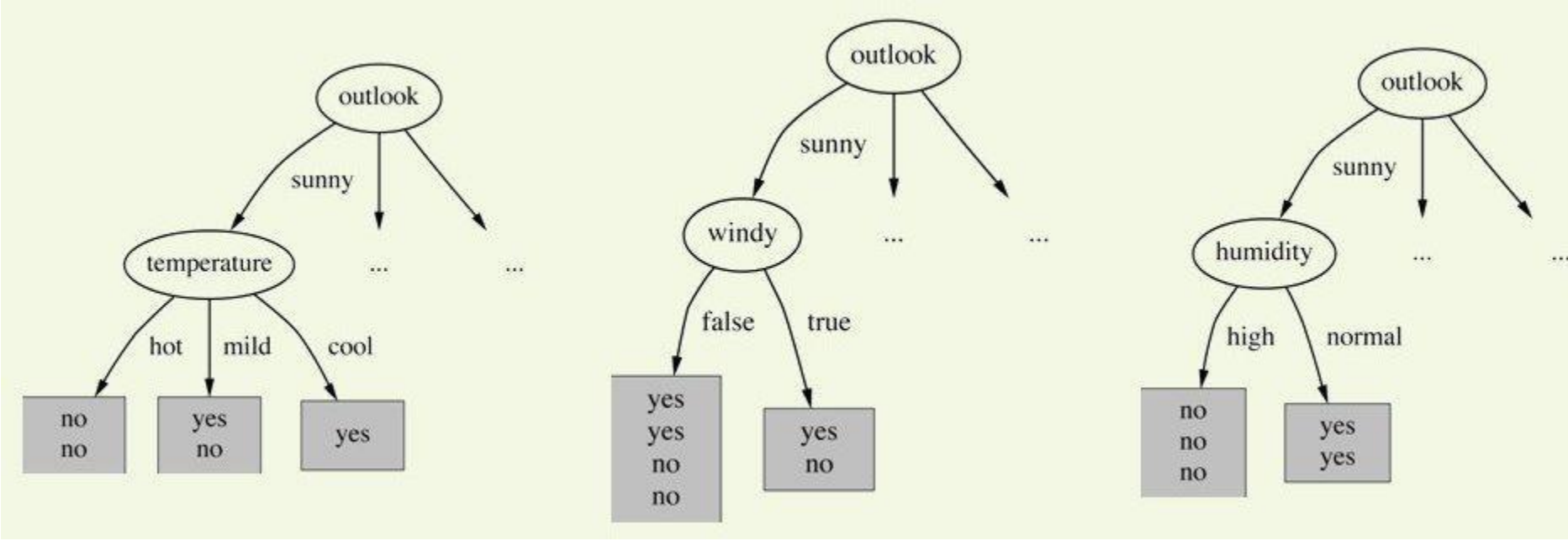


INFORMATION GAIN = informacja przed rozdzieleniem – informacja po Rozdzieleniu

- gain("Outlook") = info([9,5]) - info([2,3],[4,0],[3,2]) = 0.940 - 0.693 = 0.247 bits
- Przyrosty informacji dla poszczególnych atrybutów w danych testowych:
- gain("Outlook") = 0.247 bits
- gain("Temperature") = 0.029bits
- gain("Humidity") = 0.152 bits
- gain("Windy") = 0.048 bits

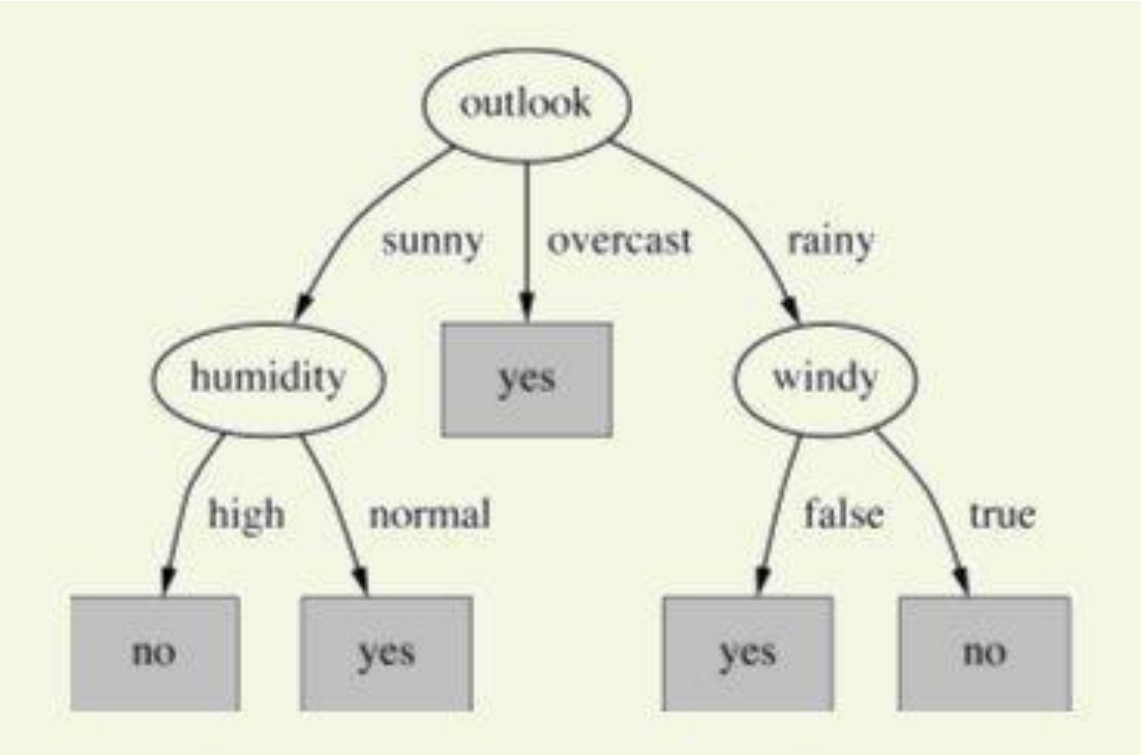
Przykład: Dalszy podział

- gain("Temperature") = 0.571bits
- gain("Humidity") = 0.971bits
- gain("Windy") = 0.020 bits



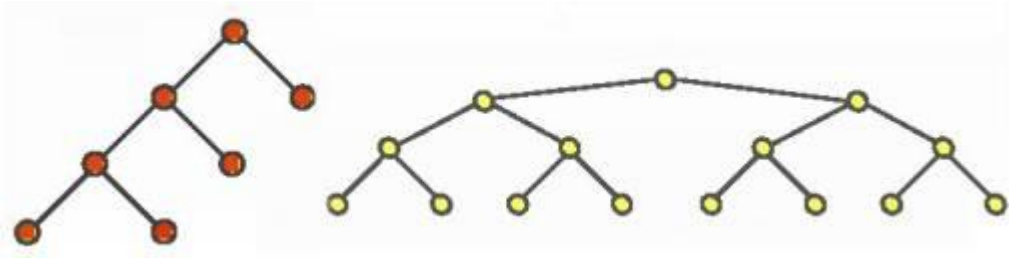
Przykład: Końcowe drzewo

- Uwaga: Nie wszystkie liście muszą być „czyste”. Czasami identyczne instancje mają różne klasy.
- Proces budowy drzewa zatrzymuje się gdy dalszy podział nie jest możliwy (lub spełnione jest inne kryterium stopu)



Rodzaje drzew

Drzewo binarne



Drzewo niebinarne, regresyjne

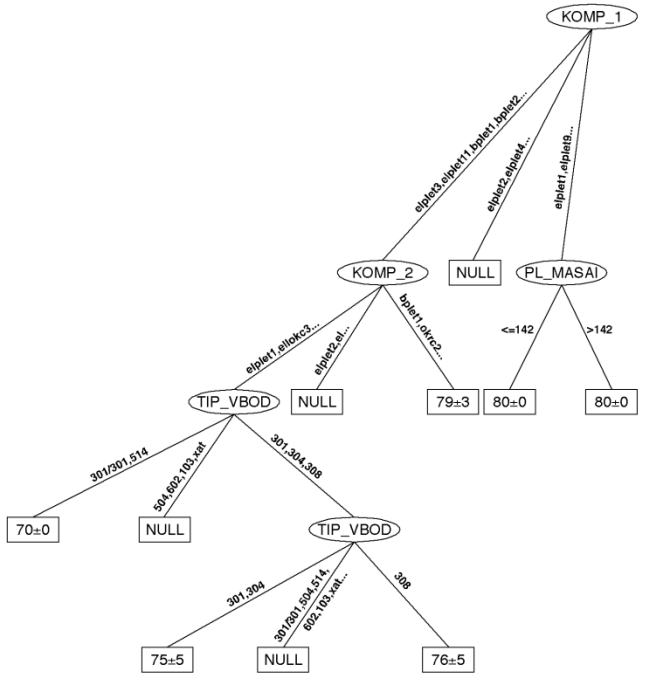
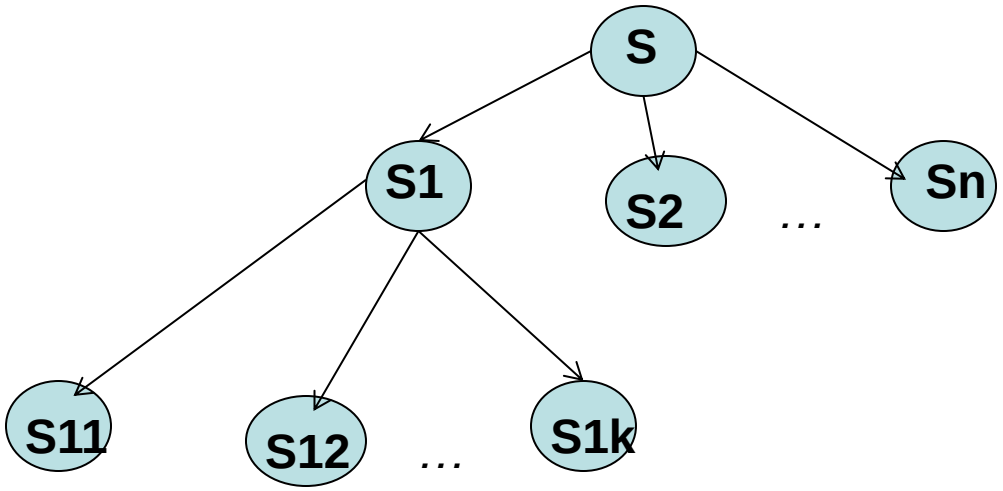


Figure 1.
A part of a regression tree

Budowanie drzewa

- *Mając zbiór obiektów S, sprawdź, czy należą one do tej samej klasy. Jeśli tak, to zakończ pracę.*
- *2. W przeciwnym przypadku rozważ wszystkie możliwe podziały zbioru S na podzbiory* $S_1, S_2, \ldots, S_n$ *tak, aby były one jak najbardziej jednorodne.*
- *3. Dokonaj oceny jakości każdego z tych podziałów zgodnie z przyjętym kryterium i wybierz najlepszy z nich.*
- *4. Podziel zbiór S w wybrany sposób.*
- *5. Wykonaj kroki1-4 rekurencyjnie dla każdego z podzbiorów.*



Algorytmy

- Id3 – Quinlan 1986
- C4.5 – na podstawie id3 – Quinlan 1993
- Cart – Breiman 1984
- Chaid – już lata 70-te „Chi-squared-Automatic-Interaction-Detection” – (oryginalnie tylko dla danych nominalnych)
- Hunt’s Algorithm (jeden z pierwszych)
- SLIQ,SPRINT

ID 3 (Quinlan, 1986)

- Bardzo prosty algorytm
- Opiera się na kryterium „information gain”
- Bada czy wszystkie obiekty w danym poddrzewie mają tę samą wartość analizowanej cechy
- ID3 nie zakłada prunningu drzew
- ID3 nie dopuszcza też danych numerycznych ani wartości brakujących

Algorytm ID 3 Quinlana

Cechy algorytmu:

- wybór atrybutów, dla których kolejno przeprowadzane są testy, aby końcowe drzewo było jak najprostsze i jak najefektywniejsze. Wybór opiera się na liczeniu entropii, która decyduje, który z atrybutów da największy przyrost informacji. Czyli ten, który podzieli zbiór przykładów na jak najbardziej równe podzbiory.
- Poważną wadą jest wymóg dyskretności wszystkich cech przestrzeni klasyfikacji. Ta sama metoda ID3 może dawać bardzo różne rezultaty dla różnych metod dyskretyzacji. Kryterium oceniającym podziały jest kryterium przyrostu czystości. Miarą niejednorodności węzła jest miara entropii. Przyrost czystości jest nazywany wówczas przyrostem informacyjnym.
- Metoda ta polega na rekurencyjnym dzieleniu węzłów na podwęzły, aż do uzyskania maksymalnego drzewa. W każdym kroku metoda dzieli dany węzeł na tyle podwęzłów ile wartości ma najbardziej informatywna cecha (cechą oferująca maksymalną redukcję entropii). Niekorzystną konsekwencją takiej strategii jest tendencja do częstszego wykorzystywania cech, które mają dużą (w stosunku do innych) liczbę możliwych wartości.

Algorytm ID 3 Quinlana

Zalety algorytmu:

- prostota algorytmu,
- jeżeli w zbiorze treningowym nie ma zjawiska hałasu, tzn. nie ma rekordów, które dla tych samych wartości atrybutów mają przypisaną różną kategorię, wtedy ID3 daje poprawny wynik dla wszystkich rekordów ze zbioru treningowego.

Wady algorytmu:

- algorytm nie radzi sobie z ciągłymi dziedzinami atrybutów (zakłada, że wartości atrybutów są dyskretne),
- zakłada, że wszystkie rekordy w zbiorze treningowym są wypełnione. Algorytm nie zadziała, jeżeli choć jeden rekord zawiera niepełne dane.
- duży rozmiar drzewa,
- brak odporności na zjawisko overfitting - algorytm nie radzi sobie z danymi zaburzającymi ogólną ich informację może to prowadzić do wysokiego współczynnika błędów na danych testowych.

C4.5 (Quinlan, 1993)

- Rozwinięcie ID3
- Stosuje kryterium „gain ratio”
- Podział drzewa kończy się gdy liczba obiektów do podziału jest już mniejsza niż pewna wartości progowa
- Przycinanie drzew jest możliwe już po zbudowaniu drzewa
- Dopuszcza się wartości numeryczne

Algorytm C4.5

Podział danych na podwęzły wprowadza wówczas wagi dla wektorów treningowych, które dla wektora z brakującą wartością atrybutu decyzyjnego odpowiadają rozkładowi pozostałych danych w podwęzłach. Stosownej modyfikacji podlegają wówczas współczynniki Pi ze wzoru a – zamiast mocy zbiorów liczy się sumy wag elementów tych zbiorów. Współczynniki pi są uwzględniane również przy podejmowaniu decyzji na podstawie drzewa, by wyliczyć prawdopodobieństwa wpadania do poszczególnych węzłów oraz przynależenia do poszczególnych klas. System C4.5 oprócz metody indukcji drzewa decyzji oferuje klasyfikator będący zbiorem reguł logiki klasycznej. Reguły traktowane są tutaj jako różny od drzewa model klasyfikacji, ponieważ nie są one wierną reprezentacją drzewa. Reguły wiernie opisujące drzewo są poddawane procesowi oczyszczania: w każdej regule usuwane są przesłanki, których pominięcie nie powoduje spadku jakości klasyfikacji zbioru treningowego. Wykonywane jest dla każdej reguły z osobna, można w ten sposób otrzymać klasyfikator istotnie różny od drzewa (zwykle dający w testach niższe wartości poprawności).

CART (Classification and Regression Trees) – (Breiman, 1984)

- Buduje drzewa binarne
- Kryterium podziału jest tzw. „twoing criteria”
- Drzewo można przycinać stosując kryterium kosztu i złożoności przycinania
- Buduje drzewa regresyjne (w liściach są wartości rzeczywiste a nie klasy)
- Algorytm stara się tak podzielić obiekty by minimalizować błąd predykcji (metoda najmniejszych kwadratów)
- Predykcja w każdym liściu drzewa opiera się na ważonej średniej dla węzła

CHAID – oparty na metodzie Chi^2

- Przewidziany dla wartości nominalnych .
- Przewiduje wartości brakujące.
- Nie przewiduje przycinania drzewa.
- Nie buduje drzew binarnych – w budowanym drzewie dany węzeł może mieć więcej niż 2 potomków.
- Jeśli mamy dane nominalne – to mamy od razu podział na kategorie.
- Jeśli mamy dane ilościowe – musimy podzielić obserwacje na kategorie tak by mniej więcej każda kategoria miała tyle samo obserwacji.
- Wybiera się z wszystkich cech (pogrupowanych w kategorie) takie pary, które się od siebie najmniej różnią - oblicza się test F.
- Jeśli test nie wykaże istotnej różnicy taką parę cech łączy się w jedną kategorię – i ponownie szuka się kolejnej pary.
- Potem wybiera się taką cechę, która daje najbardziej istotny podział i robi się to tak długo aż kolejne wybierane cechy dają wartość niższą niż zadana wartość p.
- Proces jest powtarzany tak długo, póki możliwe są kolejne podziały.

Cechy algorytmu	**Algorytm ID3**	**Algorytm C4.5**
Wartości atrybutów	tylko dyskretne – dla wartości ciągłych niezbędny proces dyskretyzacji wartości atrybutów	dyskretne jak i ciągłe dla atrybutów ciągłych algorytm rozpatruje wszystkie możliwe podziały na dwa podzbiory zdeterminowane punktem podziału, atrybuty ciągłe mogą pojawiać się na wielu poziomach tej samej gałęzi drzewa, dla każdego z możliwych podziałów ocenia się jego jakość mierząc wartość względnego zysku informacyjnego i wybiera ten, który maksymalizuje zysk.
Odporność na brak wartości atrybutów	brak – algorytm nie zadziała jeśli w zestawie danych wejściowych brakuje chodź jednej wartości atrybutu	jest – przyrost informacji nie bierze pod uwagę danych, dla których brakuje wartości atrybutu, przyrost informacji skaluje się mnożąc go przez częstość występowania wartości tej cechy w próbie treningowej
Odporność na nadmierne dopasowanie ***(ang. overfitting)***	brak – z reguły algorytm tworzy duże drzewa	jest – zastosowanie przycinania bardzo skutecznie zapobiega nadmiernemu rozrostowi drzewa – dane poddrzewo, które ma zostać przycięte zastępowane jest przez tą wartość atrybutu, która w przycinanym poddrzewie występuje najczęściej

Tworzenie drzewa decyzyjnego

Drzewo decyzyjne powstaje na podstawie rekurencyjnego podziału zbioru uczącego na podzbiory, do momentu uzyskania ich jednorodności ze względu na przynależność obiektów do klas. Postuluje się jak najmniejszą liczbę węzłów w powstałym dendrogramie, co powinno zapewnić prostotę powstałych reguł klasyfikacji.

Ogólny schemat tworzenia drzewa decyzyjnego na podstawie zbioru uczącego można przedstawić w pięciu punktach:

- 1. Badając zbiór obiektów *S należy sprawdzić, czy należą one do tej samej klasy. Jeżeli tak, można zakończyć pracę.*
- 2. Jeżeli analizowane obserwacje nie należą do jednej klasy, należy zbadać wszystkie możliwe podziały zbioru *S na podzbiory jak najbardziej jednorodne.*
- 3. Następnie należy ocenić jakość każdego podzbioru według przyjętego kryterium i wybrać najlepszy z nich.
- 4. W następnym kroku należy podzielić zbiór początkowy w wybrany sposób.
- 5. Wyżej opisane algorytm należy zastosować dla każdego z podzbiorów.

Uczymy się budować drzewa decyzyjne algorytmem ID 3

- Rozważmy binarną klasyfikację C. Jeśli mamy zbiór przykładów S, dla którego potrafimy stwierdzić, które przykłady ze zbioru S dają wartość pozytywną dla C (p_+) a które wartość negatywną dla C (p_-) wówczas entropię (ang. **Entropy)** dla S obliczymy jako:

$$Entropy(S) = -p_+ log_2(p_+) - p_- log_2(p_-)$$

- Zatem mając daną klasyfikację na większą liczbę klas (C dzieli się na kategorie c_1, ..., c_n) i wiedząc które przykłady ze zbioru S mają klasę c_i obliczymy odpowiednią proporcję dla p_i, a wtedy entropię dla S policzymy jako:

$$Entropy(S) = \sum_{i=1}^{n} -p_i log_2(p_i)$$

- Interpretacja wartości: $-p*log_2(p)$ jest taka, że kiedy *p* osiąga wartość bliską zeru (0) (tak się dzieje m.in. wtedy, gdy kategoria ma tylko kilka przykładów (jest mało liczna), wtedy wartość *log(p)* osiąga dużą wartość ujemną, ale po wymnożeniu przez wartość *p* ostatecznie cała entropia osiąga wartość bliską 0.
- Podobnie, jeśli *p* osiąga wartość bliską 1 (gdy kategoria rozważana pochłania większość przykładów ze zbioru), wtedy wartość *log(p)* jest bliska 0, a wtedy całościowa entropia także osiąga wartość 0.
- To ma na celu pokazać, że entropia osiąga wartość bliską 0 albo wtedy gdy dla danej kategorii jest bardzo mało przykładów, albo wręcz dana kategoria pochłania większość przykładów zadanych w analizowanym zbiorze.

Information Gain

- Na każdym etapie budowy drzewa trzeba wybrać atrybut który będzie tworzył dany węzeł w drzewie. Na początku powstaje problem, który atrybut powinien budować korzeń drzewa.
- W tym celu oblicza się zysk informacyjny jaki da dany atrybut jeśli to jego użyjemy do budowy węzła w drzewie:
- Jeśli więc mamy atrybut A z jego możliwymi wartościami, dla każdej możliwej wartości *v*, obliczymy entropię dla tej wartości danego atrybutu (S_v):

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Information Gain – przyrost informacji

Wybór najlepszego atrybutu do węzła

Information gain – miara zmiany entropii – wybieramy atrybut, dla którego ta miara jest największa

- Entropia zbioru uczącego S na początku:

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

- Wybieramy atrybut A, który dzieli zbiór S na podzbiory $S_1,\ldots,S_v$
  Podzbiór S_j zawiera m_j elementów. Wtedy entropia względna wynosi:

$$E(A) = \sum_{j=1}^{v} \frac{m_j}{n} E(S_j)$$

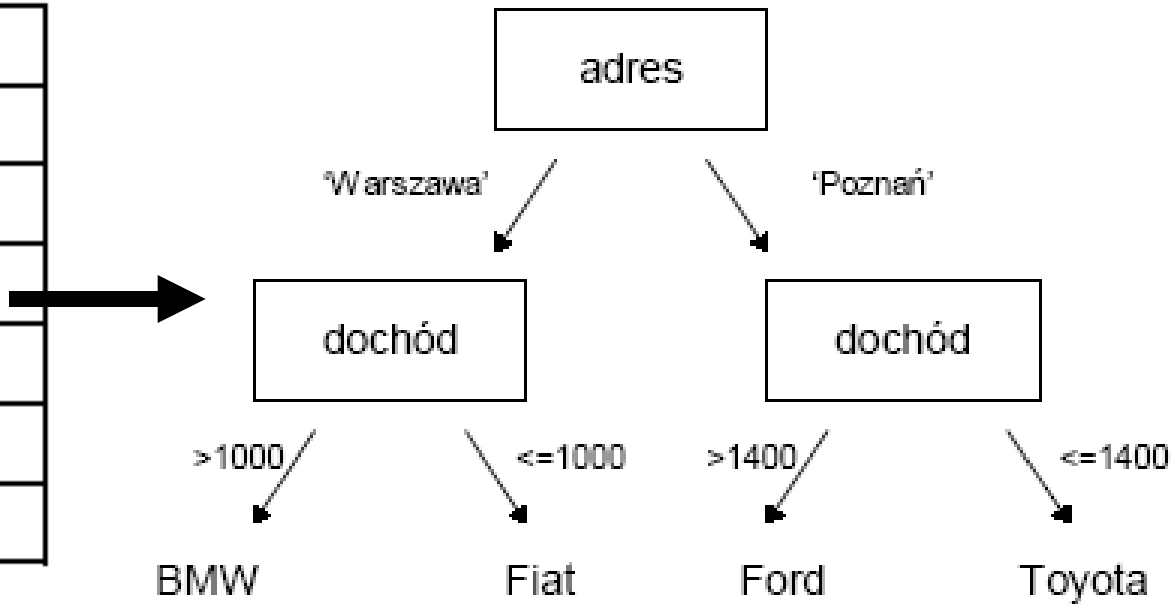
- Wybieramy atrybut, dla którego *information gain* jest największy

$$gain(A) = E(S) - E(A)$$

Drzewa decyzyjne

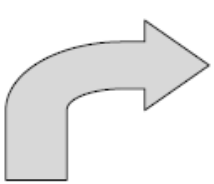
Najpowszechniejsza forma reprezentowania wiedzy przez dostępne oprogramowanie. Węzły są opisywane przez atrybuty eksplorowanej relacji, krawędzie opisują możliwe wartości dla atrybutu. Klasyfikacja odbywa się poprzez przeglądanie drzewa od korzenia do liści przez krawędzie opisane wartościami atrybutów.

Adres	Dochód	Samochód
Warszawa	4000	BMW
Poznań	2900	Ford
Poznań	1400	Toyota
Warszawa	1000	Fiat
Poznań	1600	Ford
Poznań	3500	Ford



Drzewa decyzyjne

Przykład: automatyczny podział kierowców na powodujących i nie powodujących wypadki drogowe Dwie klasy: TAK i NIE



Zbiór uczący

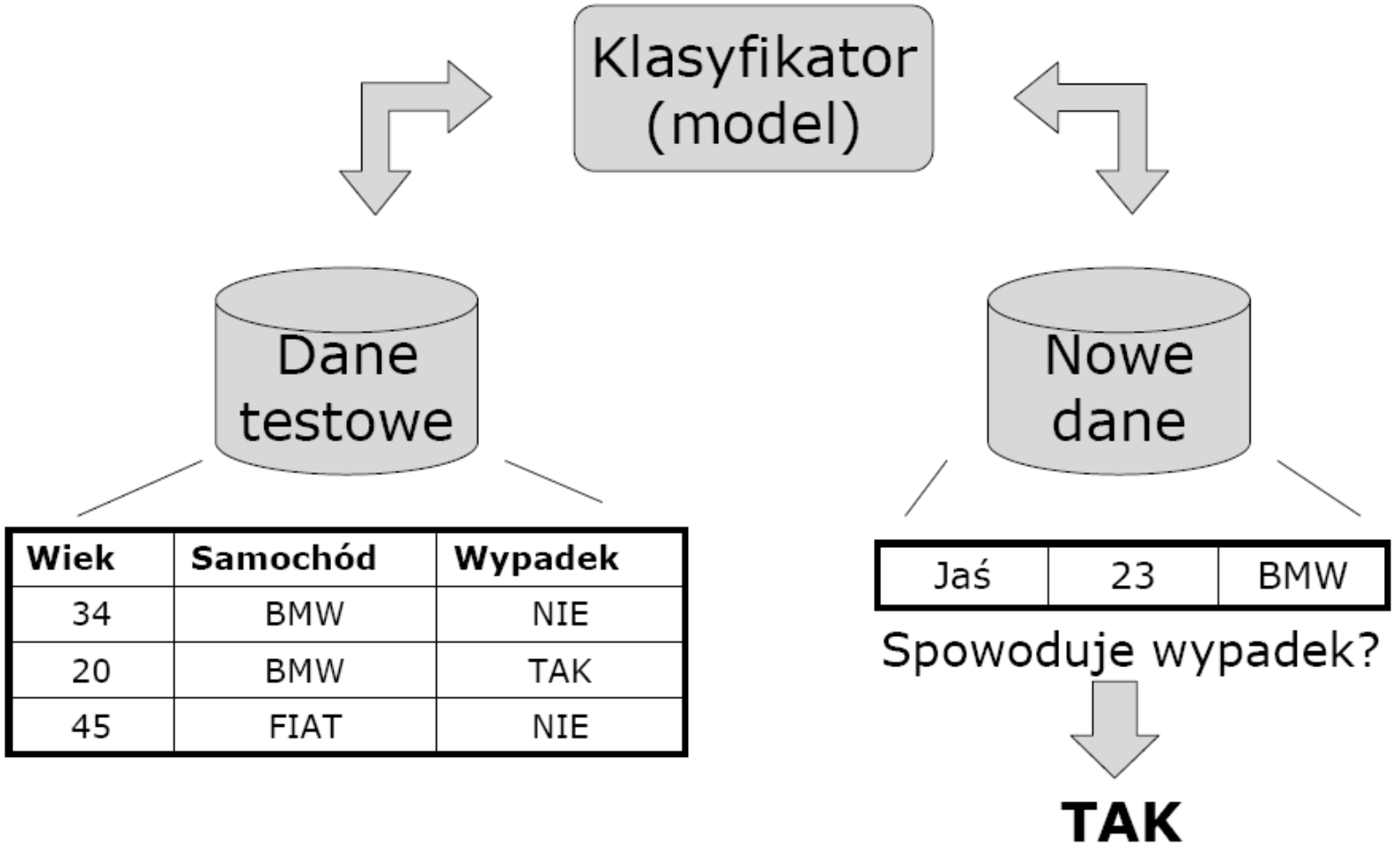
Wiek	Samochód	Wypadek
24	BMW	TAK
22	BMW	TAK
45	FIAT	NIE
20	FIAT	NIE
50	BMW	NIE

Algorytm uczenia się

Klasyfikator (model)

IF marka = BMW
AND wiek_kier < 25
THEN wypadek = TAK

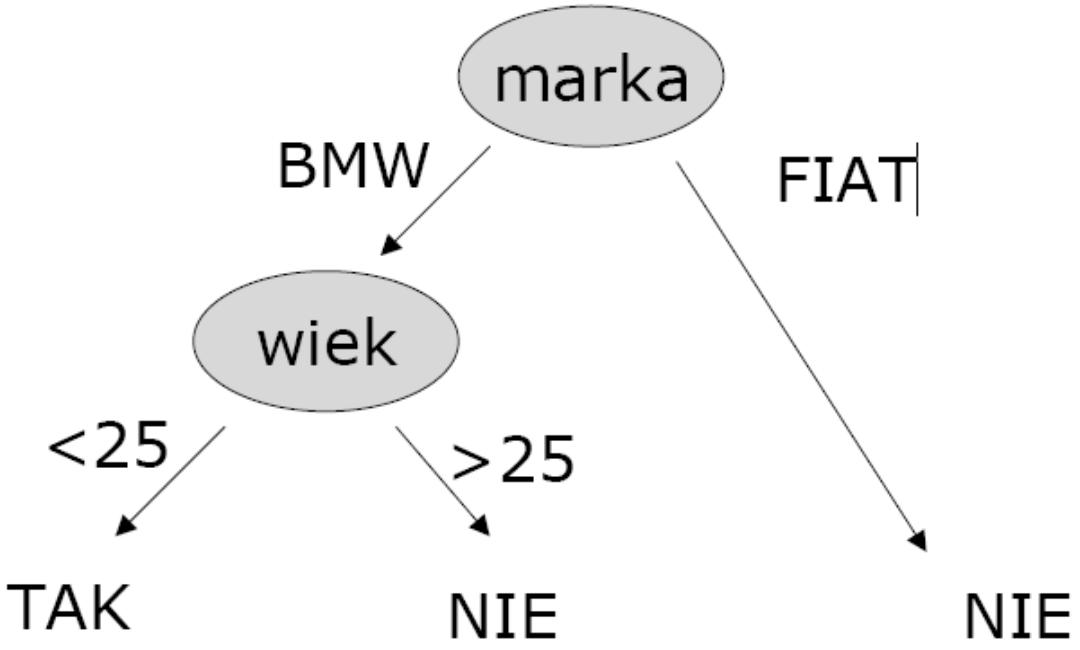
Drzewa decyzyjne




```
IF marka = BMW AND wiek_kier < 25 THEN wypadek = TAK
```

Drzewa decyzyjne

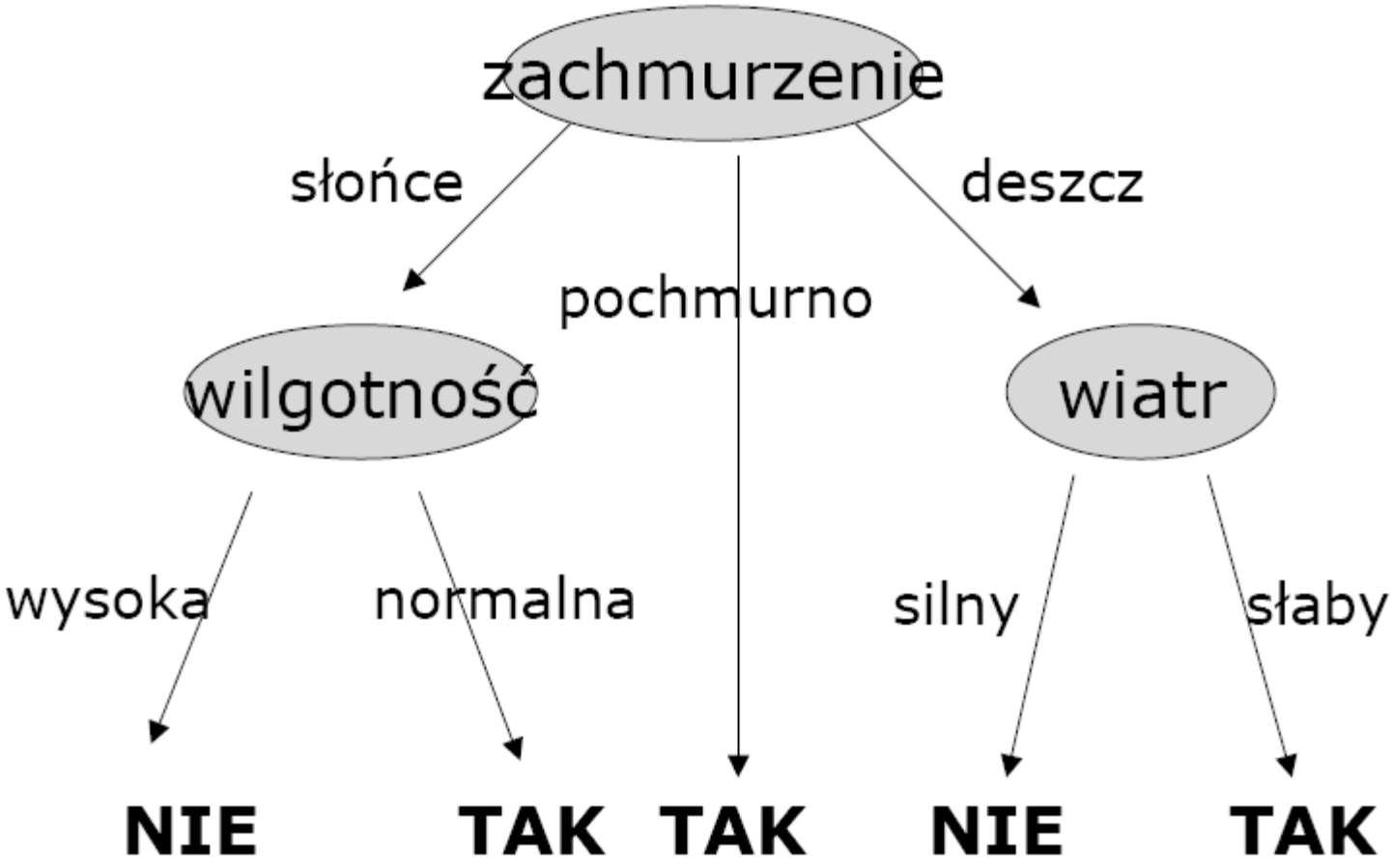
- Drzewo decyzyjne jest formą opisu wiedzy klasyfikującej
- Węzłom drzewa odpowiadają atrybuty eksplorowanej relacji
- Krawędzie opisują wartości atrybutów
- Liśćmi drzewa są wartości atrybutu klasyfikacyjnego



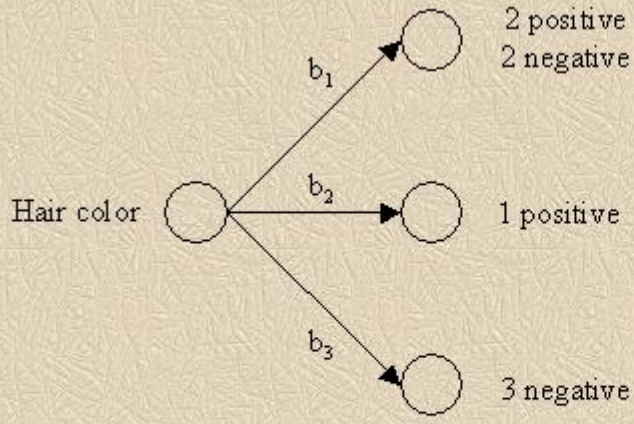
Drzewa decyzyjne - przykład

zachmurzenie	temperatura	wilgotność	wiatr	decyzja
słońce	gorąco	wysoka	słaby	nie
słońce	gorąco	wysoka	silny	nie
pochmurno	gorąco	wysoka	słaby	tak
deszcz	średnio	wysoka	słaby	tak
deszcz	chłodno	normalna	słaby	tak
deszcz	chłodno	normalna	silny	nie
pochmurno	chłodno	normalna	silny	tak
słońce	średnio	wysoka	słaby	nie
słońce	chłodno	normalna	słaby	tak
deszcz	średnio	normalna	słaby	tak
słońce	średnio	normalna	silny	tak
pochmurno	średnio	wysoka	silny	tak
pochmurno	gorąco	normalna	słaby	tak
deszcz	średnio	wysoka	silny	nie

Drzewa decyzyjne dla przykładu



Name	Hair	Height	Weight	Lotion	Result
Sarah	blonde	average	light	no	sunburned (positive)
Dana	blonde	tall	average	yes	none (negative)
Alex	brown	short	average	yes	none
Annie	blonde	short	average	no	sunburned
Emily	red	average	heavy	no	sunburned
Pete	brown	tall	heavy	no	none
John	brown	average	heavy	no	none
Katie	blonde	short	light	yes	none



$$= \sum_b \left(\left(\frac{n_b}{n_t}\right) \times \left[\sum_c - \left(\frac{n_{bc}}{n_b}\right) \log_2 \left(\frac{n_{bc}}{n_b}\right) \right] \right)$$

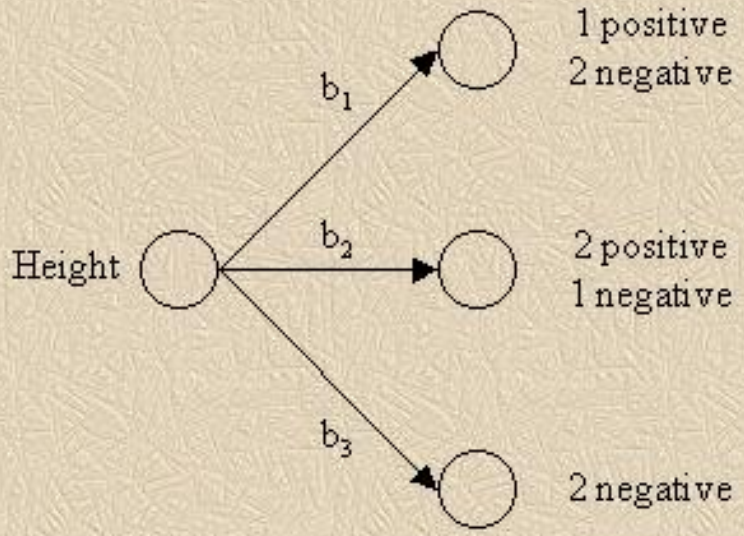
$$= \frac{4}{8} \times \left[-\frac{2}{4} \log_2 \left(\frac{2}{4}\right) - \frac{2}{4} \log_2 \left(\frac{2}{4}\right) \right] + \frac{1}{8} \times (-\log_2 1) + \frac{3}{8} \times (-\log_2 1)$$

$$= \frac{4}{8} \times \left[-\frac{2}{4} \log_2 \left(\frac{2}{4}\right) - \frac{2}{4} \log_2 \left(\frac{2}{4}\right) \right]$$

$$= 0.50$$

b_1 = blonde
b_2 = red
b_3 = brown

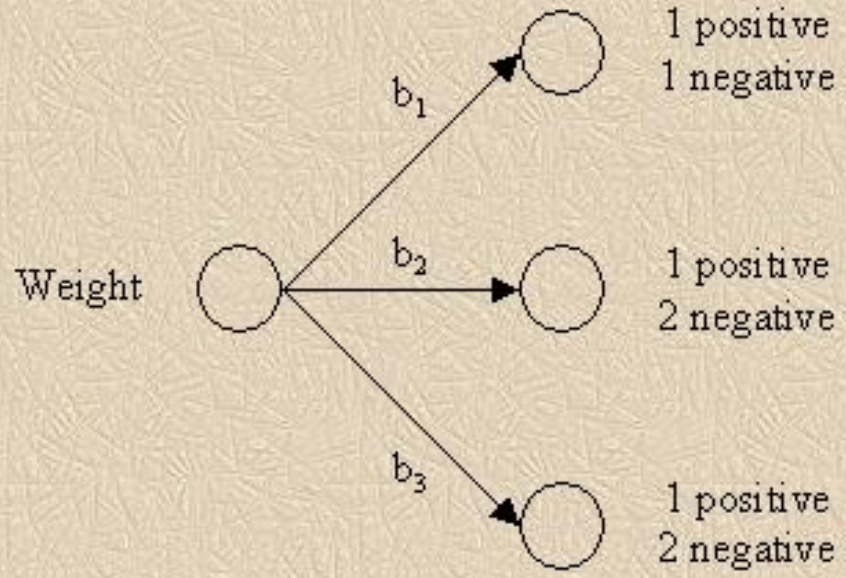
Average Entropy = 0.50



b_1 = short
b_2 = average
b_3 = tall

$$= \frac{3}{8} * (-\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3}))$$
$$+ \frac{3}{8}(-\frac{2}{3}\log_2(\frac{2}{3}) - \frac{1}{3}\log_2(\frac{1}{3})) + \frac{2}{8}(0)$$
$$= -\frac{1}{8}\log_2(\frac{1}{3}) - \frac{1}{4}\log_2(\frac{2}{3}) - \frac{1}{4}\log_2(\frac{2}{3}) - \frac{1}{8}\log_2(\frac{1}{3})$$
$$= -\frac{1}{4}\log_2(\frac{1}{3}) - \frac{1}{2}(\log_2(\frac{2}{3})$$
$$= -\frac{1}{4}\left(\frac{\log_{10}(\frac{1}{3})}{\log_{10}(2)}\right) - \frac{1}{2}\left(\frac{\log_{10}(\frac{2}{3})}{\log_{10}(2)}\right)$$
$$= 0.3962 + 0.2925$$
$$= 0.69$$

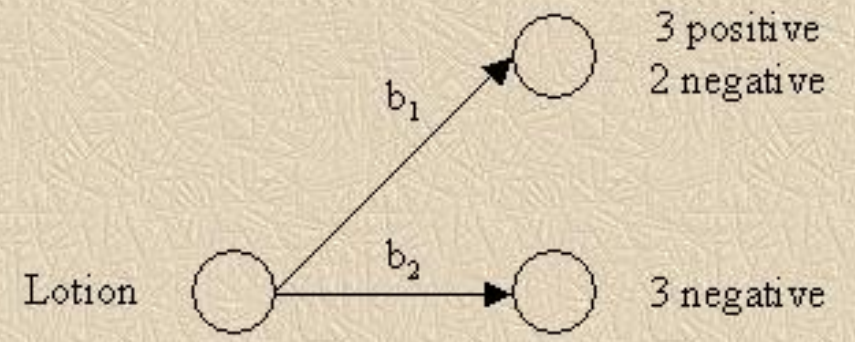
Average Entropy = 0.69



b_1 = light
b_2 = average
b_3 = heavy

$$= \frac{2}{8}(-\frac{1}{2}\log_2(\frac{1}{2}) - \frac{1}{2}\log_2(\frac{1}{2})) +$$
$$\frac{3}{8}(-\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3})) +$$
$$\frac{3}{8}(-\frac{1}{3}\log_2(\frac{1}{3}) - \frac{2}{3}\log_2(\frac{2}{3})$$
$$= -\frac{1}{8}\log_2(\frac{1}{4}) - \frac{1}{8}\log_2(\frac{1}{3}) - \frac{1}{4}\log_2(\frac{2}{3}) - \frac{1}{8}\log_2(\frac{1}{3}) - \frac{1}{4}\log_2(\frac{2}{3})$$
$$= -\frac{1}{8}\log_2(\frac{1}{4}) - \frac{1}{4}\log_2(\frac{1}{3}) - \frac{1}{2}\log_2(\frac{2}{3})$$
$$= -\frac{1}{8}\left(\frac{\log_{10}(\frac{1}{4})}{\log_{10} 2}\right) - \frac{1}{4}\left(\frac{\log_{10}(\frac{1}{3})}{\log_{10} 2}\right) - \frac{1}{2}\left(\frac{\log_{10}(\frac{2}{3})}{\log_{10} 2}\right)$$
$$= 0.25 + 0.3962 + 0.2925$$
$$= 0.94$$

Average Entropy = 0.94



$b_1 = \text{no}$

$b_2 = \text{yes}$

Average Entropy = 0.61

Sample average entropy calculation for the attribute" Lotion"

$$= \frac{5}{8}(-\frac{3}{5}\log_2(\frac{3}{5}) - \frac{2}{5}\log_2(\frac{2}{5})) + \frac{3}{8}(0)$$

$$= -\frac{3}{8}\log_2(\frac{3}{5}) - \frac{1}{4}\log_2(\frac{2}{5})$$

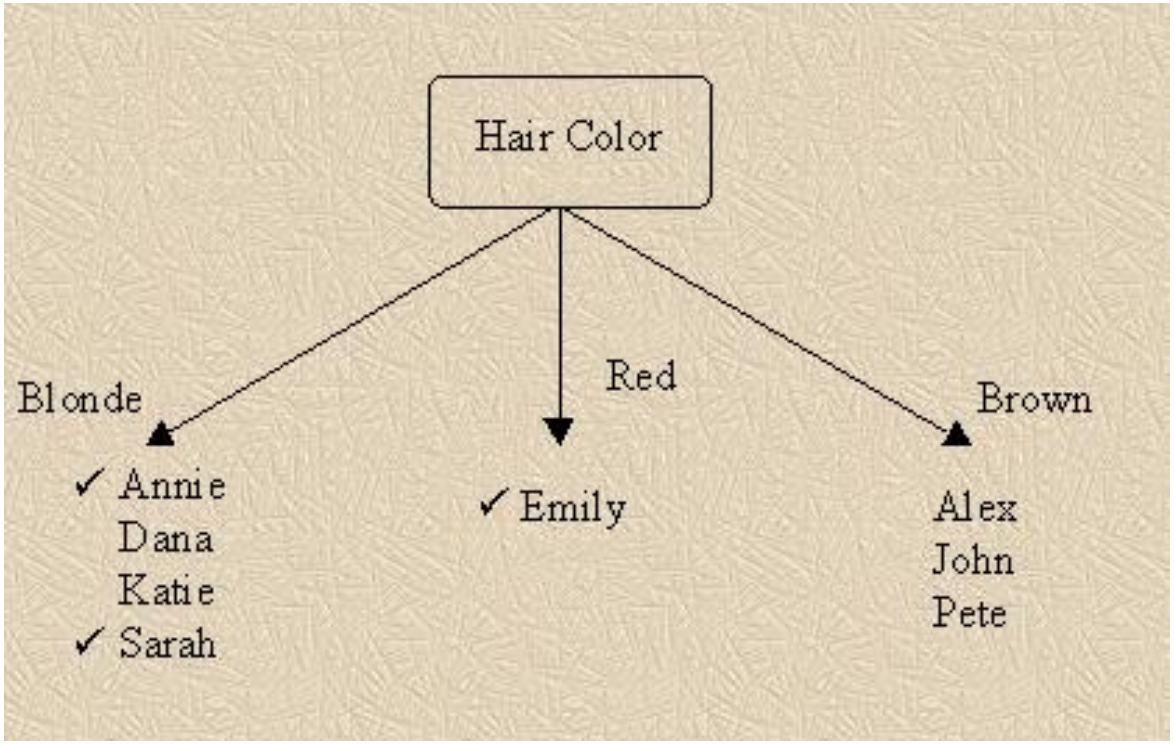
$$= -\frac{3}{8}\left(\frac{\log_{10}(\frac{3}{5})}{\log_{10} 2}\right) - \frac{1}{4}\left(\frac{\log_{10}(\frac{2}{5})}{\log_{10} 2}\right)$$

$$= 0.2764 + 0.3305$$

$$= 0.61$$

Wyniki obliczeń….i budujemy drzewo

Attribute	Average Entropy
Hair Color	0.50
Height	0.69
Weight	0.94
Lotion	0.61



Ocena jakości drzewa

Jakość drzewa ocenia się:

- rozmiarem: im drzewo jest mniejsze, tym lepsze
- małą liczbą węzłów, małą wysokością, lub małą liczbą liści;
- dokładnością klasyfikacji na zbiorze treningowym
- dokładnością klasyfikacji na zbiorze testowym

Przykład

obiekt	pogoda	temperatura	wilgotność	wiatr	grać(x)
x_1	słonecznie	gorąco	wysoka	słaby	nie
x_2	słonecznie	gorąco	wysoka	silny	nie
x_3	pochmurno	gorąco	wysoka	słaby	tak
x_4	deszczowo	łagodnie	wysoka	słaby	tak
x_5	deszczowo	zimno	normalna	słaby	tak
x_6	deszczowo	zimno	normalna	silny	nie
x_7	pochmurno	zimno	normalna	silny	tak
x_8	słonecznie	łagodnie	wysoka	słaby	nie
x_9	słonecznie	zimno	normalna	słaby	tak
x_{10}	deszczowo	łagodnie	normalna	słaby	tak
x_{11}	słonecznie	łagodnie	normalna	silny	tak
x_{12}	pochmurno	łagodnie	wysoka	silny	tak
x_{13}	pochmurno	gorąco	normalna	słaby	tak
x_{14}	deszczowo	łagodnie	wysoka	silny	nie

Drzewa decyzyjne - przykład

zachmurzenie	temperatura	wilgotność	wiatr	decyzja
słońce	gorąco	wysoka	słaby	nie
słońce	gorąco	wysoka	silny	nie
pochmurno	gorąco	wysoka	słaby	tak
deszcz	średnio	wysoka	słaby	tak
deszcz	chłodno	normalna	słaby	tak
deszcz	chłodno	normalna	silny	nie
pochmurno	chłodno	normalna	silny	tak
słońce	średnio	wysoka	słaby	nie
słońce	chłodno	normalna	słaby	tak
deszcz	średnio	normalna	słaby	tak
słońce	średnio	normalna	silny	tak
pochmurno	średnio	wysoka	silny	tak
pochmurno	gorąco	normalna	słaby	tak
deszcz	średnio	wysoka	silny	nie

W tabeli znajdują się cztery atrybuty warunkowe (pogoda, temperatura, wilgotność wiatr) oraz atrybut decyzyjny grać. Na początku należy wyznaczyć liczność każdego zbiorów trenujących:

$|T_{tak}| = |\{3, 4, 5, 7, 9, 10, 11, 12, 13\}| = 9,$
oraz
$|T_{nie}| = |\{1, 2, 6, 8, 14\}| = 5.$

$|T(\text{pogoda, słonecznie})| = |\{1, 2, 8, 9, 11\}| = 5$
$|T_{tak(}\text{pogoda, słonecznie})| = |\{9, 11\}| = 2$
$|T_{nie}(\text{pogoda, słonecznie})| = |\{1, 2, 8\}| = 3$

$|T(\text{pogoda, pochmurno})| = |\{3, 7, 12, 13\}| = 4$
$|T_{tak}(\text{pogoda, pochmurno})| = |\{3, 7, 12, 13\}| = 4$
$|T_{nie}(\text{pogoda, pochmurno})| = |\{\}| = 0$

$|T(\text{pogoda, deszczowo})| = |\{4, 5, 6, 10, 14\}| = 5$
$|T_{tak}(\text{pogoda, deszczowo})| = |\{4, 5, 10\}| = 3$
$|T_{nie}(\text{pogoda, deszczowo})| = |\{6, 14\}| = 2$

Po obliczeniu liczności zbiorów trenujących należy obliczyć entropię.

Korzystając ze wzoru , $E_{tr}(P) = \sum_{d \in C} -\frac{|P_{tr}^d|}{P_{tr}} \log \frac{|P_{tr}^d|}{P_{tr}}$

przyjmując logarytmy dwójkowe kolejno obliczamy:

Ent(S) $= -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0{,}4097 + 0{,}5305 = 0{,}9402$

Ze wzoru $I(P) = \sum_{d \in C} -\frac{|P^d|}{P} \log \frac{|P^d|}{P}$ obliczamy informację zawartą w zbiorze

etykietowanych przykładów P I(P)

Następnie ze wzoru $E_t(P) = \sum_{r \in R_t} \frac{|P_{tr}|}{P} E_{tr}(P)$ obliczam entropię ważoną:

$$= -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} \quad = 0{,}4097 + 0{,}5305 = 0{,}9402$$

Ent(S | Pogoda) $= \frac{5}{14}(-\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5}) + \frac{4}{14}(-\frac{4}{4}\log_2\frac{4}{4} - \frac{0}{4}\log_2\frac{0}{4}) +$

$\frac{5}{14}(-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}) = \quad \frac{5}{14} * 0.97 + \frac{5}{14} * 0.97 = 0.6928$

Ent (S|Temperatura) $= \frac{4}{14}(-\frac{2}{4}\log_2\frac{2}{4}-\frac{2}{4}\log_2\frac{2}{4}) + \frac{6}{14}(-\frac{4}{6}\log_2\frac{4}{6}-\frac{2}{6}\log_2\frac{2}{6}) +$

$\frac{4}{14}(-\frac{3}{4}\log_2\frac{3}{4}-\frac{1}{4}\log_2\frac{1}{4}) = \frac{4}{14}*1+\frac{6}{14}*0.91+\frac{4}{14}*0.81 = 0.9110$

Ent (S|Wilgotność) $= \frac{7}{14}(-\frac{3}{7}\log_2\frac{3}{7}-\frac{4}{7}\log_2\frac{4}{7}) + \frac{7}{14}(-\frac{6}{7}\log_2\frac{6}{7}-\frac{1}{7}\log_2\frac{1}{7}) =$

$\frac{7}{14}*0.98+\frac{7}{14}*0.59 = 0.7884$

Ent(S|Wiatr) $= \frac{8}{14}(-\frac{6}{8}\log_2\frac{6}{8}-\frac{2}{8}\log_2\frac{2}{8}) + \frac{6}{14}(-\frac{3}{6}\log_2\frac{3}{6}-\frac{3}{6}\log_2\frac{3}{6}) = \frac{8}{14}*0.81+\frac{6}{14}*1 = 0.8921$

Teraz możemy policzyć przyrost informacji dla każdego atrybutu, korzystając ze wzoru na przyrost informacji

$$g_t(P) = I(P) - E_t(P)$$

```
Gain Information (Pogoda) = 0.9402 - 0.6935 = 0.2474
Gain Information (Temperatura) = 0.9402 - 0.9110 = 0.0292
Gain Information (Wilgotność) = 0.9402 - 0.7884 = 0.1518
Gain Information (Wiatr) = 0.9402 - 0.8921 =0.0481
```

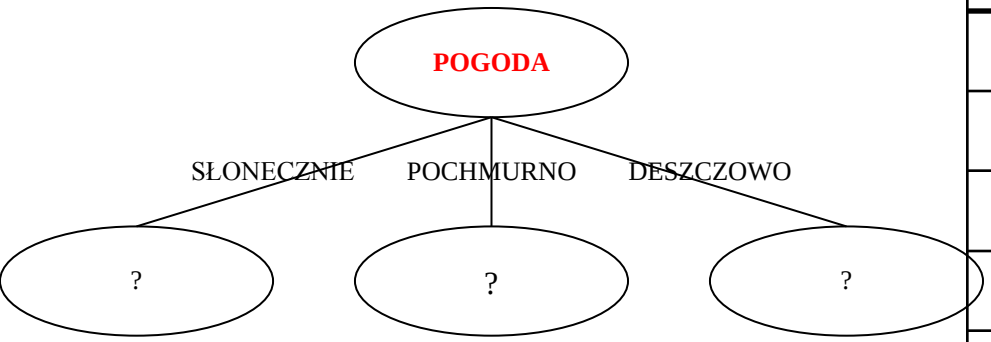
Z powyższych obliczeń wynika, że największy przyrost informacji da atrybut **pogoda** Rysujemy zatem drzewo decyzyjne

obiekt	pogoda	temperatura	wilgotność	wiatr	grać(x)
x_1	słonecznie	gorąco	wysoka	słaby	nie
x_2	słonecznie	gorąco	wysoka	silny	nie
x_3	pochmurno	gorąco	wysoka	słaby	tak
x_4	deszczowo	łagodnie	wysoka	słaby	tak
x_5	deszczowo	zimno	normalna	słaby	tak
x_6	deszczowo	zimno	normalna	silny	nie
x_7	pochmurno	zimno	normalna	silny	tak
x_8	słonecznie	łagodnie	wysoka	słaby	nie
x_9	słonecznie	zimno	normalna	słaby	tak
x_{10}	deszczowo	łagodnie	normalna	słaby	tak
x_{11}	słonecznie	łagodnie	normalna	silny	tak
x_{12}	pochmurno	łagodnie	wysoka	silny	tak
x_{13}	pochmurno	gorąco	normalna	słaby	tak
x_{14}	deszczowo	łagodnie	wysoka	silny	nie

obiekt	pogoda	temperatura	wilgotność	wiatr	grać
x_4	deszczowo	łagodnie	wysoka	słaby	tak
x_5	deszczowo	zimno	normalna	słaby	tak
x_6	deszczowo	zimno	normalna	silny	nie
x_{10}	deszczowo	łagodnie	normalna	słaby	tak
x_{14}	deszczowo	łagodnie	wysoka	silny	nie

obiekt	pogoda	temperatura	wilgotność	wiatr	grać
x_1	słoneczne	gorąco	wysoka	słaby	nie
x_2	słoneczne	gorąco	wysoka	silny	nie
x_8	słoneczne	łagodnie	wysoka	słaby	nie
x_9	słoneczne	zimno	normalna	słaby	tak
x_{11}	słoneczne	łagodnie	normalna	silny	tak

obiekt	pogoda	temperatura	wilgotność	wiatr	grać(x)
x_3	pochmurno	gorąco	wysoka	słaby	tak
x_7	pochmurno	zimno	normalna	silny	tak
x_{12}	pochmurno	łagodnie	wysoka	silny	tak
x_{13}	pochmurno	gorąco	normalna	słaby	tak



obiekt	pogoda	temperatura	wilgotność	wiatr	grać
x_1	słonecznie	gorąco	wysoka	słaby	nie
x_2	słonecznie	gorąco	wysoka	silny	nie
x_8	słonecznie	łagodnie	wysoka	słaby	nie
x_9	słonecznie	zimno	normalna	słaby	tak
x_{11}	słonecznie	łagodnie	normalna	silny	tak

Rozpoczynamy drugą iterację algorytmu: generujemy lewe poddrzewo wykonując kolejne obliczenia entropii oczywiście w ramach atrybutów z podzbioru $\text{Pogoda}_{\text{Słonecznie}}$

Ent(S | Wilgotność) = $\frac{3}{5}(-\frac{0}{3}\log_2\frac{0}{3}-\frac{3}{3}\log_2\frac{3}{3})-\frac{2}{5}(-\frac{2}{2}\log_2\frac{2}{2}-\frac{0}{2}\log_2\frac{0}{2})=$ 0

Ent(S | Wiatr) = $\frac{3}{5}(-\frac{1}{3}\log_2\frac{1}{3}-\frac{2}{3}\log_2\frac{2}{3})-\frac{2}{5}(-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2})=\frac{3}{5}*0.9182-\frac{2}{5}*1=0.15092$

Ent(S | Temperatura) $\frac{2}{5}(-\frac{0}{2}\log_2\frac{0}{2}-\frac{2}{2}\log_2\frac{2}{2})-\frac{2}{5}(-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2})-\frac{1}{5}(-\frac{1}{1}\log_2\frac{1}{1}-\frac{0}{1}\log_2\frac{0}{1})=\frac{2}{5}*1=\frac{2}{5}$

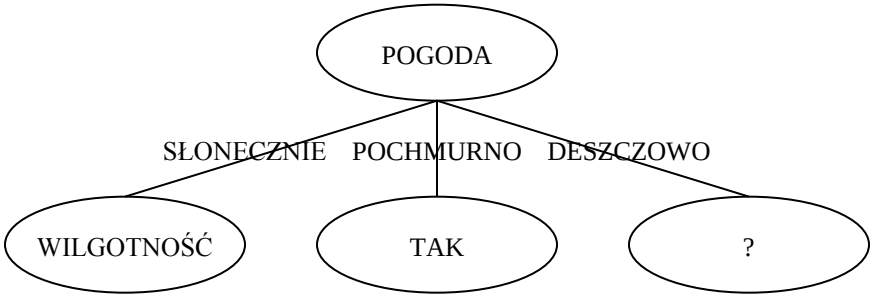
Następnie po obliczeniu entropii generujemy przyrost informacji:

Gain Information (PogodaSłonecznie + Wilgotność) = 0.97

Gain Information (PogodaSłonecznie + Wiatr) = $0.97-0.15092=0{,}81908$

Gain Information (PogodaSłonecznie + Temperatura) = $0.97-0-\frac{2}{5}-0=0.57$

Największy przyrost informacji spośród tych atrybutów da **wilgotność** dorysowujemy zatem kolejny węzeł drzewa decyzyjnego.



obiekt	pogoda	temperatura	wilgotność	wiatr	grać (x)
x_3	pochmurno	gorąco	wysoka	słaby	tak
x_7	pochmurno	zimno	normalna	silny	tak
x_{12}	pochmurno	łagodnie	wysoka	silny	tak
x_{13}	pochmurno	gorąco	normalna	słaby	tak

Dla środkowego poddrzewa nie ma potrzeby tworzenia kolejnych węzłów ponieważ wszystkie atrybuty tego podzbioru należą do tej samej klasy decyzyjnej. Możemy zatem wyznaczyć węzeł dla prawego poddrzewa:

$$Ent(S|PogodaDeszczowo+Wilgotność) = \frac{2}{5}(-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2}) - \frac{3}{5}(-\frac{2}{3}\log_2\frac{2}{3}-\frac{1}{3}\log_2\frac{1}{3}) = \frac{2}{5}*1-\frac{3}{5}*0.9182 = 0.15092$$

$$Ent(S|PogodaDeszczowo+Wiatr) = \frac{3}{5}(-\frac{3}{3}\log_2\frac{3}{3}-\frac{0}{3}\log_2\frac{0}{3}) - \frac{2}{5}(-\frac{0}{2}\log_2\frac{0}{2}-\frac{2}{2}\log_2\frac{2}{2}) = 0$$

$$Ent(S|PogodaDeszczowo+Temperatura) = \frac{3}{5}(-\frac{2}{3}\log_2\frac{2}{3}-\frac{1}{3}\log_2\frac{1}{3}) - \frac{2}{5}(-\frac{1}{2}\log_2\frac{1}{2}-\frac{1}{2}\log_2\frac{1}{2}) - 0 = \frac{3}{5}*0.9182-\frac{2}{5}*1-0 = 0.15092$$

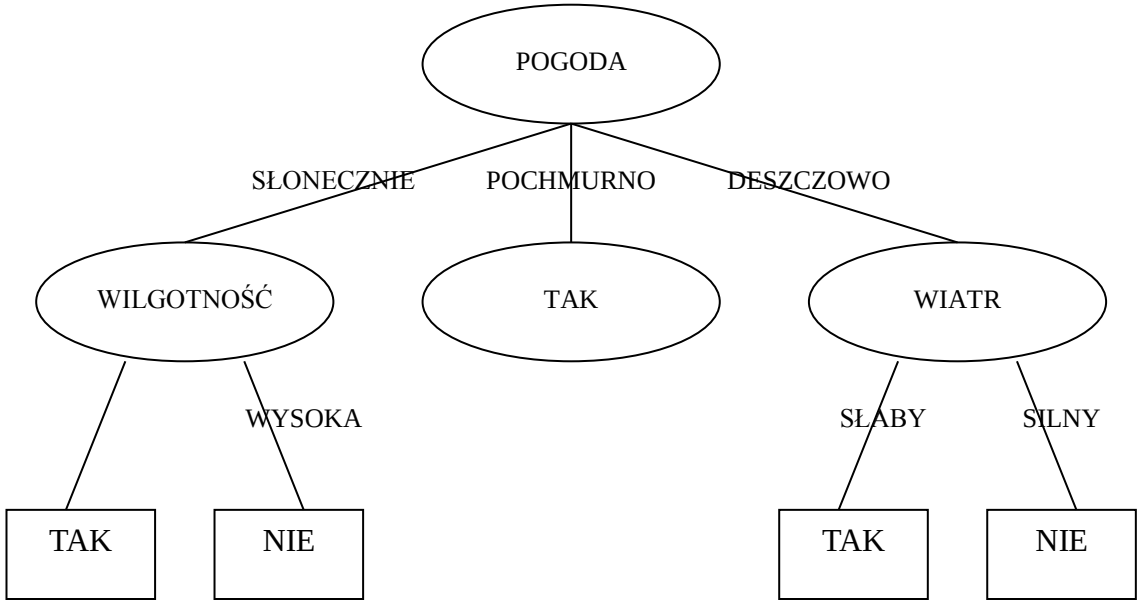
obiekt	pogoda	temperatura	wilgotność	wiatr	grać
x_4	deszczowo	łagodnie	wysoka	słaby	tak
x_5	deszczowo	zimno	normalna	słaby	tak
x_6	deszczowo	zimno	normalna	silny	nie
x_{10}	deszczowo	łagodnie	normalna	słaby	tak
x_{14}	deszczowo	łagodnie	wysoka	silny	nie

GainInformation(PogodaDeszczowo+Temperatura) $= 0.97 - \frac{3}{5} * 0.9182 - \frac{2}{5} * 1 - 0 = 0{,}01908$

GainInformation(PogodaDeszczowo+Wilgotność) $= 0.97 - \frac{2}{5} * 1 - \frac{3}{5} * 0.9182 = 0{,}01908$

GainInformation(PogodaDeszczowo+Wiatr) $= 0.97$

Możemy zatem dorysować kolejny węzeł drzewa:



Koniec

Każde drzewo decyzyjne jest reprezentacją pewnego zbioru reguł decyzyjnych, przy czym każdej ścieżce od korzenia do liścia odpowiada jedna reguła.

-Jeżeli Kłopoty z kręgosłupem=Tak, to Sport=Pływanie

-Jeżeli Kłopoty z kręgosłupem=Nie & Kontuzja kolana=Tak, to Sport=Pływanie

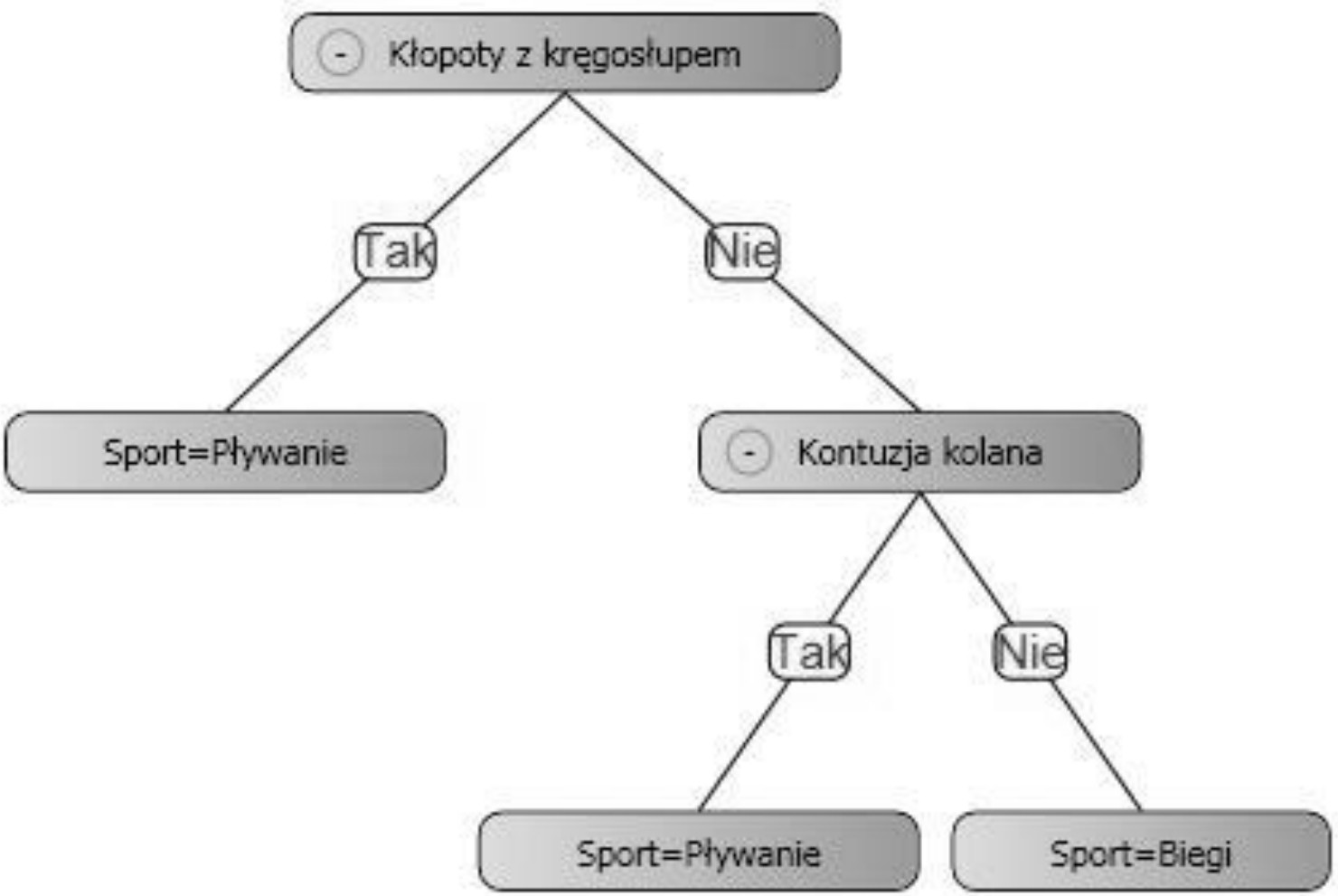
-Jeżeli Kłopoty z kręgosłupem=Nie & Kontuzja kolana=Tak, to Sport=Biegi

Oczywiście, nie każdy zbiór reguł będzie miał swój odpowiednik w postaci drzewa. Możliwe jest także, że algorytm generujący drzewo decyzyjne będzie generował również zbiór reguł decyzyjnych nie pokrywających się z tymi, które wynikają bezpośrednio z samej struktury drzewa. Zabieg ten stosuje się, aby dostarczyć reguły w formie jak najbardziej zminimalizowanej.

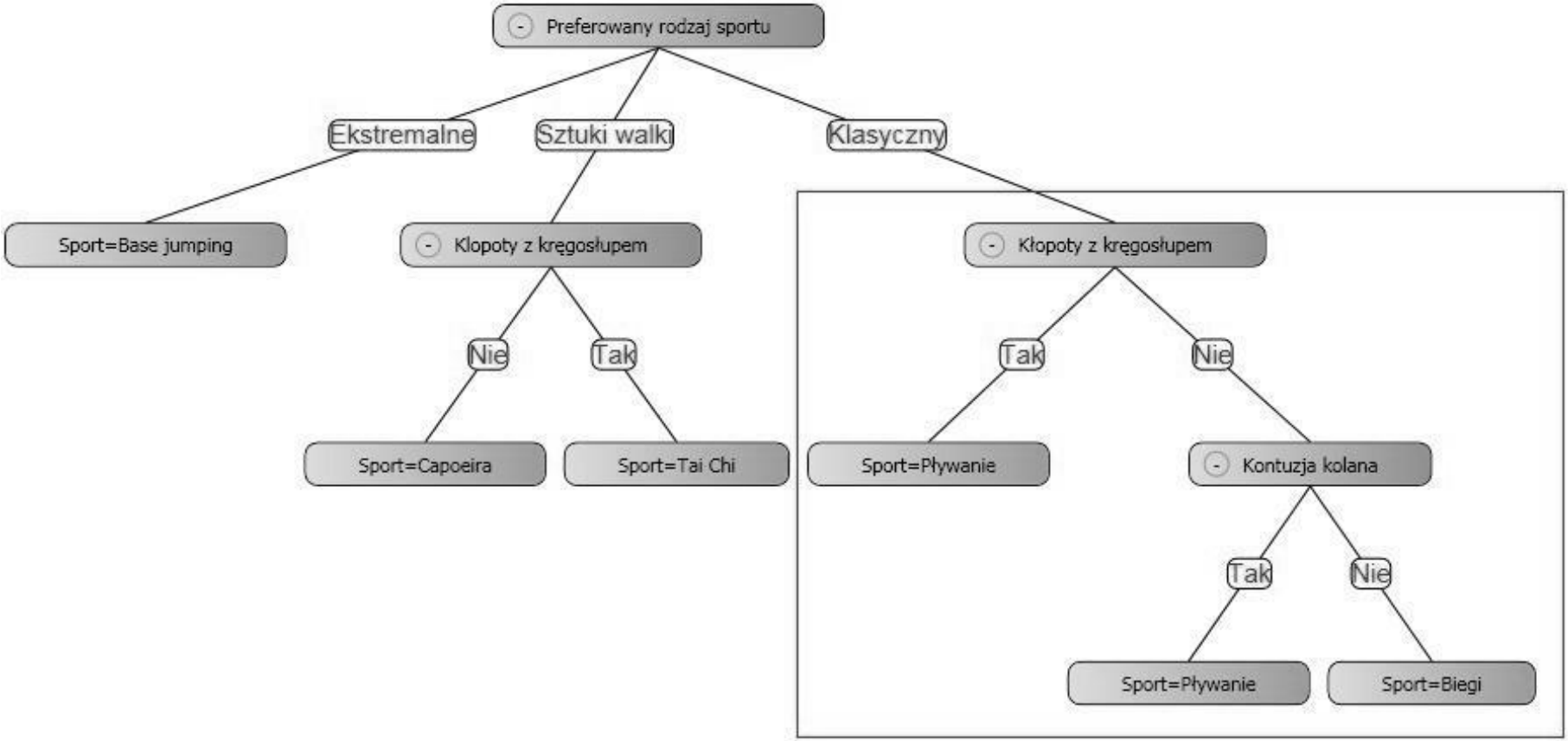
-Jeżeli Kłopoty z kręgosłupem=Tak, to Sport=Pływanie

-Jeżeli Kłopoty z kręgosłupem=Nie & Kontuzja kolana=Tak, to Sport=Pływanie

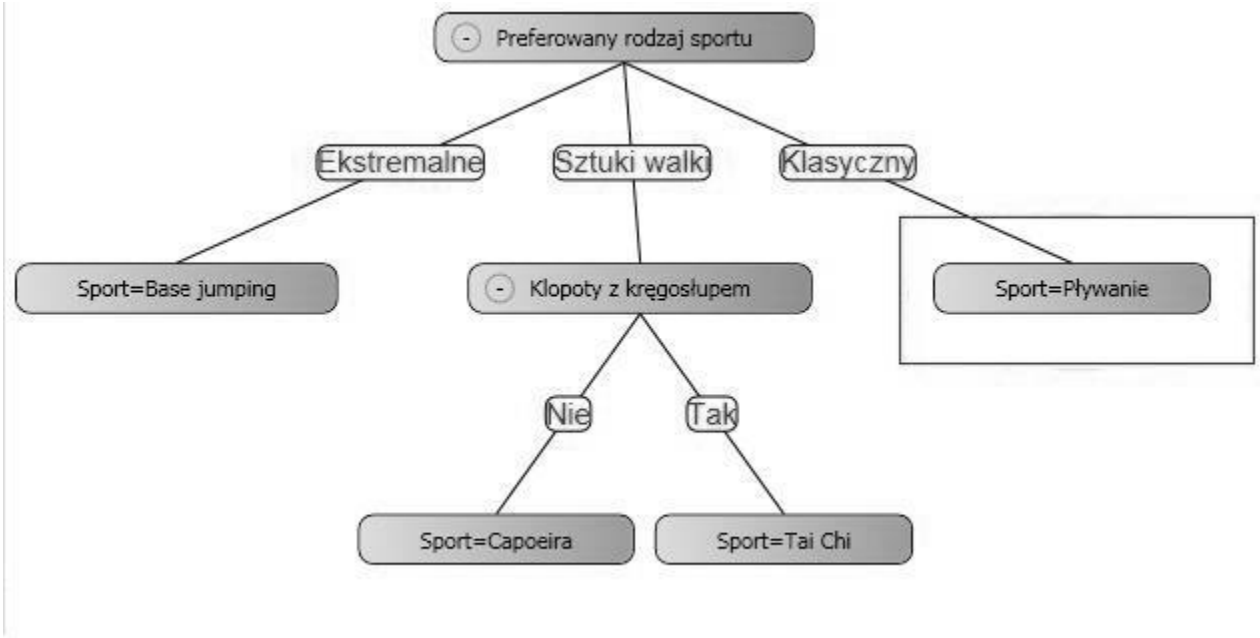
-Jeżeli Kłopoty z kręgosłupem=Nie & Kontuzja kolana=Tak, to Sport=Biegi



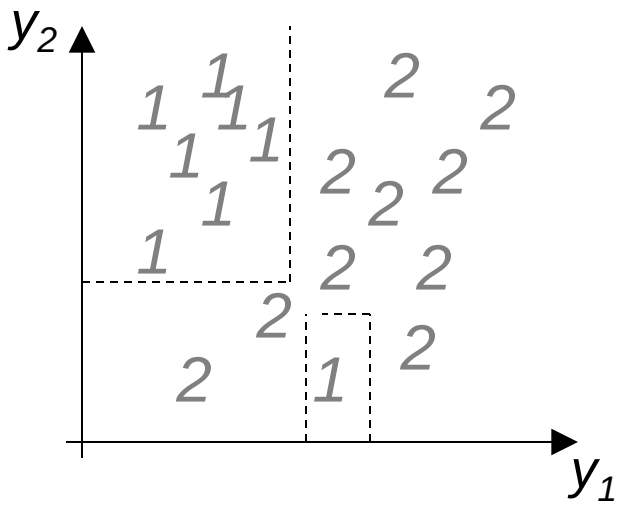
Przycinanie drzewa…



Przycinanie drzewa…



Przycinanie drzew



Cel: *zlikwidować nadmierne dopasowanie klasyfikatora do niepoprawnych danych.*

Przycinanie drzew

- Na podstawie oddzielnego zbioru przycinania
- Na podstawie zbioru uczącego
- Wykorzystujące zasadę minimalnej długości kodu

Metody przycinania drzew

Metody przycinania drzew można dzielić na grupy według różnych kryteriów.

Podział podstawowy to:

-Przycinanie w trakcie wzrostu (ang. pre-pruning). Drzewo jest przycinane w trakcie jego tworzenia tzn. algorytm rozrostu zawiera kryterium stopu zapobiegające powstawaniu zbyt dużych drzew. Bardzo trudno jest znaleźć takie kryterium, co prowadzi do tego, że metoda ta nie daje zbyt dobrych wyników.

Przykładem może być tutaj algorytm ID3 IV, gdzie jako jedno z kryteriów stopu został wprowadzony test $\chi 2$.

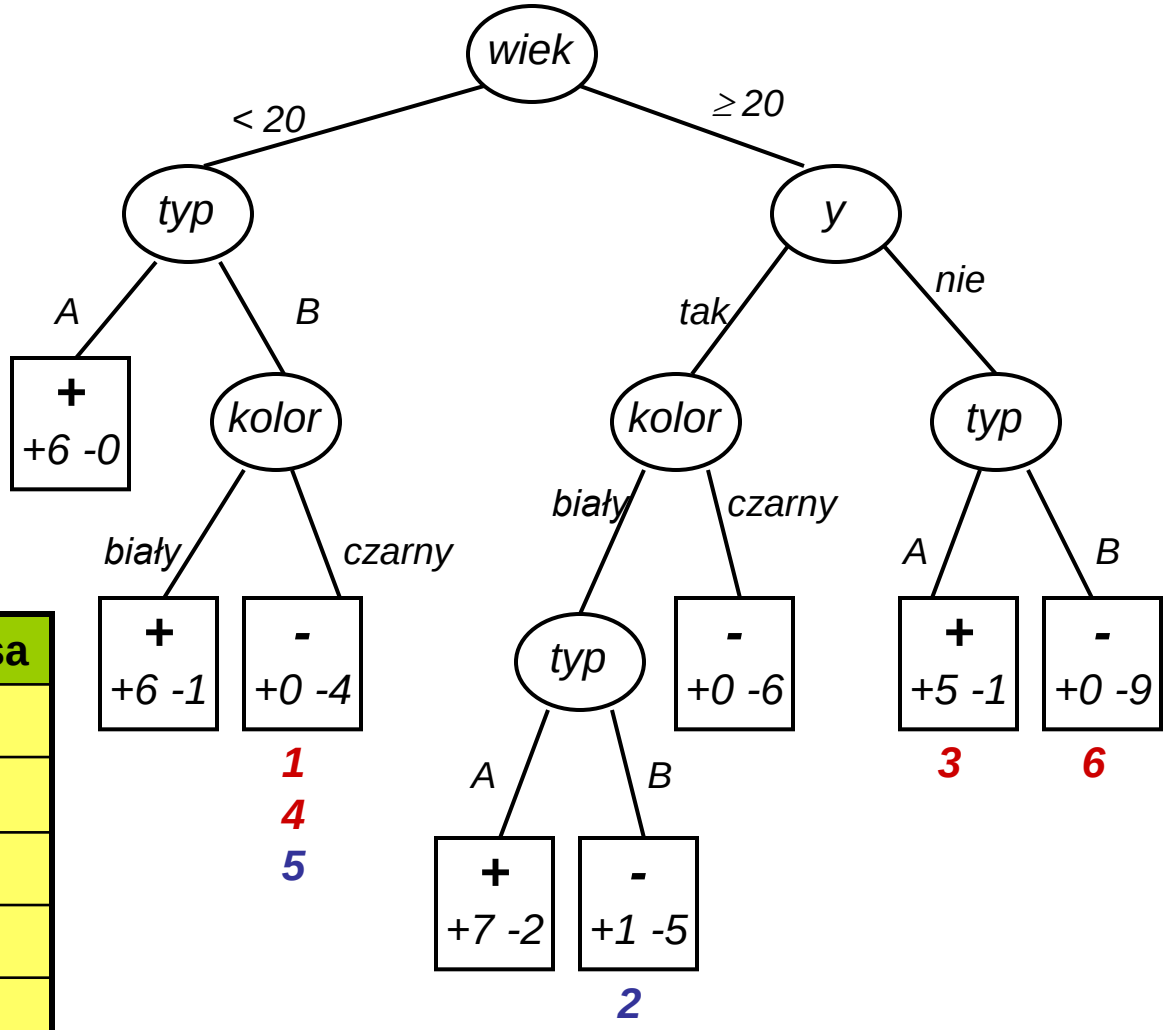
-Przycinanie rozrośniętego drzewa (ang. post-pruning). Generowane jest pełne drzewo, a przycinanie następuje potem. Proces ten jest rozpoczynany przy liściach i sukcesywnie „wędruje” w kierunku korzenia. Przykładem stosującym takie podejście może być algorytm C4.5.

-Metody mieszane – Kombinacje metod z dwóch poprzednich grup.

Przycinanie drzew – *reduced error pruning*

- Błąd szacowany na podstawie odrębnego zbioru przycinania.
- Węzły przeglądane od dołu.
- Poddrzewo T_t zastępowane liściem t gdy $\boldsymbol{error(t) \leq error(T_t)}$.
- Procedura powtarzana dopóki dalsze przycinanie nie zwiększa błędu.
- Zalety: prostota, niski koszt obliczeniowy.
- Wady: konieczność poświęcenia części danych na przycinanie; czasem drzewo zostaje przycięte zbyt mocno (zwłaszcza gdy zbiór przycinania jest znacznie mniejszy niż zbiór uczący).

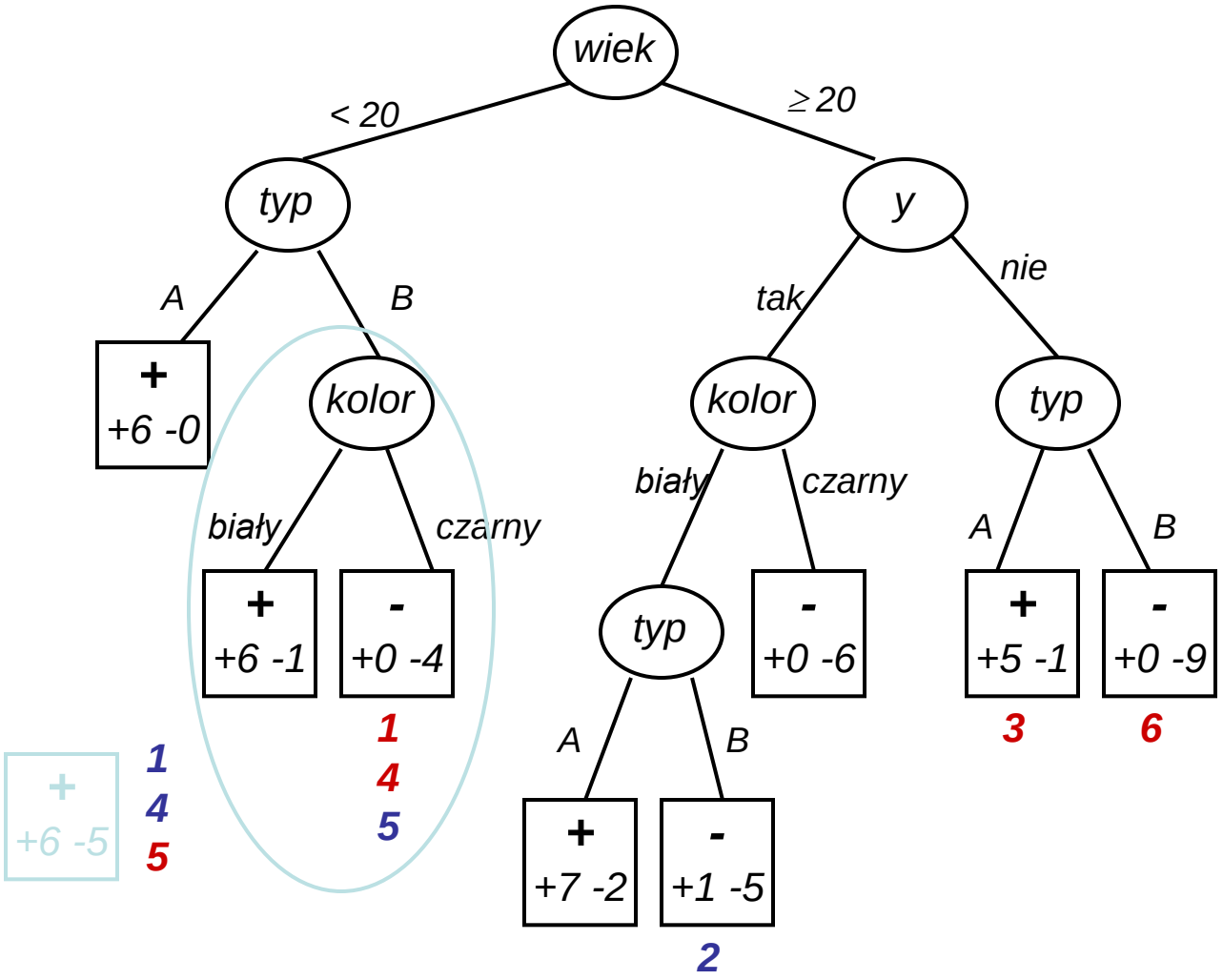
Przycinanie drzew – *reduced error pruning, przykład*



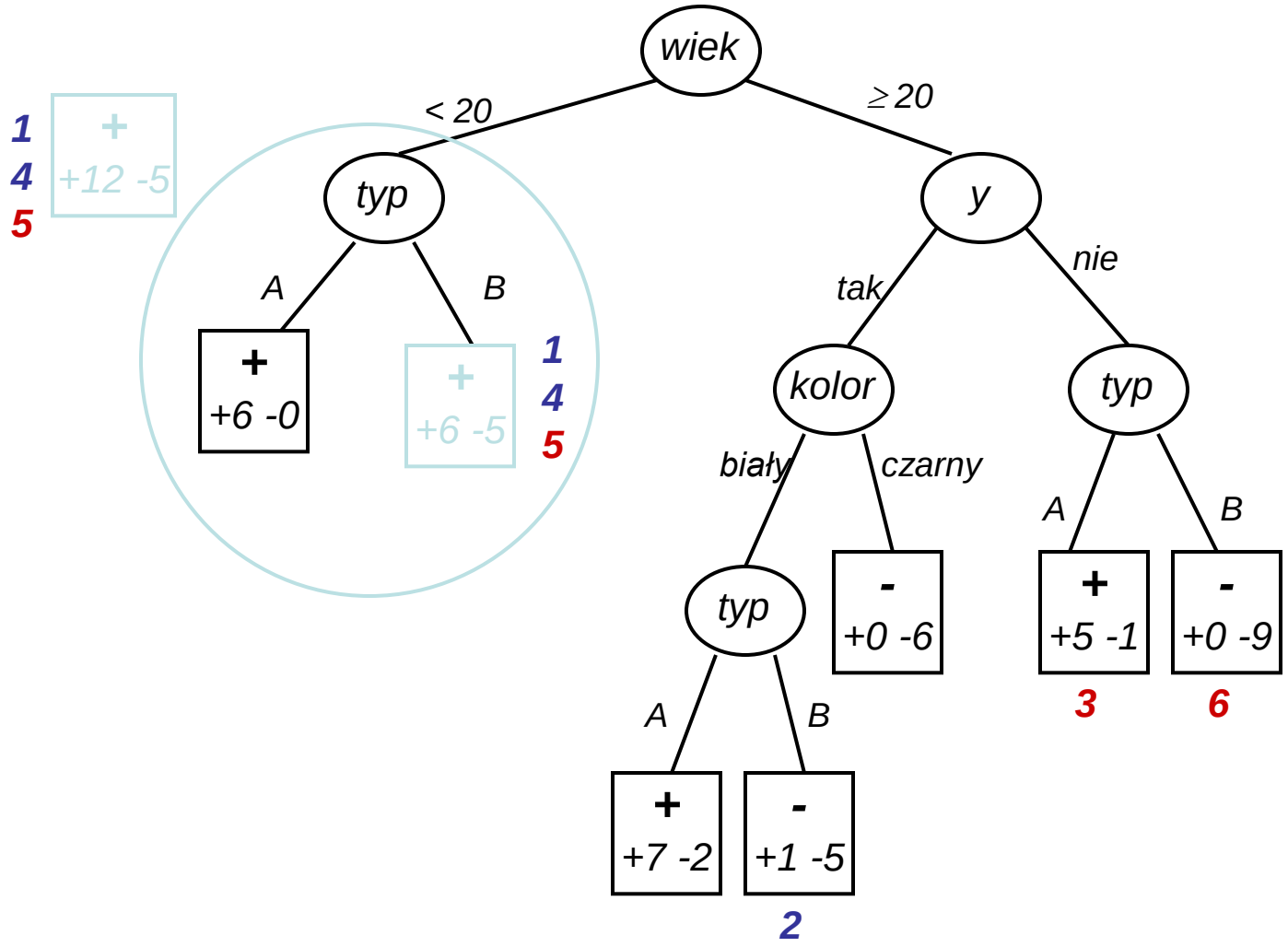
Zbiór przycinania

	kolor	wiek	typ	y	**Klasa**
1	czarny	11	B	tak	**+**
2	biały	23	B	tak	**-**
3	czarny	22	A	nie	**-**
4	czarny	18	B	nie	**+**
5	czarny	15	B	tak	**-**
6	biały	27	B	nie	**+**

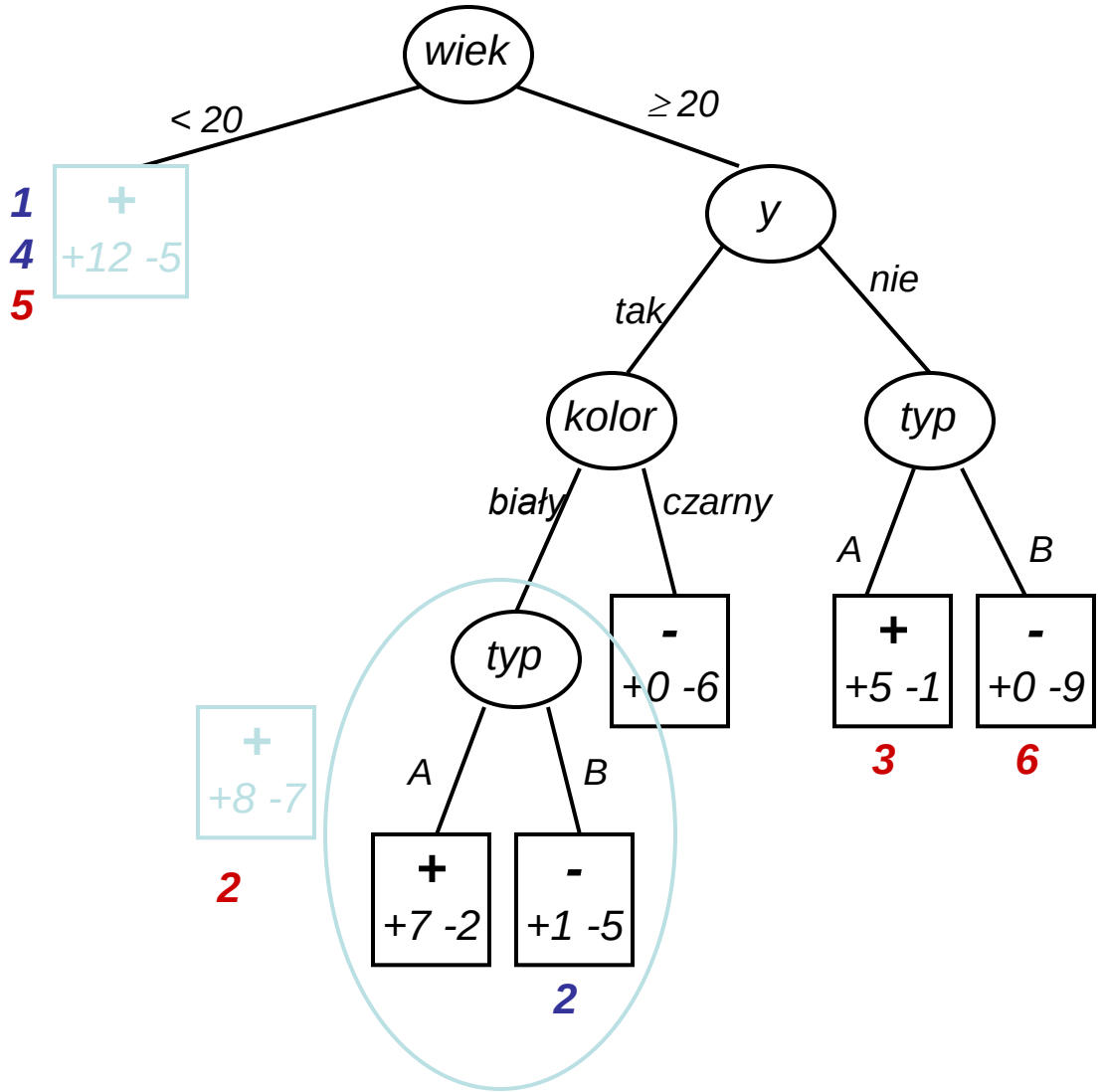
Przycinanie drzew – *reduced error pruning, przykład*



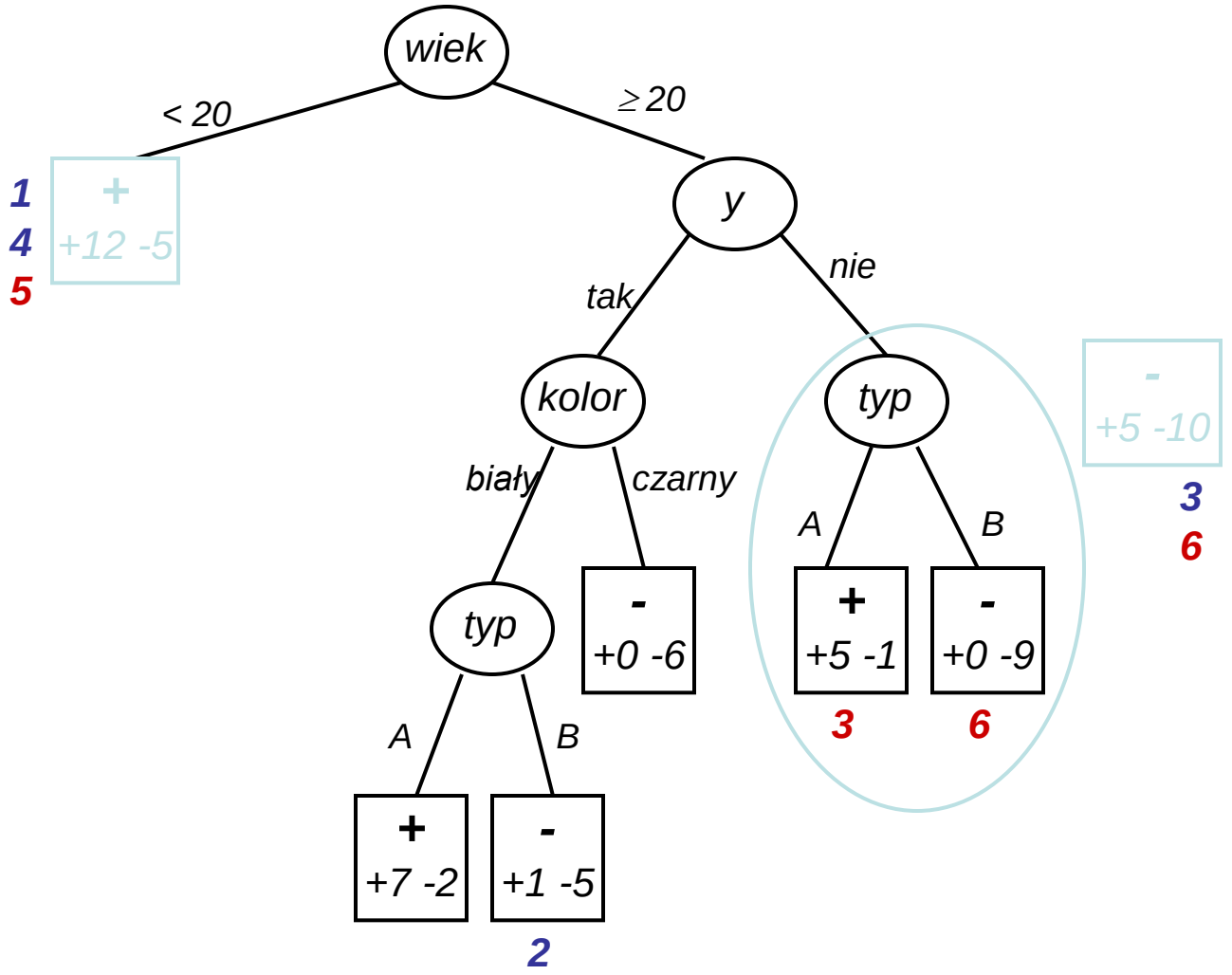
Przycinanie drzew – *reduced error pruning, przykład*



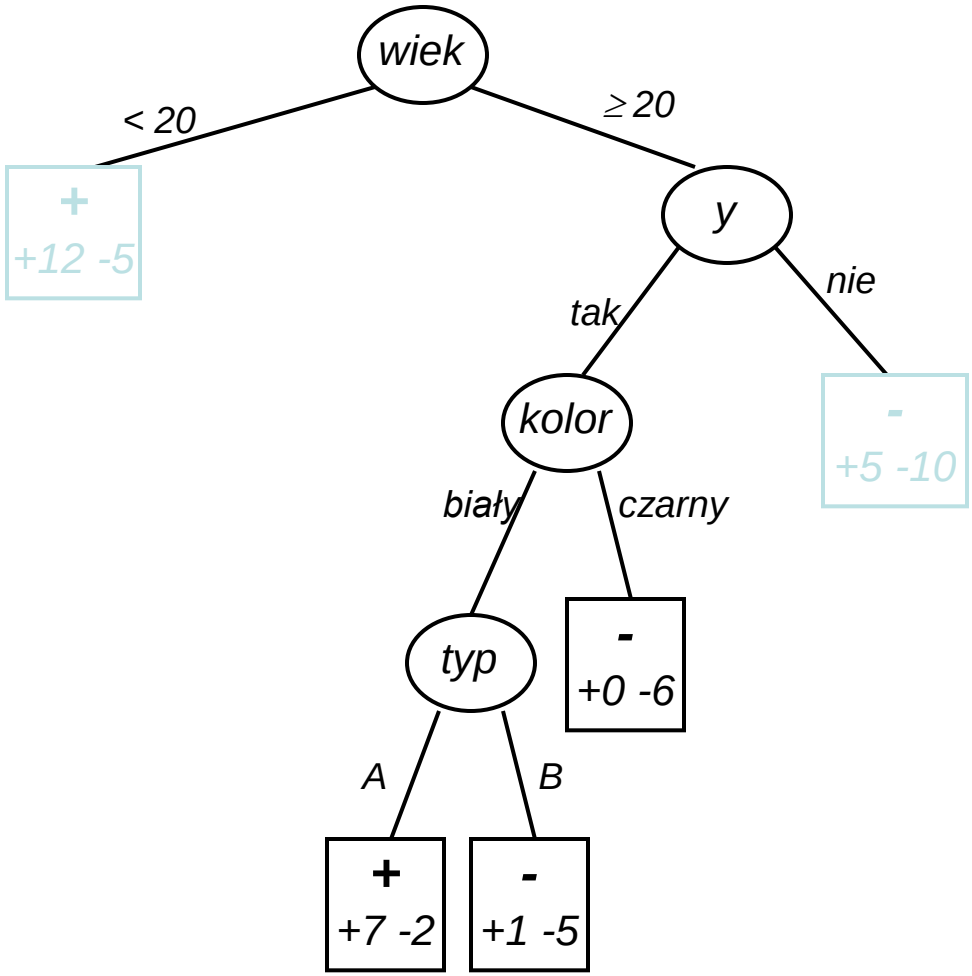
Przycinanie drzew – *reduced error pruning, przykład*



Przycinanie drzew – *reduced error pruning, przykład*



Przycinanie drzew – *reduced error pruning, przykład*



Po co stosujemy przycinanie drzew?

- Aby uniknąć nadmiernego przystosowania się hipotez do przykładów, tzw. 'przeuczenia' w algorytmach opartych na drzewach decyzyjnych.
- Nadmiernie dopasowane drzewo charakteryzuje się małym błędem dla przykładów trenujących, jednak na innych przykładach ma zbyt duży błąd. Taka sytuacja może mieć miejsce na przykład w przypadku zaszumienia zbioru trenującego i zbytniego dopasowania się algorytmu do błędnych danych.
- Przycinanie polega na zastąpieniu poddrzewa liściem reprezentującym kategorię najczęściej występującą w przykładach z usuwanego poddrzewa. Kategoria ta zwana jest kategorią większościową. Przycięcie drzewa, oprócz zwiększenia dokładności klasyfikacji dla danych rzeczywistych prowadzi także do uproszczenia budowy drzewa.

Zalety drzew decyzyjnych

- szybka klasyfikacja
- zrozumiały proces decyzyjny
- możliwość stosowania cech różnego typu
- efektywne z punktu widzenia przechowywania w pamięci

Wady drzew decyzyjnych

- im więcej klas oraz im bardziej się one nakładają, tym większe drzewo decyzyjne
- trudno zapewnić jednocześnie wysoką jakość klasyfikacji i małe rozmiary drzewa
- lokalna optymalizacja
- metody nieadaptacyjne

Bibliografia

- J.Koronacki, J.Ćwik: Statystyczne systemy uczące się, wydanie drugie, Exit,Warsaw,2008,rozdział 4.
- Gatnar E. : Symboliczne metody klasyfikacji danych, PWN, 1998