Uczenie Maszynowe i Sieci Neuronowe
Raport

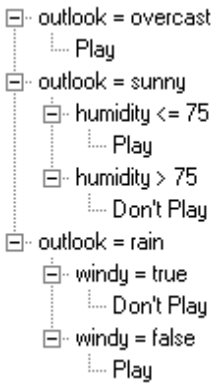
Karol Bonenberg

6 kwietnia 2008

Zadanie 1. Generowanie drzewa

a. Zbiór uczący zawiera 14 przykładów, każdy jest opisany czterema atrybutami.

b. Wygenerowane drzewo:



Rysunek 1: Drzewo decyzyjne dla zbioru uczącego golf.dat

c. Wygenerowane drzewo nie zawiera żadnych niepewnych danych, każda ścieżka prowadzi do jednoznacznej, nieobarczonej błędem decyzji. Przeprowadzenie pruningu nie spowodowało zmniejszenia rozmiaru drzewa.

d. Drzewo posiada 3 węzły warunkowe i 5 liści.

e. Przykładowa ścieżka dla przykładu: $\{outlook = sunny, temperature = 80, humidity = 70, windy = true\}$

- $outlook = sunny$
- $humidity \leqslant 75$
- $decision = Play$

f. Estymata błędu dla drzewa oryginalnego wynosi 0%, natomiast dla drzewa uproszczonego wynosi ona 38.5%, mimo iż oba drzewa są identyczne. Spowodowane jest to tym, iż estymata dla drzewa oryginalnego jest ilością błędów popełnionych przy tworzeniu drzewa, natomiast algorytm pruningu wylicza estymatę błędu wzorami heurystycznymi.

g. Macierz pomyłek:

Orig. \ C4.5	Play	Don't Play
Play	9	
Don't Play		5

Rysunek 2: Macierz pomyłek drzewa decyzyjnego wygenerowanego dla zbioru uczącego golf.dat

Ponieważ w danych wejściowych nie było żadnych wieloznaczności, wygląd macierzy pomyłek był łatwy do przewidzenia.

Zadanie 2. Konsultowanie

a. Przeprowadzilismy konsultację dla przykładu $\{outlook = sunny, temperature = 80, humidity = 70, windy = true\}$. Wynik był pewny w 100%:

Class	Probability
Play	1.00 [0.50 - 1.00]
Don't Play	0.00 [0.00 - 0.50]

Rysunek 3: Rezultat konsultacji jednoznacznego przykładu

b. Dla przykładu $\{outlook = rain\}$ decyzja to *Play* z prawdopodobieństwem 60%. Prawdopodobieństwo to wynika z tego, iż decyzja *Play* jest w węźle $\{outlook = rain\}$ popierana przez 3 przykłady, natomiast decyzja przeciwna, *Don't Play*, jest popierana przez 2 przykłady.

Class	Probability
Play	0.60 [0.38 - 0.80]
Don't Play	0.40 [0.20 - 0.62]

Rysunek 4: Rezultat konsultacji przykładu z jednym pewnym atrybutem

c. W przykładzie dla $\{outlook = overcast(p = 0.2) \vee outlook = rain(p = 0.8)\}$ decyzja to *Play* z prawdopodobieństwem 68%. Prawdopodobieństwo to jest średnią ważoną szansy wystąpienia decyzji w każdym węźle z osobna. Dla przypadku $outlook = rain$ szukane prawdopodobieństwo, jak już wiemy, wynosi 60%, natomiast dla $outlook = overcast$ prawdopodobieństwo to wynosi 100%. Jak łatwo można sprawdzić, $0.2 \cdot 100\% + 0.8 \cdot 60\% = 68\%$.

Class	Probability
Play	0.68 [0.44 - 0.84]
Don't Play	0.32 [0.16 - 0.56]

Rysunek 5: Rezultat konsultacji przykładu z atrybutem przyjmującym dwie wartości z zadanym prawdopodobieństwem

Zadanie 3. Różnica między *gain ratio* a *gain* w praktyce

a. W zbiorze TESTGAIN znajduje się 8 przykładów, każdy opisany dwoma atrybutami.

b. Na poniższym rysunku są widoczne dwa drzewa. Drzewo po lewej wygenerowane jest przy użyciu metody *info gain* a po prawe przy użyciu *gain ratio*. Drzewo po lewej jest bardziej rozgałęzione, gdyż metoda *info gain* ma tendencję do wybierania atrybutów posiadających dużo wartości, gdyż dzielą one przestrzeń problemu na wiele części zawierających niedużo przykładów co prowadzi do znacznego zmniejszenia entropii. Nie jest to zjawisko porządane, gdyż powstałe w ten sposób drzewa są malo ogólne i podatne na zjawisko przeuczenia.

Rysunek 6: Dwa drzewa wygenerowane dla tego samego zbioru uczącego – lewe przy użyciu *info gain*, a prawe *gain ratio*

Zadanie 4. Grupowanie wartości atrybutów

a. Przy włączonej opcji grupowania algorytm *info gain* wygeneruje drzewo posiadające dwa liście czyli dwukrotnie mniejsze niż bez tej opcji (po lewej drzewo wygenerowane bez grupowania, po prawej z grupowaniem).

Rysunek 7: Dwa drzewa wygenerowane dla tego samego zbioru uczącego – lewe przy użyciu *info gain*, a prawe *info gain* z grupowaniem

b. Zbiór przykładów CRX składa się z 490 przykładów. Przykłady posiadają 15 atrybutów, z których 6 przyjmuje wartości ciągłe, a dwa spośród przyjmujących wartości dyskretne (A6 i A7) pzryjmują odpowiedio 14 i 9 różnych wartości. Opisuje on problem przyznawania kart kredywotych, jednak nazwy i wartości atrybutów zostały zamienione na nic nieznaczące symbole aby chronić dane osobowe klientów. Drzewo wygenerowane bez opcji grupowania metodą *info gain* po pruningu składa sie z 48 węzłów i na biorze testowym popełnia 41 błędów (20.5%). Drzewo wygenerowane z opcja grupowania składa się z 38 węzłów i popełnia 35 błędów (17.5%) czyli użycie grupowania podniosło skuteczność drzewa.

c. Na poniższym rysunku przedstawione są macierze pomyłek dla zbioru uczącego i testowego. Widać, iż zdarzają się przypadki, w których klient dostał kartę kredytową, chociaż nie powinien i nie dostał chociaż powinien. Z punktu widzienia banku pierwsza sytuacja jest o wiele gorsza, ponieważ w przypadku niewypłacalności klienta bank będzie na tym tracił. Z drugiej strony nieprzyznanie karty kredytowej osobie, która na nią niezasługuje nie przyniesie bankowi strat w przyszłości, więc macierz kosztów pomyłek powinna być tak stworzona, aby niedopuścić do pierwszej z wymienionych sytuacji.

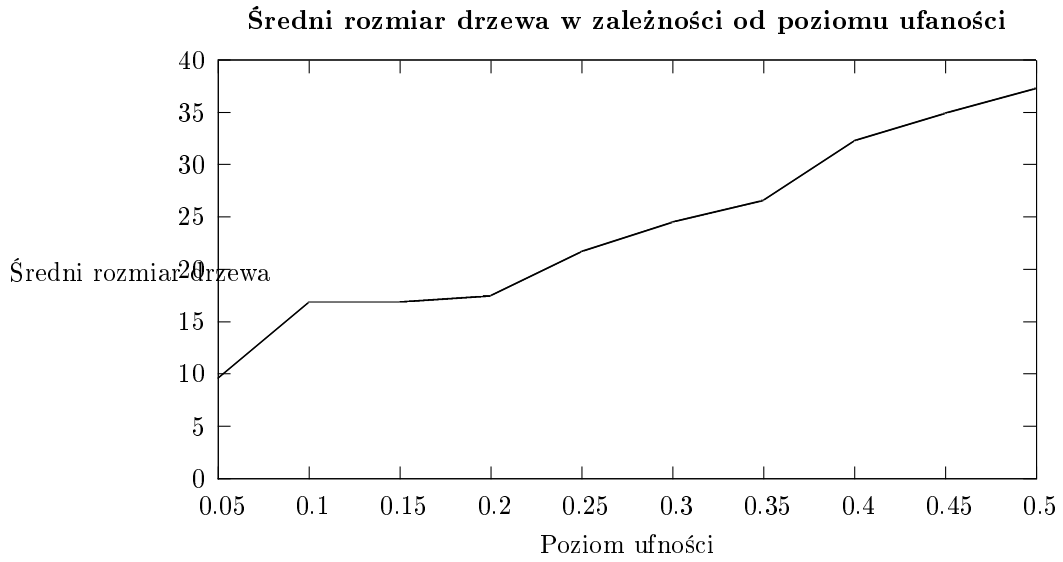
Orig. \ C4.5	+	-
+	198	19
-	14	259

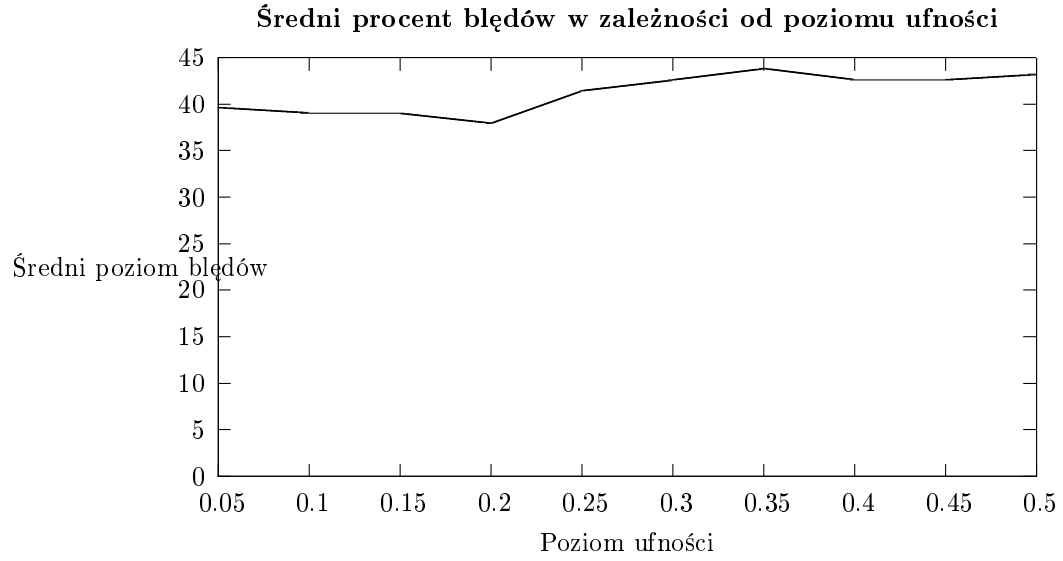
Orig. \ C4.5	+	-
+	78	12
-	23	87

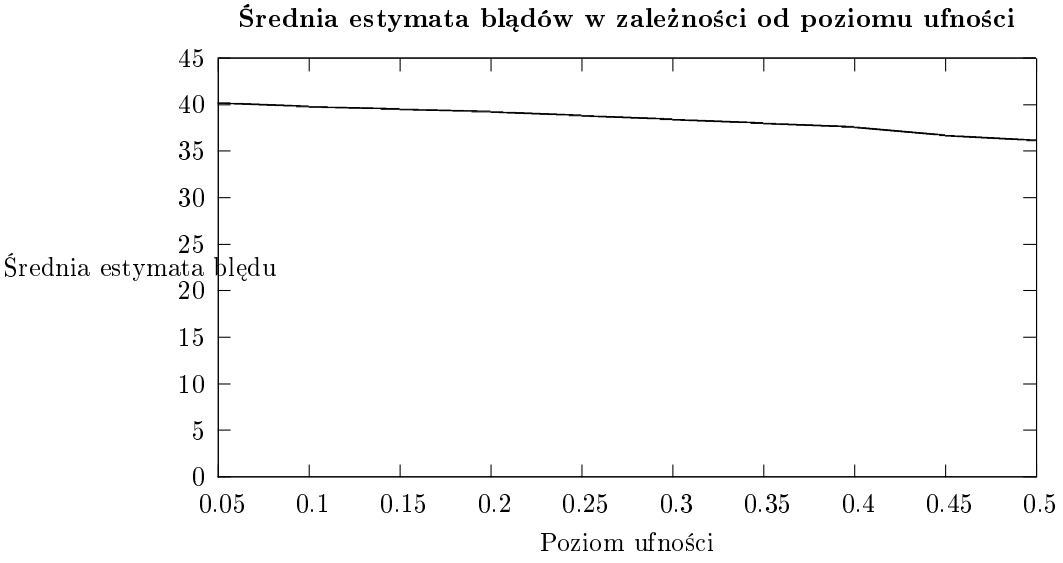
Rysunek 8: Po lewej macierz pomyłek dla zbioru uczącego, po prawej dla testowego

Zadanie 5. Poszukiwanie optymalnej wielkości drzewa uproszczonego

a. Poszukiwanie optymalnej wielkości drzewa uproszczonego przez dobór poziomu ufności procedury upraszczającej dla zbioru MONK2







b. Poszukiwanie optymalnej wielkości drzewa uproszczonego poprzez *prepruning*. Jako kryterium optymalizacyjne przyjęto liczbę błędów klasyfikacyjnych popełnianych dla przypadków testowych. Dla zbioru CRX najlepszy wynik osiągnięto dla $minimum\_objects = 4$ ($Errors(test) = 16.0\%$). Zwiększenie wartości tego parametru powoduje generowanie drzewa bardziej odpornego na szumy (pojedyńczy błąd nie wpływa

na strukture drzewa).

c. Analizując drzewo otrzymane w poprzednim podpunkcie można wskazać kilka słabych węzłów w jego strukturze. Ze względu na brak zaznaczonej opcji *Subsetting* przy węzłach decyzyjnych operujących na wartościach dyskretnych powstają węzły o pokryciu mniejszym niż narzucony poziom. Węzeł ($A15 \leqslant 228, A6 = j$) posiada tylko 2 obiekty. Niekorzystnym efektem narzucenia minimalnej liczby wierzchołków w węźle jest niedzielenie węzła, zawierającego elementy z dwóch klas, gdy każda z grup jest mniejliczna niż wartość parametru *minimum_objects*. Przypadek taki można zaobserwować w węźle ($A15 \leqslant 228, A6 = d$). Węzeł zawiera 6 obiektów, stanowiących 2 podgrupy, które zostały połączone ze względu na małą liczność.

```
Node information

Items:    6.0
Errors:   3.0
Estimate: 50.0%

Class distribution:
+  3.0
-  3.0

Decision: +
```

Rysunek 9: węzeł ($A15 \leqslant 228, A6 = d$)

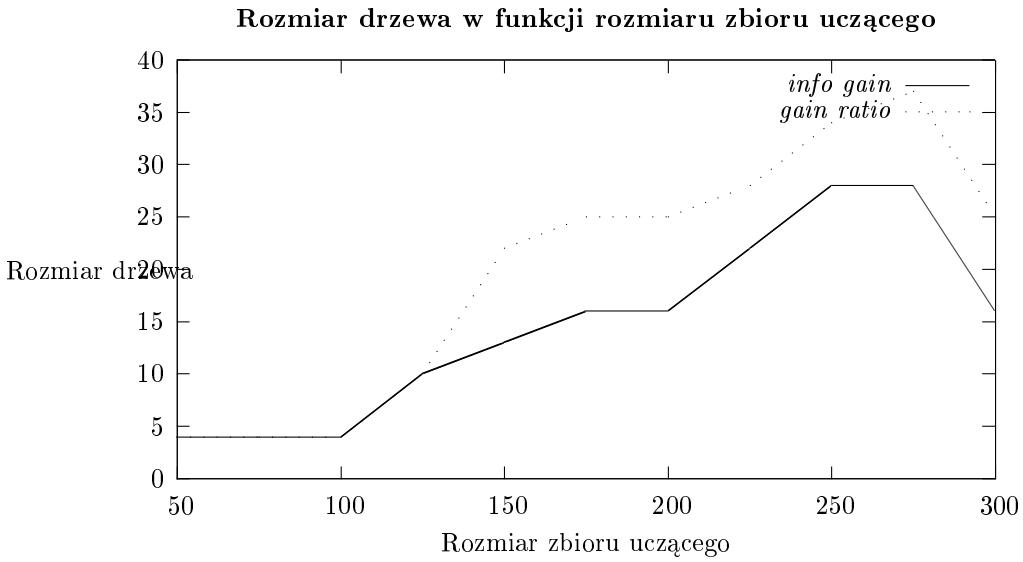
Zadanie 6. Windowing

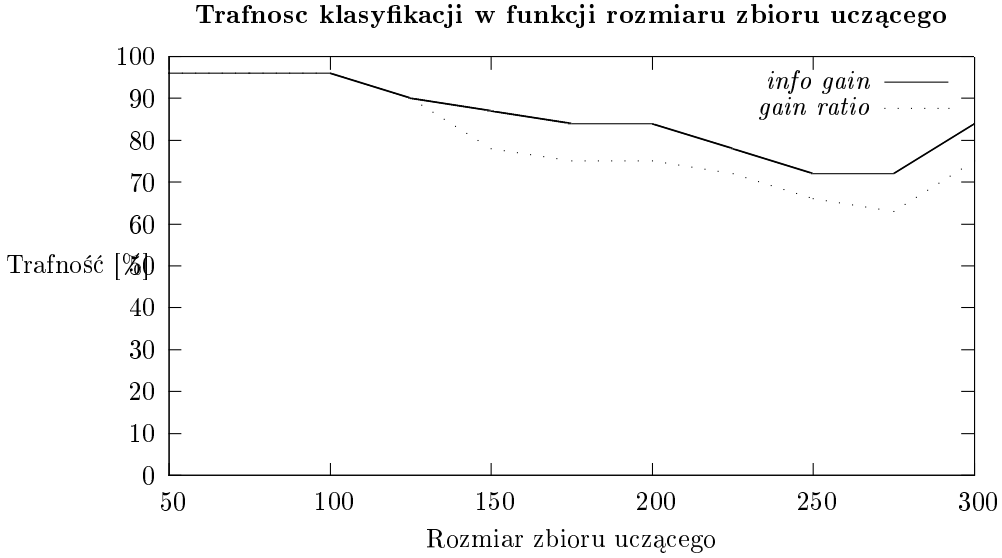
a. Technika *windowing* polega na przyrostowym tworzeniu drzewa ze zbioru uczącego. W pierwszym przebiegu algorytmu ze zbioru uczącego wybierana są losowe przykłady, w ilości określonej przez zmienną *Initial window size*, budowane jest z nich drzewo decyzyjne a następnie przy każdej iteracji wybierane są nowy przykłady, określone przez zmienną *Window increment* i jeżeli są one błędnie klasyfikowane następuje przebudowa drzewa. Ponieważ drzewo jest budowane w sposób losowy, więc aby skompensować losowość budowane jest więcej niz jedno drzewo i wybierane jest najlepsze. Liczba generowanych drezw określana jest przez zmienną *Trials*.

b. Dla zbioru CRX wygenerowano dwa drzewa – jedno metodą *info gain*, a drugie *info gain* z opcją *windowing*. Przedstawione wyniki zostały wygenerowane dla parametrów: $Initialwindowsize = 98; Windowincrement = 19 lTrails = 20$. Pierwsze dwa parametry zostaly automgicznie dobrane przez program i mimo, iż próbowano zmieniać je na inne, dały one najlepszy wynik. Z poniższych zrzutów ekranu widać, iż użycie opcji *windowing* zmniejszyło liczbę pomyłek drzewa na zbiorze testowym o prawie 25%, z 20.5% do 15.5.

Zadanie 7. Generowanie krzywej uczenia

a. Do przygotowania wykresów przygotowano zbiory uczące zawierające od 50 do 300 elementów z krokiem co 25 elementów.



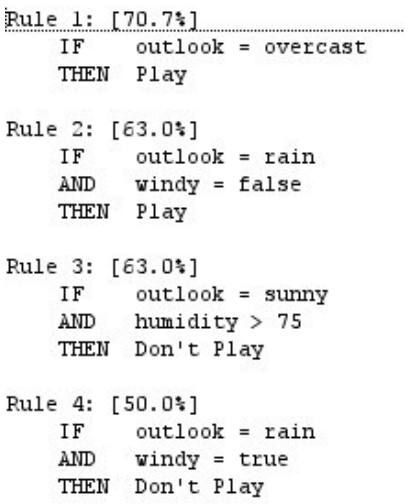


b. Patrząc na wykresy widać, iż duży zbiór uczący prowadzi do przeuczenia drzewa i spadku jego wydajności. Drugą obserwacją jest fakt, iż jakość drzewa maleje z jego rozmiarem, co również jest objawem przeuczenia.

Zadanie 8. Konkurs – uzyskać jak najwyższą trafność klasyfikowania ze zbioru GERMAN w eksperymencie 10-fold-cv

Zadanie 9. Metoda pośrednia generowania reguł (C4.5rules)

a. wygenerować reguły dla zbioru GOLF za pomocą programu C4.5 for Windows:



Rysunek 10: Reguły wygenerowane dla drzewa dla zbioru GOLF dla domyślnych parametrów

b. Wśród wygenerowanych reguł brak reguły odpowiadającej ścieżce ($outlook = sunny, humidity <= 75$). Sytuacja taka wynika z istnienia reguły domyślnej, która jest stosowana w przypadku niedopasowania wariantu do żadnej z wygenerowanych reguł. W opisywanym przypadku domyślna reguła ma postać $Default\_class : Play$.

Zadanie 10. Porównanie klasyfikowania za pomocą drzew decyzyjnych i reguł decyzyjnych (C4.5rules)

a. Przeprowadzono pod jednym teście 10-fold CV dla następujących zbiorów: CRX,

–	**Trafność klasyfikacji**		**Rozmiar**	
Zbiór	**Drzewo**	**Reguły**	**Drzewo**	**Reguły**
CRX	16.3%	13.9%	43.3	6.7
GERMAN	31.5%	30.1%	63	17.7
MONK2	41.4%	34.9%	21.7	8.6
VOTE	5%	5.7%	6.4	4.2