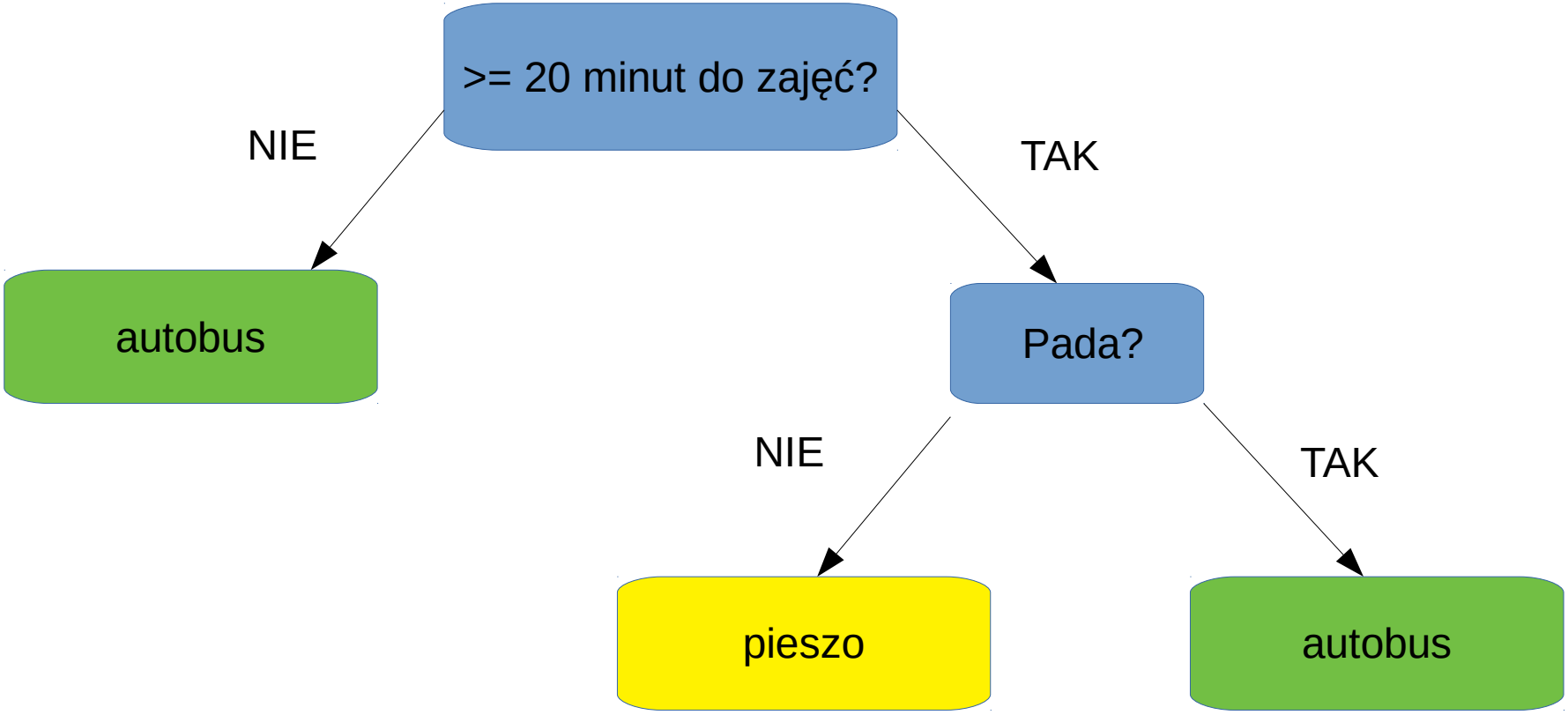
Drzewa decyzyjne

Kamil Michalak

Czym są drzewa decyzyjne?

- Decyzja/klasyfikacja na podstawie atrybutów.
- Testy/pytania w węzłach, odpowiedzi w liściach
- Jak dotrzeć na uczelnię?

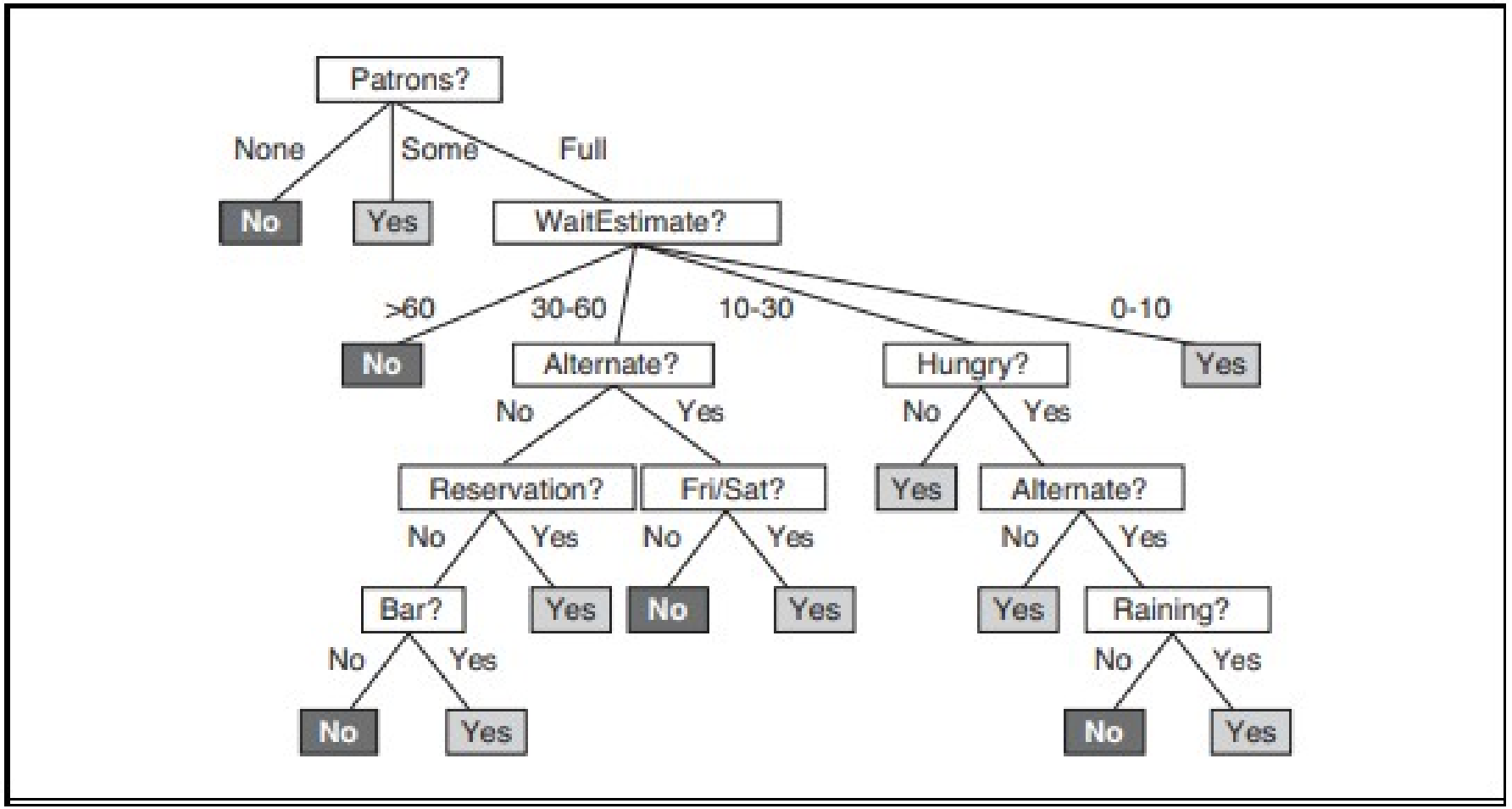


Przykład (*AI: A Modern Approach*)

Czy zaczekać na stolik w restauracji?

- *Alternate*: czy jest inna restauracja w pobliżu.
- *Bar* : czy restauracja ma bar, w którym można zaczekać.
- *Fri/Sat*: czy jest piątek lub sobota.
- *Hungry*: czy jesteśmy bardzo głodni.
- *Patrons*: ile osób jest w restauracji (*None/Some/Full*).
- *Price*: przedział cenowy (*$, $$, $$$*).
- *Raining*: czy pada.
- *Reservation*: czy mamy rezerwację.
- *Type*: rodzaj restauracji (*French/Italian/Thai/burger*).
- *WaitEstimate*: czas oczekiwania oszacowany przez właściciela (0–10 minut, 10–30, 30–60, lub >60)

Przykład (*AI: A Modern Approach*)



Utworzenie drzewa decyzyjnego

function DECISION-TREE-LEARNING($examples$, $attributes$, $parent\_examples$) **returns** a tree

if $examples$ is empty **then return** PLURALITY-VALUE($parent\_examples$)
else if all $examples$ have the same classification **then return** the classification
else if $attributes$ is empty **then return** PLURALITY-VALUE($examples$)
else
  $A \leftarrow \text{argmax}_{a \in attributes}$ IMPORTANCE(a, $examples$)
  $tree \leftarrow$ a new decision tree with root test A
  for each value v_k of A **do**
    $exs \leftarrow \{e : e \in examples$ **and** $e.A = v_k\}$
    $subtree \leftarrow$ DECISION-TREE-LEARNING(exs, $attributes - A$, $examples$)
    add a branch to $tree$ with label $(A = v_k)$ and subtree $subtree$
  return $tree$

Zbiór danych

Example	Input Attributes										Goal
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
$\mathbf{x}_1$	*Yes*	*No*	*No*	*Yes*	*Some*	*$$$*	*No*	*Yes*	*French*	*0–10*	$y_1 = Yes$
$\mathbf{x}_2$	*Yes*	*No*	*No*	*Yes*	*Full*	*$*	*No*	*No*	*Thai*	*30–60*	$y_2 = No$
$\mathbf{x}_3$	*No*	*Yes*	*No*	*No*	*Some*	*$*	*No*	*No*	*Burger*	*0–10*	$y_3 = Yes$
$\mathbf{x}_4$	*Yes*	*No*	*Yes*	*Yes*	*Full*	*$*	*Yes*	*No*	*Thai*	*10–30*	$y_4 = Yes$
$\mathbf{x}_5$	*Yes*	*No*	*Yes*	*No*	*Full*	*$$$*	*No*	*Yes*	*French*	*>60*	$y_5 = No$
$\mathbf{x}_6$	*No*	*Yes*	*No*	*Yes*	*Some*	*$$*	*Yes*	*Yes*	*Italian*	*0–10*	$y_6 = Yes$
$\mathbf{x}_7$	*No*	*Yes*	*No*	*No*	*None*	*$*	*Yes*	*No*	*Burger*	*0–10*	$y_7 = No$
$\mathbf{x}_8$	*No*	*No*	*No*	*Yes*	*Some*	*$$*	*Yes*	*Yes*	*Thai*	*0–10*	$y_8 = Yes$
$\mathbf{x}_9$	*No*	*Yes*	*Yes*	*No*	*Full*	*$*	*Yes*	*No*	*Burger*	*>60*	$y_9 = No$
$\mathbf{x}_{10}$	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	*$$$*	*No*	*Yes*	*Italian*	*10–30*	$y_{10} = No$
$\mathbf{x}_{11}$	*No*	*No*	*No*	*No*	*None*	*$*	*No*	*No*	*Thai*	*0–10*	$y_{11} = No$
$\mathbf{x}_{12}$	*Yes*	*Yes*	*Yes*	*Yes*	*Full*	*$*	*No*	*No*	*Burger*	*30–60*	$y_{12} = Yes$

Figure 18.3 Examples for the restaurant domain.

Wynikowe drzewo

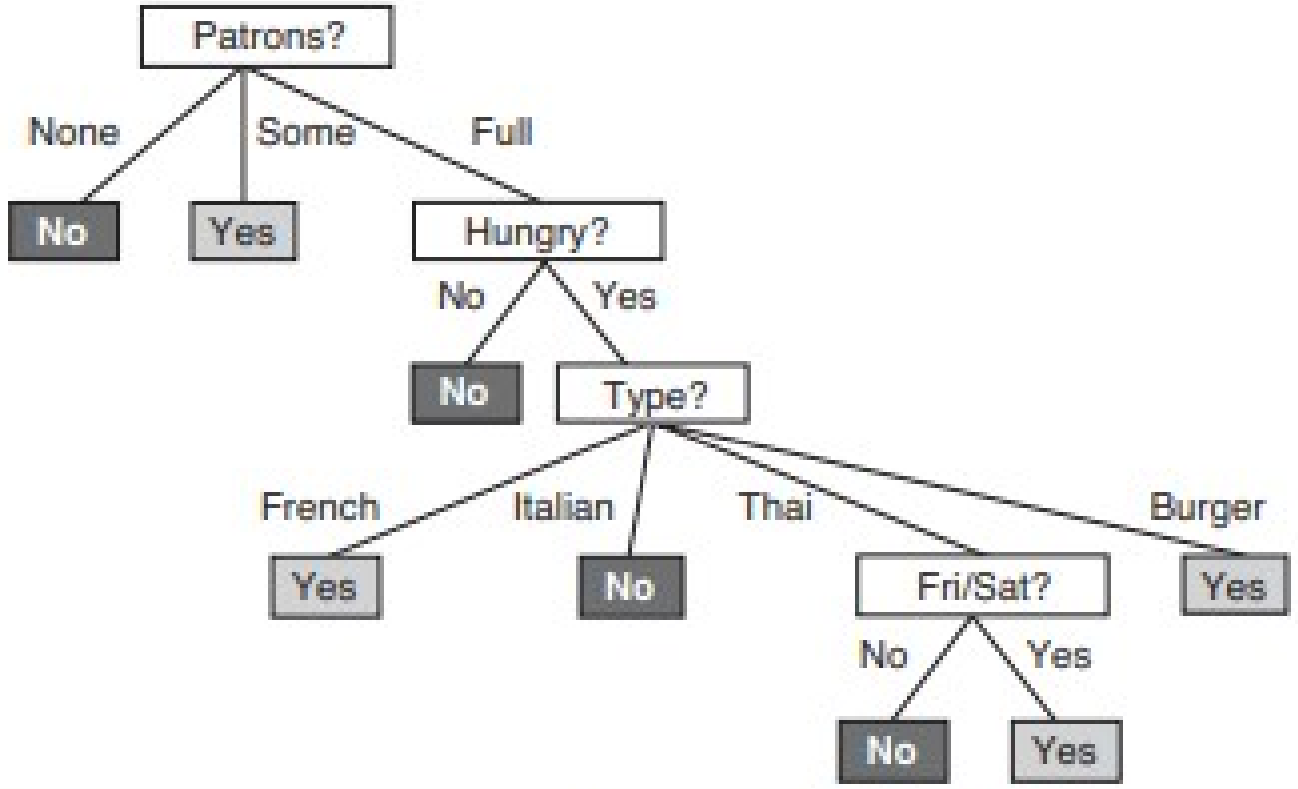


Figure 18.6 The decision tree induced from the 12-example training set.

Wybór cechy do testowania

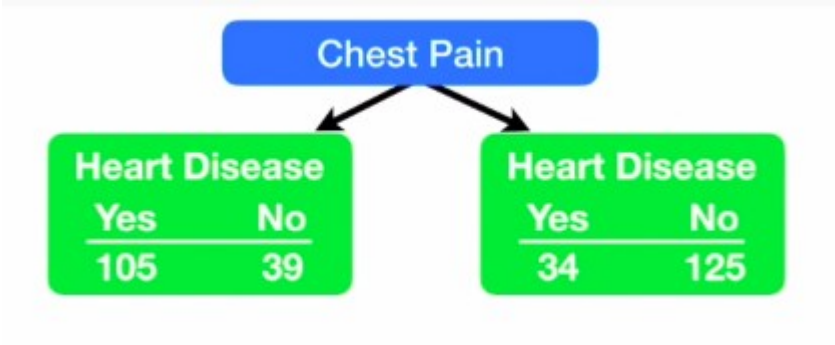
- Sposób 1: maksymalizowanie spadku entropii
- Entropia – niepewność wystąpienia danego zdarzenia w danej chwili, jeśli zdarzenie jest pewne to entropia wynosi 0.
- Wzór na entropię zmiennej losowej V o zbiorze wartości $\{v_1, \ldots, v_n\}$

$$H(V) = -\sum_{k=1}^{n} P(v_k) \cdot \log_2(P(v_k))$$

Wybór cechy do testowania

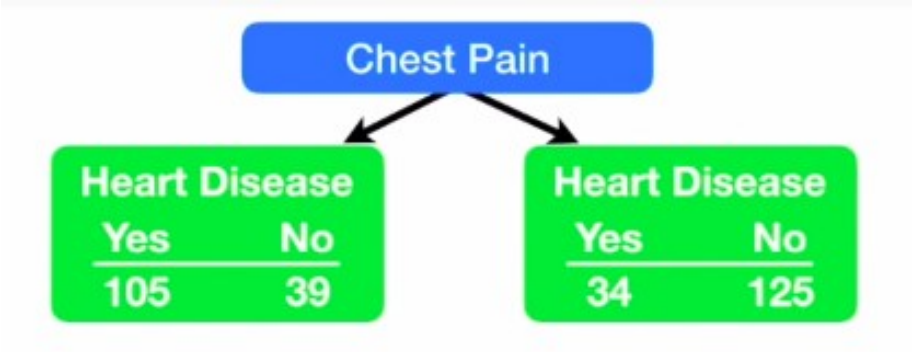
- Sposób 2: „Gini impurity”

Chest Pain	Good Blood Circulation	Blocked Arteries	Heart Disease
No	No	No	No
Yes	Yes	Yes	Yes
Yes	Yes	No	No
Yes	No	???	Yes
etc…	etc…	etc…	etc…

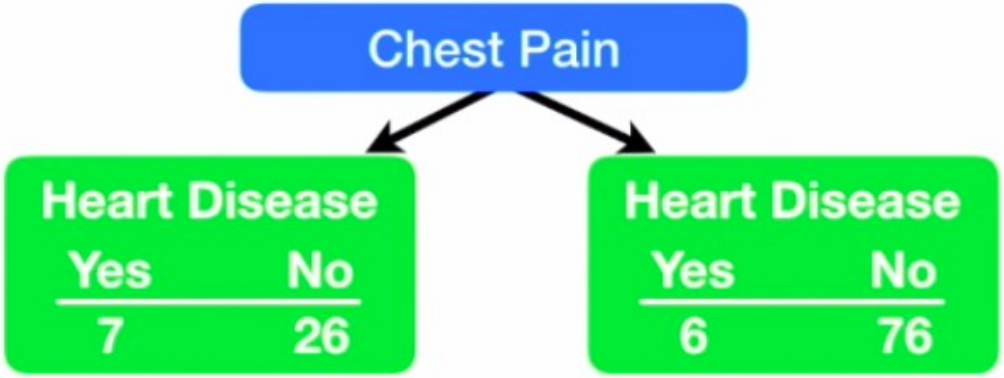


Wybór cechy do testowania

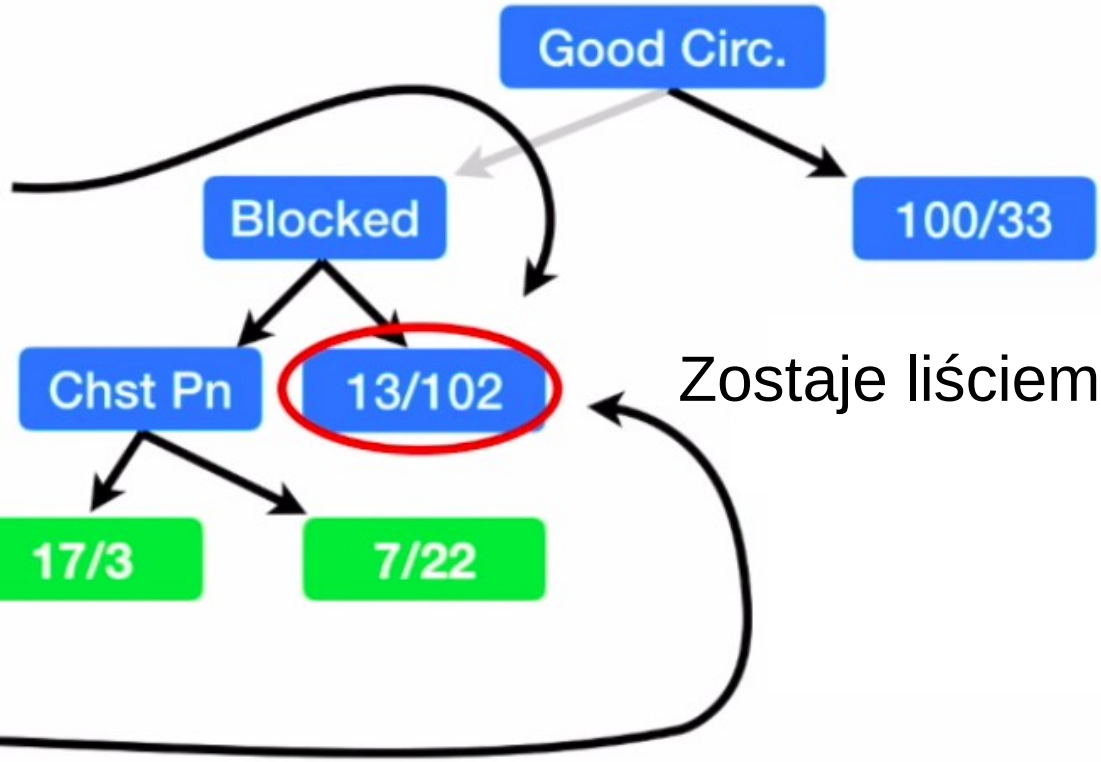
- Sposób 2: „Gini impurity”



- $G(L) = 1 – P(\text{„Yes”})^2 - P(\text{„No”})^2 = 0.395$
- $G(R) = 0.336$
- G(„Chest Pain”) = średnia ważona z L i R
- Wybieramy cechę z najmniejszą wartością G



G(„Chest Pain”) = 0.29



Zostaje liściem

G(węzeł) = 0.2

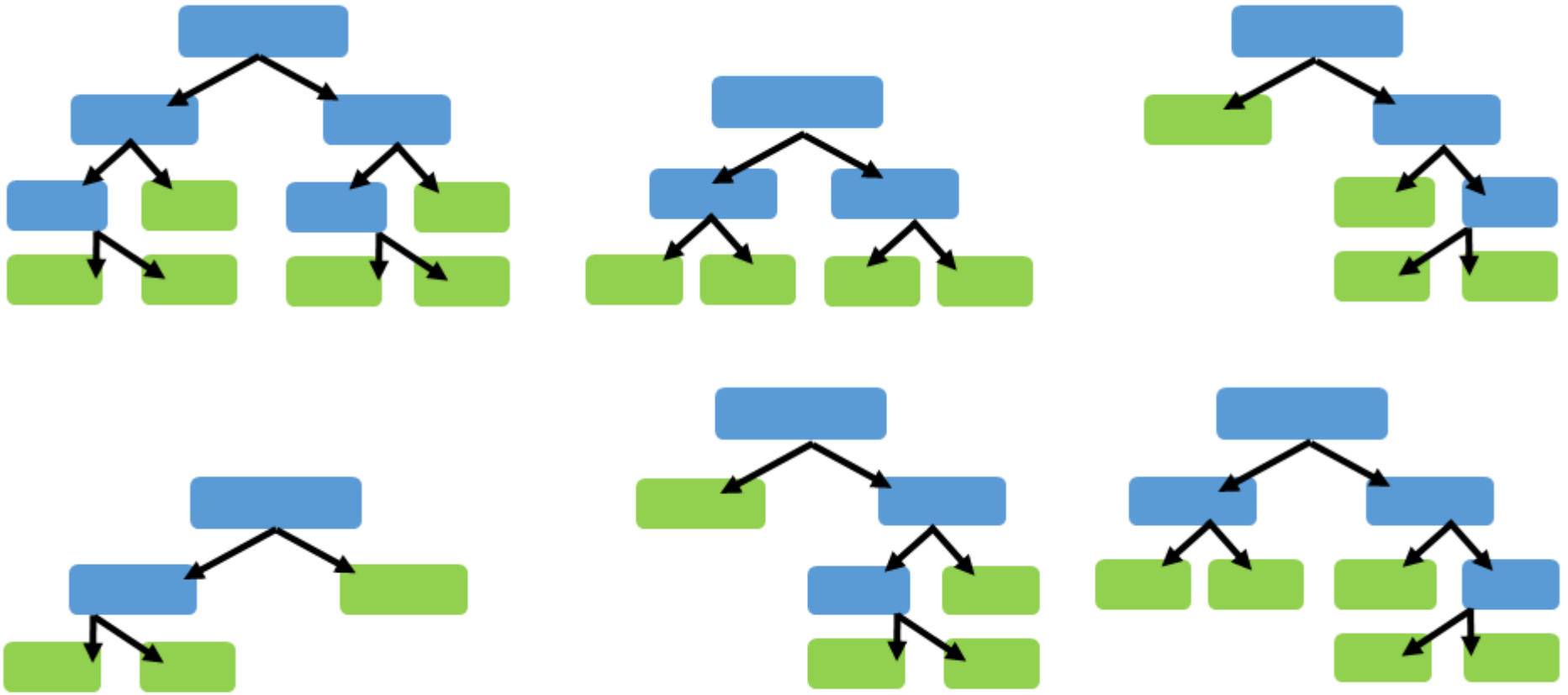
Zastosowania drzew decyzyjnych

- Medycyna – diagnozowanie pacjentów na podstawie reguł znalezionych w bazie danych
- Ocena inwestycji
- Udzielanie kredytów

Problem

- „overfitting” - drzewo działa dobrze dla zbioru uczącego, ale nie dla dowolnych danych

Las losowy (Random forest)



Las losowy (Random forest)

- Tworzymy n drzew
- Dokonujemy klasyfikacji obiektu na każdym drzewie
- Ostateczna klasyfikacja – taka jak na większości drzew.

Tworzenie lasu losowego

- Krok 1: Tworzymy „bootstrapped dataset”

  Dla oryginalnego k-elementowego zbioru danych losujemy k-elementów z powtórzeniami.

- Krok 2: Tworzymy n drzew decyzyjnych.
  W każdym kroku tworzenia drzewa wybieramy najlepszą cechę jedynie z podzbioru wszystkich cech.

Testowanie lasu losowego

- Bierzemy dane, które nie znalazły się w „bootstrapped dataset” („out-of-bag”)
- Dla każdego rekordu sprawdzamy czy jest dobrze klasyfikowany przez las
- Miarą skuteczności lasu jest % poprawnie sklasyfikowanych danych „out-of-bag”.

Testowanie lasu losowego

- Teraz możemy sprawdzić jak liczne podzbiory cech przy generowaniu lasu dają najlepszą skuteczność.
- Zwykle ~ $\sqrt{liczba\,cech}$

Dziękuję za uwagę