Inteligencja obliczeniowa - Uczenie maszynowe: klasyfikacja

Grzegorz Madejski

Uczenie maszynowe

Uczenie maszynowe

Uczenie maszynowe (ang. machine learning) – obszar sztucznej inteligencji poświęcony algorytmom, które poprawiają się automatycznie poprzez doświadczenie.

Nadzorowane vs nienadzorowane

Uczenie maszynowe nadzorowane

Uczenie maszynowe nadzorowane (ang. Supervised Machine Learning): Uczymy algorytm na danych, dla których istnieją odpowiedzi. Trening nastepuje tak długo, aż algorytm przestaje udzielać złych odpoiwiedzi. Działamy jak nadzorca / nauczyciel. Wyuczony algorytm może zgadywac odpowiedzi dla danych, które nie mają odpowiedzi. Przykłady: klasyfikacja, regresja.

Uczenie maszynowe nienadzorowane

Uczenie maszynowe nienadzorowane (ang. Unsupervised Machine Learning): Uczenie maszynowe, w którym nie nadzorujemy uczenia modelu, bo (zwykle) nie ma odpowiedzi, na których może się on uczyć. Zamiast tego, algorytm sam musi odkrywać w danych wzory lub wydobywać informacje. Przykłady: grupowanie, reguły asocjacyjne, alg. genetyczny.

Uczenie maszynowe

Uczenie maszynowe, zwłaszcza nadzorowane, można podzielić na dwa etapy:

- Trening modelu, algorytmu.
- Ewaluacja (ocenianie, testowanie) algorytmu.

Gdy algorytmu ma złą ocenę, zmieniamy go na inny lub modyfikujemy lub dotrenowujemy. Gdy ma dobrą ocenę, możemy go wykorzystać do przewidzianego zadania.

Klasyfikacja

Klasyfikacja

Klasyfikacja (ang. classification) – zadanie przyporządkowywania danej próbce (obserwacji) jednej z kategorii (klas). Jest to zadanie uczenia maszynowego nadzorowanego.

- Próbka: wyniki badań krwi pacjenta. Klasa: zdrowy (test-negatywny), chory (test-pozytywny).
- Próbka: oceny studenta w ostatnim roku akademickim. Klasa: stypendium-wysokie, stypendium-niskie, brak-stypendium.

Klasyfikator

Klasyfikator

Klasyfikator (ang. classifier) – algorytm służący do klasyfikowania obserwacji.

Popularne algorytmy klasyfikujące:

- drzewo decyzyjne (decision tree)
- k-najbliższych sąsiadów (k-nearest neighbors)
- naiwny bayesowski (naive Bayes)
- las losowy decyzyjny (random forest)
- sieć neuronowa (neural network)
- maszyna wektorów nośnych (support-vector machine)

Zbiór treningowy i testowy

Leukocyty [G/l]	Limfocyty (G/l)	Monocyty [G/l]	Choroba
5,1	3,2	0,3	neg
10	4,1	0,2	neg
3,2	6,5	0,7	poz

- Klasyfikatory uczą się na zbiorach danych (datasets).
- Klasyfikator musi też zostać oceniony. Czy działa efektywnie?
- Zbiór danych dzielimy na dwie części: zbiór treningowy i zbiór testowy.
- *Zbior treningowy* służy do uczenia się klasyfikatora. Jest z reguły większy niż testowy.
- *Zbior testowy* służy do oceniania (*ewaluacji*) klasyfikatora. Klasyfikator udziela odpowiedzi dla danej próbki i porównujemy ją z prawdziwą odpowiedzią ze zbioru testowego.

Ewaluacja klasyfikatora

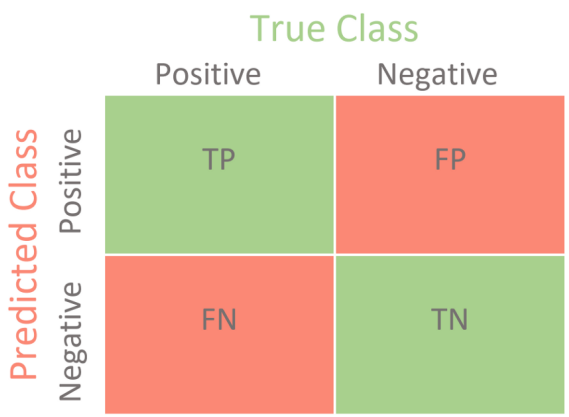
- Jest wiele sposobów i miar na ewaluację klasyfikatora.
- Podstawowa miara to *dokładność* klasyfikatora (ang. accuracy) liczona według wzoru:

$$\text{Dokładność} = \frac{\text{Liczba próbek dobrze sklasyfikowanych w zbiorze testowym}}{\text{Liczba wszystkich próbek w zbiorze testowym}}$$

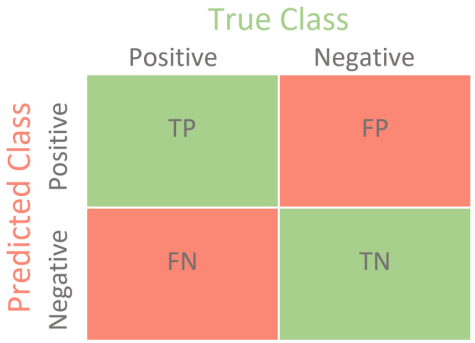
- Dobre klasyfikatory osiągają z reguły bardzo wysokie (min. 90%) dokładności.

Ewaluacja klasyfikatora

- Często oprócz dokładności dla klasyfikatora podajemy *macierz błędu* (ang. confusion matrix), która zestawia jakie błędy zostały popełniane.
- Przykładowo: osobę zdrową klasyfikator może ocenić jako chorą, a chorą jako zdrową.
- Jeśli klasyfikujemy do dwóch kategorii (tak/nie, dobry/zły, chory/zdrowy) to w macierzy znajdą się 4 pola: true negative, true positive, false negative, false positive.



Ewaluacja klasyfikatora



- *True negative* - liczba próbek sklasyfikowanych jako negatywne, które w rzeczywistości też były negatywne (np. zgadliśmy, że zdrowe to zdrowe).
- *True positive* - liczba próbek sklasyfikowanych jako pozytywne, które w rzeczywistości też były pozytywne (np. zgadliśmy, że chore to chore).
- *False positive* - liczba próbek sklasyfikowanych jako pozytywne, które w rzeczywistości były negatywne (np. osobę zdrową sklasyfikowaliśmy jako chorą). Też: błąd typu I, "fałszywy alarm", "przeszacowanie".
- *False negative* - liczba próbek sklasyfikowanych jako negatywne, które w rzeczywistości były pozytywne (np. osobę chorą sklasyfikowaliśmy jako zdrową). Też: błąd typu II, "chybienie", "niedoszacowanie".

Ewaluacja klasyfikatora

Biorąc pod uwagę powyższe oznaczenia, mamy miary:

- *Dokładność*: $Acc = \frac{TP+TN}{P+N}$ (P = próbki z odpowiedzą tak, N = próbki z odpowiedzią nie)
- *Czułość* (ang. sensitivity, true positive rate), wysoka wykrywalność odpowiedzi "tak": $TPR = \frac{TP}{P} = \frac{TP}{TP+FN}$
- *Swoistość* (ang. specificity, true negative rate), wysoka wykrywalność odpowiedzi "nie": $TNR = \frac{TN}{N} = \frac{TN}{TN+FP}$
- *Szansa chybienia* (ang. miss rate, false negative rate), błąd wykrywalności odpowiedzi "tak": $FNR = \frac{FN}{P} = \frac{FN}{TP+FN} = 1 - TPR$
- *Szansa fałszywego alarmu* (ang. fall-out, false positive rate), błąd wykrywalności odpowiedzi "nie": $FPR = \frac{FP}{N} = \frac{FP}{TN+FP} = 1 - TNR$

Drzewa decyzyjne

Omawiamy teraz:

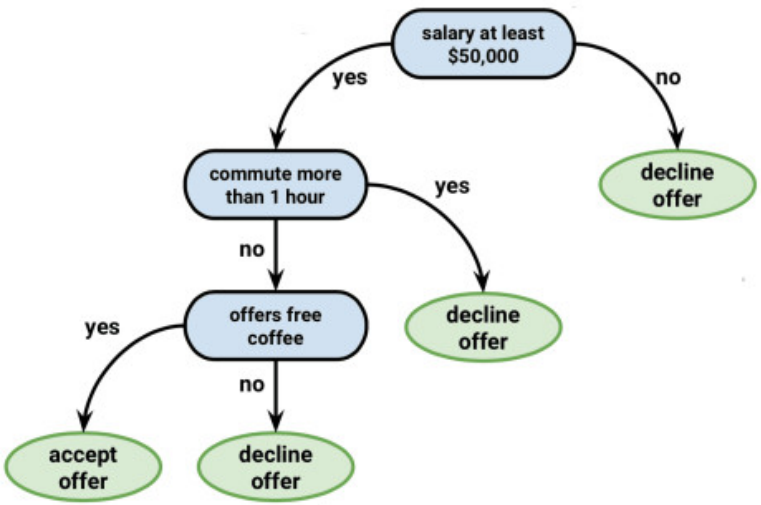
- *Drzewo decyzyjne*
- *k* najbliższych sąsiadów
- Naiwny Bayes

Drzewa decyzyjne

Drzewo decyzyjne

Drzewo decyzyjne (ang. decision tree) to model klasyfikujący, który podejmuje decyzję na podstawie zestawu pytań dotyczących parametrów i ich wartości.

W korzeniu i innych węzłach wewnętrznych rozpatrujemy parametry/pytania. Na krawędziach wybieramy odpowiedzi. W liściach są klasy. Przykład:



Drzewa decyzyjne

Drzewo decyzyjne to już wytrenowany model. Sam proces tworzenia modelu można wykonać różnymi algorytmami np.:

- ID3 (Iterative Dichotomiser 3)
- C4.5 (poprawiony ID3)
- CART (skrót od: Classification And Regression Tree)

Drzewa decyzyjne: jak działa ID3?

Spróbujmy zbudować drzewo decyzyjne zgadujące, czy warto grać w golfa w konkretnych warunkach pogodowych. Drzewo zbudujemy algorytmem ID3, który radzi sobie jedynie z danymi dyskretnymi.

Outlook	Temp.	Humidity	Wind	Decision
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Drzewa decyzyjne: jak działa ID3?

Entropia to miara ilości informacji w danej zmiennej (rozrzut, niepewność informacji). Przykład poniżej: 10 osób zdaje prawo jazdy. Sposób w jaki rozdzielą się informacje zdane/niezdane zmienia entropię.

yes	no	Wzór na entropię	Obliczona entropia
5	5	E(5,5)= -5/10*LOG(5/10)-5/10*LOG(5/10)=	0,301
4	6	E(4,6)= -4/10*LOG(4/10)-6/10*LOG(6/10)=	0,292
3	7	E(3,7)= -3/10*LOG(3/10)-7/10*LOG(7/10)=	0,265
2	8	E(2,8)= -2/10*LOG(2/10)-8/10*LOG(8/10)=	0,217
1	9	E(1,9)= -1/10*LOG(1/10)-9/10*LOG(9/10)=	0,141
0	10	E(0,10)= -0/10*LOG(0/10)-10/10*LOG(10/10)=	0,000

Drzewa decyzyjne: jak działa ID3?

Entropia warunkowa może badać, czy jakiś inny czynnik (zmienna) wpływa na klasę (zdawalność egzaminu prawa jazdy). Weźmy płeć. Jeśli zysk informacji (różnica między entropią, a entropią warunkową uwzględniającą płeć) jest duży, to taka zmienna jest wartościowa z punktu widzenia budowania drzew decyzyjnych.

kobieta	kobieta	mężczyzna	mężczyzna	Entropia	Entropia	Zysk informacji: E(5,5)-Wynik
yes	no	yes	no			
4	4	1	1	8/10*E(4,4)+2/10*E(1,1)	0,301	0,000
4	1	4	1	5/10*E(4,1)+5/10*E(4,1)	0,217	0,084
4	1	1	4	5/10*E(4,1)+5/10*E(1,4)	0,217	0,084
3	3	2	2	6/10*E(3,3)+4/10*E(2,2)	0,301	0,000
3	2	2	3	5/10*E(3,2)+5/10*E(2,3)	0,292	0,009
3	2	3	2	5/10*E(3,2)+5/10*E(3,2)	0,292	0,009
5	0	4	1	5/10*E(5,0)+5/10*E(4,1)	0,109	0,192
5	0	3	2	5/10*E(5,0)+5/10*E(3,2)	0,146	0,155
5	0	5	0	5/10*E(5,0)+5/10*E(5,0)	0,000	0,301

Drzewa decyzyjne: jak działa ID3?

Jak działa algorytm ID3?

1. Policz *entropię* dla kolumny z klasą: $E(class)$.
2. Policz *entropie warunkowe* $E(class|column)$ dla wszsytkich kolumn.
3. Policz *zysk informacji* (ang. *information gain*) $IG(class|column)$ dla wszystkich kolumn.
4. Wybierz kolumnę X z najwięszym zyskiem informacji. Umieść ją w korzeniu drzewa decyzyjnego. Z korzenia prowdzą krawędzie oznaczone wszystkimi możliwymi wartościami X. Idąc po krawędzi oznaczonej daną wartością X, przechodzimy do podzbioru danych, w którym są próbki tylko z daną wartością X. Nastepnie kolumnę X usuwamy z tego podzbioru.
5. (Rekurencja) Dla wszystkich nowo powstałych wierzchołków zawierających nowe pomniejszone zbiory danych: zacznij algorytm od punktu 1. Jesli wszystkie odpowiedzi w klasie są takie same, zwróć wierzchołek jako liść z odpowiedzią: tą wartością klasy.

Drzewa decyzyjne: jak działa ID3?

Krok 1: entropia dla klasy.

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Play Golf	
Yes	No
9	5

Entropy(PlayGolf) = Entropy (5,9)
= Entropy (0.36, 0.64)
= - (0.36 $\log_2$ 0.36) - (0.64 $\log_2$ 0.64)
= 0.94

Źródło:

https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html

Drzewa decyzyjne: jak działa ID3?

Krok 2: entropie warunkowe. Tutaj entropia $E(\text{Class}|\text{Outlook})$.
Trzeba policzyć jeszcze dla Temp, Humidity i Wind.

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

E(PlayGolf, Outlook) = **P**(Sunny)***E**(3,2) + **P**(Overcast)***E**(4,0) + **P**(Rainy)***E**(2,3)

= (5/14)*0.971 + (4/14)*0.0 + (5/14)*0.971

= 0.693

Źródło:

https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html

Drzewa decyzyjne: jak działa ID3?

Krok 3: Policzenie zysku informacji dla wszystkich 4 kolumn. Tutaj pokazane dla kolumny Outlook.

$$\text{Information Gain}(T,X) = \text{Entropy}(T) - \text{Entropy}(T, X)$$

$$\begin{aligned} \text{IG}(\text{PlayGolf}, \text{Outlook}) &= \text{E}(\text{PlayGolf}) - \text{E}(\text{PlayGolf}, \text{Outlook}) \\ &= 0.940 - 0.693 \\ &= 0.247 \end{aligned}$$

Źródło: https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html

Drzewa decyzyjne: jak działa ID3?

Zadanie: budowanie drzewa z ID3

Policz zysk informacji dla pozostałych kolumn (Temp, Humidity, Wind), w pierwszym przebiegu rekurecynjnym algorytmu ID3. Możesz to zrobić na kartce lub w Excelu (lub w Pythonie).

Sprawdź czy masz podobne odpowiedzi:

$$IG(Decision, Wind) = 0.048, IG(Decision, Temperature) = 0.029,$$

$$IG(Decision, Outlook) = 0.246, IG(Decision, Humidity) = 0.151$$

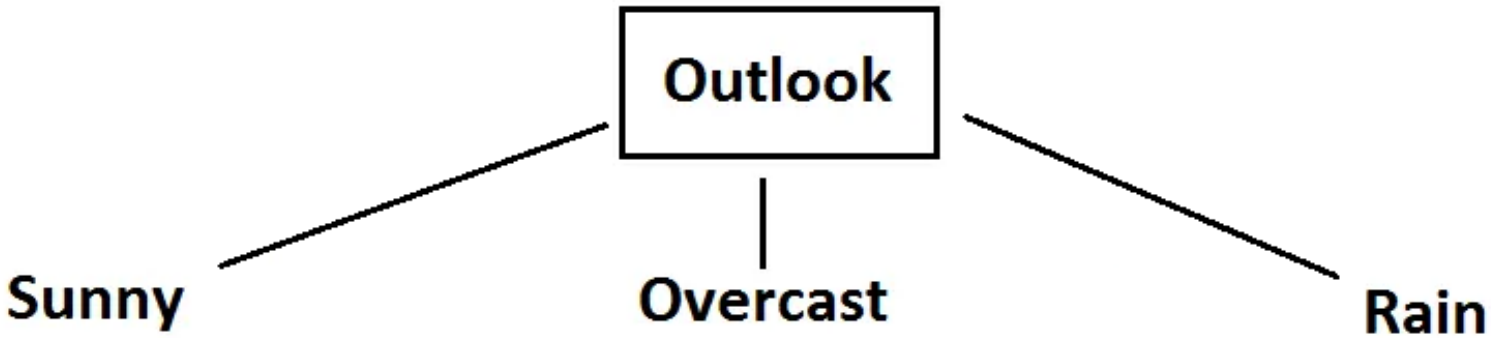
Pomocne linki: https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html,

https://sefiks.com/2017/11/20/a-step-by-step-id3-decision-tree-example/. Zbiór:

https://gist.github.com/kudaliar032/b8cf65d84b73903257ed603f6c1a2508

Drzewa decyzyjne: jak działa ID3?

Krok 4 i 5. Schodzimy rekurencją w dół. Poniżej tabelka z odfiltrowanym Outlook=Overcast. W tej pomniejszonej tablce można tę kolumnę usunąć.



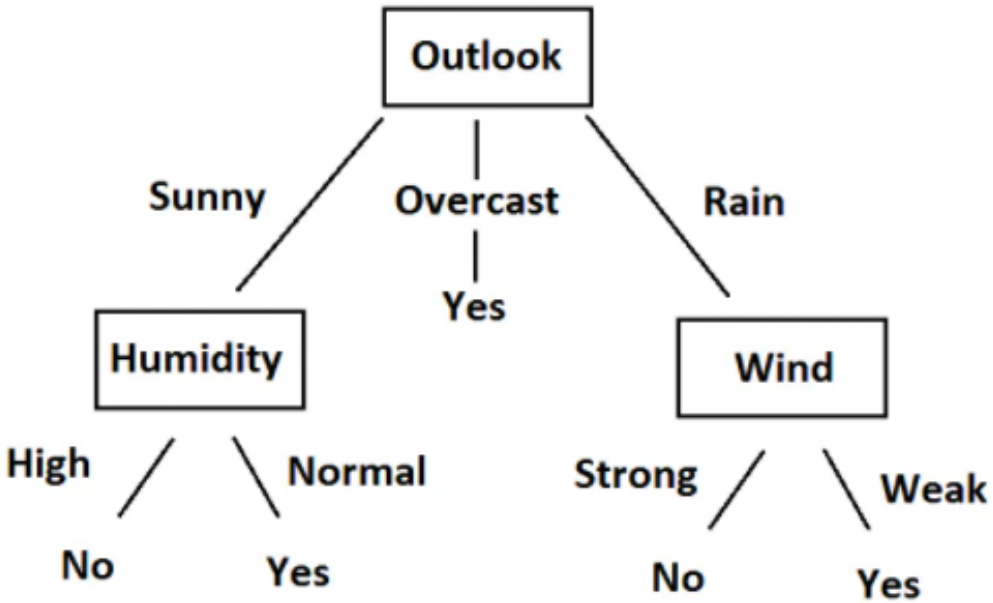
Outlook	Temp.	Humidity	Wind	Decision
Overcast	Hot	High	Weak	Yes
Overcast	Cool	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes

We continue the algorithm for all branches.

Source: https://sefiks.com/2017/11/20/a-step-by-step-id3-decision-tree-example/

Drzewa decyzyjne: jak działa ID3?

Drzewo po całym przebiegu algorytmu:



Source: https://sefiks.com/2017/11/20/a-step-by-step-id3-decision-tree-example/

k najbliższych sąsiadów

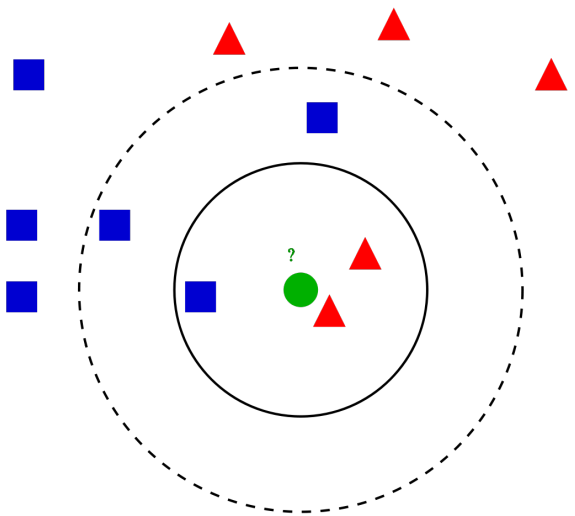
Omawiamy teraz:

- Drzewo decyzyjne
- *k najbliższych sąsiadów*
- Naiwny Bayes

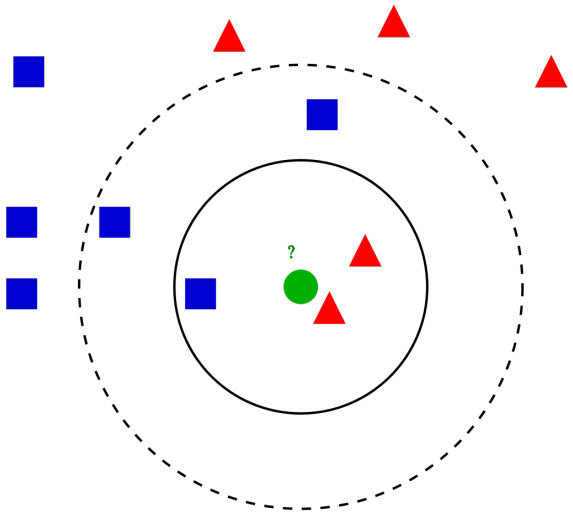
k-Najbliższych Sąsiadów (kNN)

k-Najbliższych Sąsiadów

k-najbliższych sąsiadów (ang. *k*-nearest neighbors, kNN) to klasyfikator, który przyporządkowuje danej obserwacji taką klasę, jaką ma k najbardziej podobnych do niej próbek (ze zbioru treningowego).



k-Najbliższych Sąsiadów (kNN)



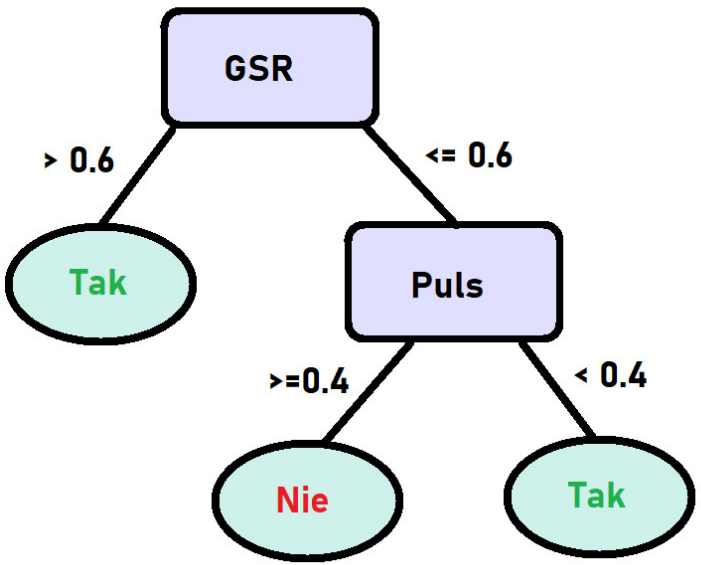
- Liczbę k ustala człowiek (1, 3, 5, ..., $\sqrt{\text{Zb.testowy}}$)
- Podobieństwo próbek można mierzyć wybraną przez siebie miarą (euklidesowa, Manhattan, cosinusowa).
- Podobieństwo próbek numerycznych może być ich odległością w przestrzeni (tak jak na rysunku powyżej)
- Podobieństwo danych nominalnych może być zero-jedynkowe (takie same = 1, różne = 0).

Wykrywacz kłamstw

Wykrywacz kłamstw

Policja zgromadziła bazę danych osób winnych zarzucanych im czynów karalnych i niewinnych, oraz ich wyników z badania wariografem - mierzono szybkość pulsu oraz przewodnictwo skóry (GSR). Ponieważ policja wraz z sądem nie miała środków pieniężnych na przeprowadzanie dochodzeń, postanowiła, że będzie wsadzać za kratki tylko na podstawie wyników wariografu. Jako eksperta od sztucznej inteligencji, poproszono Cię o stworzenie i ewaluację algorytmu klasyfikującego, który decydowałby kto jest winny, a kto nie. Dzięki temu można będzie wsadzać ludzi za kratki w trybie ekspresowym bez szukania dowodów! Na nastepnym slajdzie baza danych podzielona na zbiór testowy i treningowy (oraz drzewo decyzyjne).

Wykrywacz kłamstw



Zbiór treningowy

Puls	GSR	Winny
1	0,7	Tak
0,8	0,8	Tak
0,9	0,9	Tak
0,6	1	Tak
0,5	0,5	Tak
0,3	0,9	Tak
0,3	0,4	Nie
0,2	0	Nie
0,1	0,2	Nie
0	0,3	Nie
0,6	0,8	Nie

Zbiór testowy

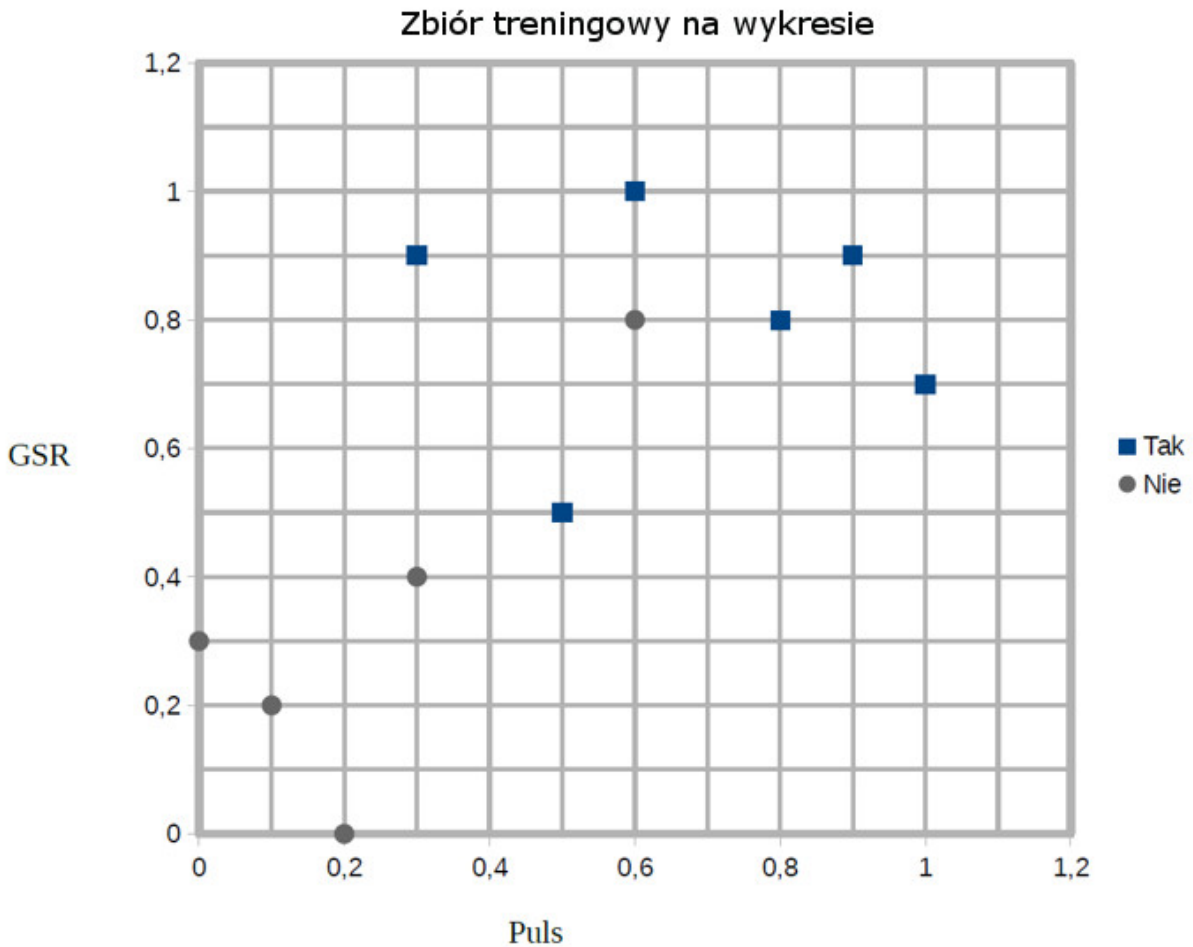
Puls	GSR	Winny
0,4	0,6	Nie
0,6	0,6	Tak
0,4	0,9	Tak
0,5	0,2	Nie
0,5	0,6	Tak

Zadanie: porównanie klasyfikatorów

Który klasyfikator działa najlepiej: 1-najbliższego sąsiada, 3-najbliższych sąsiadów czy drzewo decyzyjne z obrazka? Zewaluuj te trzy klasyfikatory, podając ich dokładność i macierz błędu.

Wykrywacz kłamstw

Wykres pomocny do 1NN i 3NN.



Naiwny Bayes

Omawiamy teraz:

- Drzewo decyzyjne
- k najbliższych sąsiadów
- *Naiwny Bayes*

Naiwny Bayes

Przypomnijmy wzór Bayesa na prawdopodobieństwo warunkowe:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

gdzie:

- $P(A|B)$ - prawodpodobieństwo, że hipoteza/odpowiedź A jest prawdziwa, pod warunkiem, że widzimy dowody/dane B (a posteriori = "na podstawie faktów")
- $P(B|A)$ - prawodpodobieństwo, częstość występowania danych B wśród próbek z odpowiedzią A
- $P(A)$ - prawodpodobieństwo, że hipoteza A jest prawdziwa (a priori = "z góry/założenia"). Wynika ze zbioru obserwacji i nie bierze pod uwagę dowodów.
- $P(B)$ - prawodpodobieństwo wystąpienia danych B (dowodów)

Naiwny Bayes

Przykład: próbujemy zdiagnozować osobę jako zdrowa lub chorą, na podstawie wyników badań. Które prawdopodobieństwo jest większe?

$$P(\text{chory}|\text{wyniki badań}) = \frac{P(\text{wyniki badań}|\text{chory}) \cdot P(\text{chory})}{P(\text{wyniki badań})}$$

$$P(\text{zdrowy}|\text{wyniki badań}) = \frac{P(\text{wyniki badań}|\text{zdrowy}) \cdot P(\text{zdrowy})}{P(\text{wyniki badań})}$$

Naiwny Bayes

Definition

Klasyfikator naiwny bayesowski, naiwny Bayes (ang. Naive Bayes Classifier) to klasyfikator bazujący na prawdopodobieństwie. Używa twierdzenia Bayesa do zgadywania klas.

$$P(class|data) = \frac{P(data|class) \cdot P(class)}{P(data)}$$

Dane są niezmienne, mianownik jest zawsze taki sam. Więc możemy go usunąć:

$$P(class|data) = P(data|class) \cdot P(class)$$

Nastepnie *naiwnie* zakładamy, że kolumny są niezależne. Wówczas prawdopodobieństwo warunkowe we wzorze można rozbić:

$$P(class|data) = P(data1|class) \cdot P(data2|class) \cdot ... \cdot P(datak|class) \cdot P(class)$$

Naiwny Bayes

Przykładowy zbiór danych osób chcących kupić komputer.

age	income	student	credit.rating	buys
31..40	high	no	fair	yes
>40	medium	no	fair	yes
>40	high	yes	excellent	yes
>40	low	yes	excellent	no
31..40	low	no	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	no

Czy osoba X będzie chciała kupić komputer?

>40	medium	no	excellent	???

Naiwny Bayes

age	income	student	credit.rating	buys
31..40	high	no	fair	yes
>40	medium	no	fair	yes
>40	high	yes	excellent	yes
>40	low	yes	excellent	no
31..40	low	no	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	no

X:

>40	medium	no	excellent	???

$$P(class|data) = P(data1|class) \cdot P(data2|class) \cdot \ldots \cdot P(datak|class) \cdot P(class)$$

- Policz $P(class)$: $P(buys = yes) = 4/7$, $P(buys = no) = 3/7$
- Policz $P(data|class)$ dla wszystkich kolumn. Dla obu wartości klas rozpatrujemy dane z X:
  $P(age > 40|buys = yes) = 2/4$, $P(age > 40|buys = no) = 1/3$,
  $P(income = medium|buys = yes) = 1/4$, $P(income = medium|buys = no) = 1/3$,
  $P(student = no|buys = yes) = 3/4$, $P(student = no|buys = no) = 1/3$,
  $P(credit.rating = excellent|buys = yes) = 2/4$, $P(credit.rating = excellent|buys = no) = 1/3$
- Mnożymy: $P(X|buys = yes) = (2/4) * (1/4) * (3/4) * (2/4) = 3/64$
  $P(X|buys = no) = (1/3) * (1/3) * (1/3) * (1/3) = 1/81$
  $P(buys = yes|X) = P(X|buys = yes) * P(buys = yes) = (3/64) * (4/7) = 0.02679$
  $P(buys = no|X) = P(X|buys = no) * P(buys = no) = (1/81) * (3/7) = 0.00529$
- Liczba 0.2679 jest większa, więc odpowiedź to "yes".

Naiwny Bayes

Zbiór danych:

age	income	student	credit.rating	buys
31..40	high	no	fair	yes
>40	medium	no	fair	yes
>40	high	yes	excellent	yes
>40	low	yes	excellent	no
31..40	low	no	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	no

Zadanie: naiwny Bayes

Użyj algorytmu naiwnego Bayesa do sklasyfikowania próbki Y o nastepujących danych:

>40	low	no	fair	???