Wybrane zagadnienia uczenia maszynowego

Zastosowania Informatyki w Informatyce – W2

Krzysztof Krawiec

Przygotowane na podstawie

1. T. Mitchell, *Machine Learning*
2. S.J. Russel, P. Norvig, *Artificial Intelligence – A modern approach*
3. P. Cichosz, *Systemy uczące się*

Plan

1. Wprowadzenie do UM
   1. paradygmat uczenia sie z przykladow
   2. problemy klasyfikacji i regresji
2. Klasyfikator minimalnoodlegloscIowy
3. Drzewa decyzyjne
4. Rozwinięcia

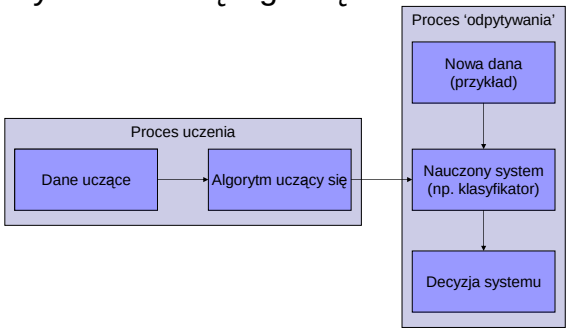
Definicja zadania uczenia

- Uczenie = automatyczne modyfikowanie (się) systemu uczącego w celu polepszania skuteczności
- Realizowane przez pozyskiwanie wiedzy z *danych uczących*

Pojęcia podstawowe

- Dane uczące (*training data*):
  - dane z których system uczący się uczy się, pozyskując wiedzę
- System uczący się (*learning/induction algorithm*)
  - algorytm pozyskujący wiedzę z danych uczących
- Klasyfikator (*classifier*)
  - ostateczna reprezentacja wiedzy wygenerowana przez system uczący się

Typowy ‘przypadek użycia’ systemu uczącego się



W ramach tego wykładu ograniczymy się do:

- Uczenie się z przykładów (*learning from examples*): dane uczące to przykłady prawidłowych decyzji podjętych w przeszłości
  - (Uczenie się z innych danych niż przykłady możliwe, ale rzadkie w praktyce)
- Reprezentacja przykładów w postaci par atrybut-wartość (*attribute-value*)
  - (Inne reprezentacje możliwe, np. teksty)

Problem gry w tenisa/golfa

- Cel: Mając dane o warunkach pogodowych, podejmij decyzję czy grać w tenisa czy też nie
- Możliwe sposoby osiągnięcia celu:
  1. Ręczna konstrukcja 'algorytmu' (reguły decyzyjnej) => system ekspercki (poza zakresem tego wykładu)
  2. Automatyczne pozyskanie wiedzy z danych => uczenie maszynowe

Problem gry w tenisa

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Analogie w terminologii statystycznej

- Przykłady – obserwacje
- Atrybuty – zmienne niezależne
- Zmienna decyzyjna – zmienna zależna

Skale atrybutów

- Nominalna
  - szczególny przypadek: atrybut binarny
- Porządkowa
- Metryczne:
  - Przedziałowa
  - Ilorazowa

Odmiany zadania uczenia się z przykładów

- Atrybut decyzyjny (zmienna zależna) dyskretny -> klasyfikacja (*classification*)
- Atrybut decyzyjny ciągły -> regresja (*regression*)
- W ramach przedmiotu skupimy się na klasyfikacji

Motywacje

- Brak wiedzy o badanym zjawisku
  - Np. robot poruszający się w nieznanym wcześniej środowisku
- ‘Lenistwo’ projektanta systemu
  - Czasami łatwiej nauczyć system niż konstruować go ręcznie od zera
- Duża liczba atrybutów
- Duża liczba przykładów

Reprezentacje wiedzy

... stosowane w systemach uczących się:

- Wybrane przykłady uczące
- Drzewa decyzyjne
- Reguły decyzyjne
- Sieci neuronowe
- Rozkłady prawdopodobieństw
- …

Cele uczenia

1. Skonstruuj możliwie prostą hipotezę możliwie dobrze opisującą (zgodną z) przykładami uczącymi
2. Skonstruuj model wyjaśniający obserwowane zjawisko

Który system/algorytm uczący się jest lepszy?

Miary skuteczności systemów uczących się:

- Trafność/błąd klasyfikowania: procent poprawnie/niepoprawnie zaklasyfikowanych przykładów
- Czułość/specyficzność
- Mniej istotne:
  - czasochłonność procesu uczenia
  - czasochłonność procesu testowania

Pożądane właściwości

Pożądane właściwości systemów uczących się:

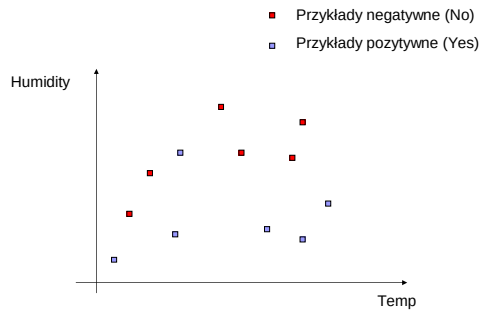
- Wysoka trafność klasyfikowania
  - w tym na nowych przykładach, czyli => zdolność uogólniania
- Odporność na szumy (na atr. warunkowych i decyzyjnym)
- Szybkość działania:
  - Szybkość uczenia
  - Szybkość odpytywania

Reprezentacja graficzna

Chwilowo załóżmy że:

1. Atrybuty Temperature i Humidity są ciągłe
2. Inne atrybuty są nieistotne

Reprezentacja graficzna



Klasyfikatory minimalnoodległościowe

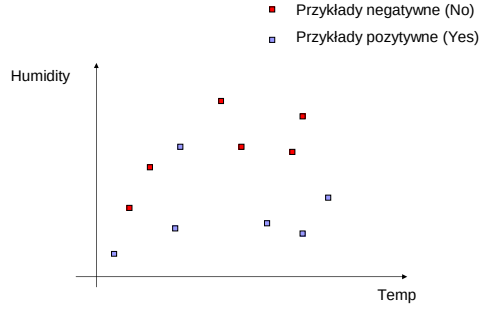
Idea

- Klasyfikuj nowe przykłady na podstawie ich podobieństwa do przykładów uczących
- Uczenie: Zapamiętaj *wszystkie* przykłady uczące
- Odpytywanie: Dla nowego przykładu X:
  - Znajdź przykład uczący Y najbardziej *podobny* do X
  - Zaklasyfikuj X do tej samej klasy do której należy Y
- Algorytm najbliższego sąsiada (*nearest neighbour, NN*)

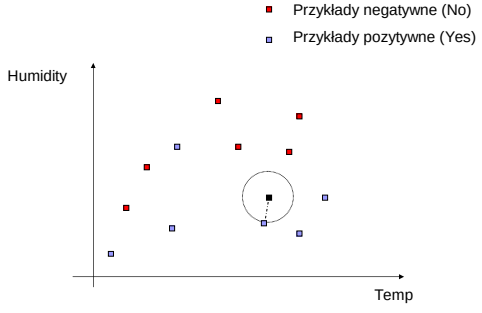
Funkcja podobieństwa

- Zazwyczaj: podobieństwo = 1/odległość (np. odległość Euklidesowa)
- Im mniejsza odległość, tym większe podobieństwo

Działanie klasyfikatora NN



Działanie klasyfikatora NN



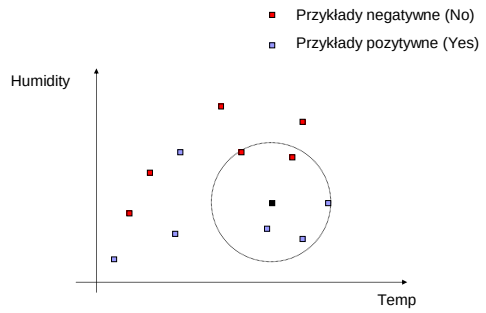
Cechy klasyfikatora NN

- Zalety:
  - Prostota
  - Bardzo szybki proces uczenia
- Wady:
  - Duże zapotrzebowanie na pamięć
  - Powolne odpytywanie
  - Wiedza = przykłady uczące (brak reprezentacji wiedzy)
  - Wszystkie atrybuty są tak samo istotne
  - Znaczna podatność na przeuczenie

Rozszerzenia NN

- kNN: wykorzystaj k>1 najbliższych sąsiadów do zaklasyfikowania nowego przykładu
- k najbliższych sąsiadów przeprowadza głosowanie (większościowe) nad przynależnością nowego przykładu

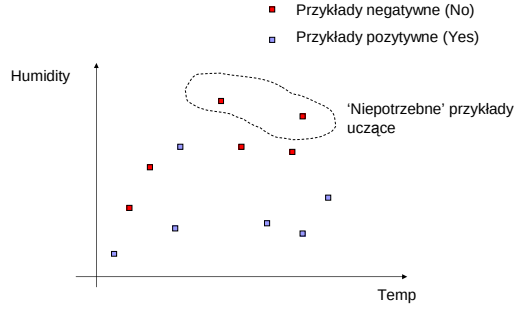
Klasyfikator kNN, k=5



Inne rozszerzenia

- Specjalne definicje odległości dla atrybutów dyskretnych/nominalnych
- Metody zapamiętujące tylko *niektóre* przykłady uczące (*instance-based learning*, IBL)
  - Krytyczne jest zapamiętanie przykładów uczących leżących na granicach klas decyzyjnych

Idea IBL2



Drzewa decyzyjne

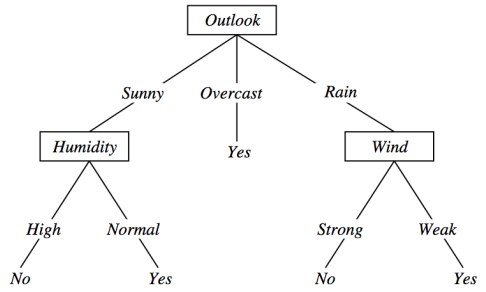
Testowanie wartości atrybutów

Elementarną operacją wykorzystywaną w drzewach decyzyjnych jest testowanie wartości pojedynczego atrybutu, np.

- Outlook = Sunny ?
- Temperature > 24 ?

Problem gry w tenisa

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No



Cechy drzew

Cechy drzew decyzyjnych jako formy reprezentacji wiedzy:

- Węzły odpowiadają testom na atrybutach
- Krawędzie odpowiadają wartościom atrybutów
- Każdy liść ma przypisaną etykietę klasy decyzyjnej

Które drzewo jest najlepsze?

- Hipotetycznie można by wygenerować wszystkie możliwe drzewa i testować czy są zgodne (spójne) ze zbiorem przykładów uczących
- Problem: jest ich bardzo dużo (wykładniczo względem liczby atrybutów, 2^{2^n} dla n atrybutów binarnych)
- Potrzeba innego algorytmu (heurystyki)

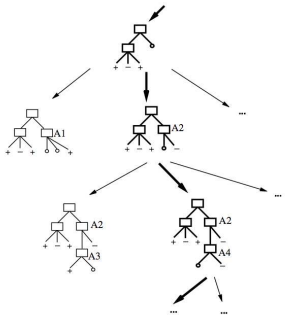
TDIDT

Top-Down Induction of Decision Trees

Dla bieżącego węzła N:

1. Wybierz *najlepszy** atrybut A dla N
2. Dla każdej wartości A, utwórz nowy węzeł potomny
3. Skieruj przykłady do węzłów potomnych (stosownie do wartości A)
4. Powtarzaj powyższe kroki dla wszystkich węzłów potomnych, aż do uzyskania idealnie ‘czystych’ węzłów

Przestrzeń hipotez (drzew)



Właściwości TDIDT

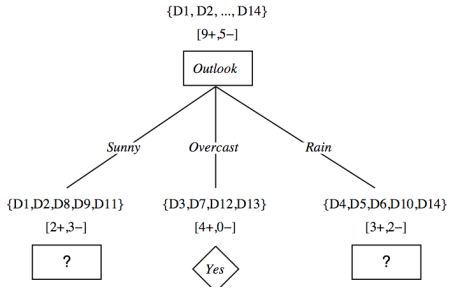
- Rekurencyjna procedura
- ‘Zachłanny’ charakter => zbudowane drzewo nie musi być optymalne (heurystyka)

Co to znaczy ‘*najlepszy*’ atrybut?

Idea:

1. Sprawdź dla każdego atrybutu A, co by było gdyby użyć go w bieżącym węźle.
2. Dla każdego takiego scenariusza oceń *jakość* wygenerowanego poddrzewa
3. Wybierz/zaakceptuj poddrzewo o najwyższej jakości

Jak zdefiniować jakość poddrzewa?

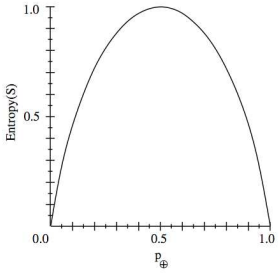


Entropia

- Entropia (zawartość informacyjna, *information content*): miara oceniająca zbiór przykładów pod kątem ‘czystości’ (jednolitości przynależności do klas decyzyjnych)
- Dla dwóch klas decyzyjnych (pozytywna, negatywna):

$$I(\frac{p}{p+n},\frac{n}{p+n}) = -\frac{p}{p+n}\log_2\frac{p}{p+n} - \frac{n}{p+n}\log_2\frac{n}{p+n}$$

Entropia



Information gain

- Entropia warunkowa: entropia po podziale zbioru przykładów przy pomocy atrybutu A (załóżmy że A przyjmuje *v* możliwych wartości):

$$EntropiaWarunkowa(A) = \sum_{i=1}^{v} \frac{p_i + n_i}{p + n} I\left(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i}\right)$$

- Zysk informacyjny (*Information Gain*): redukcja entropii przy wykorzystaniu danego atrybutu:

$$IG(A) = I - EntropiaWarunkowa(A)$$

Zalety drzew decyzyjnych

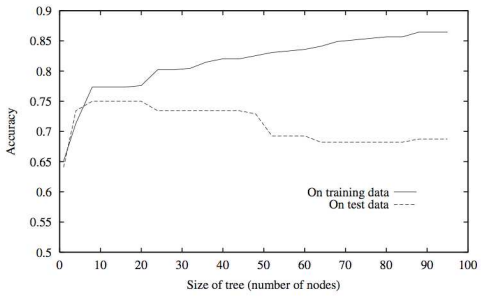
- Szybkie generowanie klasyfikatora
- Szybkie klasyfikowanie (‘odpytywanie’) klasyfikatora
- Czytelność

Problem przeuczenia

- Indukowanie wiedzy z niekompletnych danych uczących pociąga za sobą w sposób nieunikniony ryzyko *przeuczenia*.
- Formalnie: przeuczenie zachodzi gdy:

  Trafność(zb_uczący) > Trafność(zb_testujący)
- Unikanie przeuczenia to jeden z podstawowych problemów w uczeniu maszynowym.

Przeuczenie drzewa decyzyjnego



Jak uniknąć przeuczenia?

Obserwacja:

- Im dłuższa hipoteza (większa liczba warunków elementarnych), tym większe ryzyko przeuczenia
- Brzytwa Ockhama: preferuj najprostszą hipotezę która pasuje do danych (wyjaśnia dane)

Dwie możliwości:

- Zatrzymywać budowanie drzewa gdy podział na węzłach stają się niestotne statystycznie => *preprunning*
- Budować pełne drzewo, a następnie je upraszczać => *postprunning*

Przykłady warunków preprunningu

Nie dziel dalej węzła N jeżeli:

- liczba przykładów w N jest ‘niewarygodnie’ mała
- zysk na kryterium jakości jest mało satysfakcjonujący

Które drzewo jest najlepsze?

Miary oceny jakości drzewa:

- Trafność na zbiorze uczącym
- Trafność na osobnym podzbiorze walidującym
- Trafność + rozmiar drzewa

Algorytmy

- ID3
- C4.5 = ID3 + obsługa atrybutów ciągłych + inne metody upraszczania drzewa
- CART

Rozszerzenia

- Obsługa atrybutów ciągłych
- Binaryzacja drzew (zwłaszcza w obecności atrybutów nominalnych o licznych dziedzinach)
- Wyrafinowane algorytmy upraszczania drzew
- Konwertowanie drzew na reguły decyzyjne
- Obsługa wartości brakujących (*missing values*)
- Uwzględnianie kosztów atrybutów

Uzupełnienia

Zastosowania

- Diagnostyka medyczna
- Diagnostyka techniczna
- Sterowanie robotami
- Finanse (np. klasyfikacja wniosków kredytowych, wniosków o karty kredytowe)
- Internet: filtry antyspamowe
- Wykrywanie włamań do systemów informatycznych
- Rozpoznawanie obrazów (np. klasyfikacja obiektów astronimocznych)
- Zastosowania wojskowe
- …

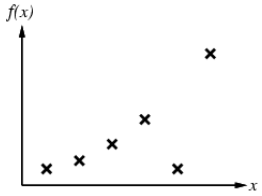
Przykład: Problem medyczny

- Cięcie cesarskie
- Drzewo wyindukowane z opisów 1000 pacjentek
- Przykłady negatywne: zastosowanie cięcia

```
[833+,167-] .83+ .17-
Fetal_Presentation = 1: [822+,116-] .88+ .12-
| Previous_Csection = 0: [767+,81-] .90+ .10-
| | Primiparous = 0: [399+,13-] .97+ .03-
| | Primiparous = 1: [368+,68-] .84+ .16-
| | | Fetal_Distress = 0: [334+,47-] .88+ .12-
| | | | Birth_Weight < 3349: [201+,10.6-] .95+ .05
| | | | Birth_Weight >= 3349: [133+,36.4-] .78+ .2
| | | Fetal_Distress = 1: [34+,21-] .62+ .38-
| Previous_Csection = 1: [55+,35-] .61+ .39-
Fetal_Presentation = 2: [3+,29-] .11+ .89-
Fetal_Presentation = 3: [8+,22-] .27+ .73-
```

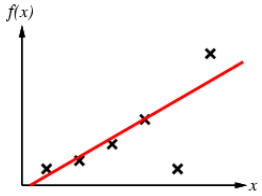
Zadanie regresji

- Tym razem celem jest przewidywanie (predykcja) wartości *ciągłej* zmiennej f(x) na podstawie zmiennych niezależnych (poniżej: x)



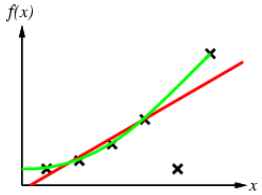
Zadanie regresji

- Prosty model (hipoteza) dokonujący w przybliżeniu poprawnej predykcji



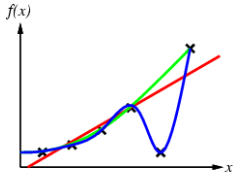
Zadanie regresji

- Bardziej wyrafinowany model: zmniejszenie błędu na wielu przykładach kosztem polepszenia na jednym przykładzie



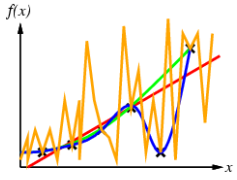
Zadanie regresji

- Model perfekcyjnie dopasowany do danych uczących.
- Czy to jest najlepszy model?



Zadanie regresji

- Odpowiedź: niekoniecznie, rzeczywiste zjawisko które wygenerowało obserwacje mogło wyglądać tak:



Inne warianty zadania uczenia

- Uczenie opisywane tutaj to uczenie nadzorowane (*supervised learning*): dane uczące zawierają poprawną decyzję dla każdego przykładu
- Inne warianty:
  - Uczenie nienadzorowane (*unsupervised learning*): decyzje nieznane => m.in. analiza skupień
  - Uczenie ze wzmacnianiem (*reinforcement learning*)