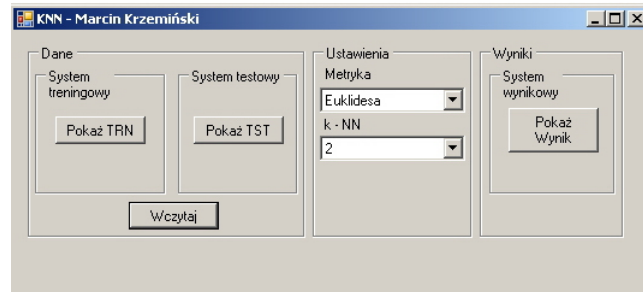


Ćwiczenie 2

Klasyfikator $k - NN$



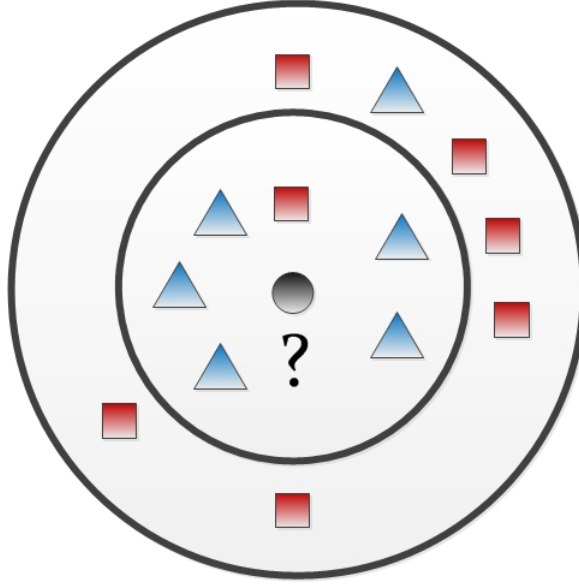
Zadanie do wykonania

- 1) Tworzymy na pulpicie katalog w formacie Imię_nazwisko, w którym umieszczamy wszystkie pliki związane z ćwiczeniem.
- 2) Czytamy teorię związaną z klasyfikacją metodą k najbliższych sąsiadów ($k - NN$), w razie problemów ze zrozumieniem, analizujemy przykłady na kartce.
- 3) Generujemy system decyzyjny treningowy i testowy za pomocą programu is_ds_generator.exe.
- 4) Otrzymany system testowy klasyfikujemy systemem treningowym metodą $2 - NN$, za pośrednictwem programu napisanego w dowolnym języku programowania. Implementujemy algorytm dla: Metryki Euklidesa, Canberra, Czebyszewa, Manhattan oraz dla Bezwzględnego współczynnika korelacji Pearsona.
- 5) W przypadku programowania w C++, ułatwieniem może być programem demonstracyjny znajdujący się na stronie <http://wmii.uwm.edu.pl/~artem> w zakładce Dydaktyka/Sztuczna Inteligencja.

Teoria do ćwiczeń z przykładami

Metody z rodziny k najbliższych sąsiadów

Przyjmując, że nie znamy figury w centrum Rys. 1 możemy przeprowadzać dedukcję na podstawie obserwacji figur stojących w jej sąsiedztwie prowadzącą do jej zdefiniowania. Przykładowo, gdy rozważamy po dwie najbliższe figury spośród trójkątów i kwadratów, widzimy, że sumaryczna odległość dwóch trójkątów od figury nieznanej jest mniejsza niż odległość pary kwadratów, stąd możemy przypuszczać, że naszą ukrytą figurą jest trójkąt. Tego typu sposób klasyfikacji nazywamy metodą k najbliższych sąsiadów, w naszym przykładzie rozważaliśmy $k=2$ w sensie wyboru po dwa obiekty z każdej dostępnej klasy obiektów.



Rysunek 1: Wizualizacja problemu klasyfikacji metodą k najbliższych sąsiadów

- Dla danego systemu testowy (X, A, c) i treningowego (Y, A, c) , gdzie X, Y to odpowiednio uniwersum obiektów testowych i treningowych, $A = (a_1, a_2, \dots, a_n)$ jest zbiorem atrybutów warunkowych, $c \in D = \{c_1, c_2, \dots, c_m\}$ jest atrybutem decyzyjnym.

Dla obiektów $x \in X, y \in Y$ postaci,

$$x = a_1(x) \ a_2(x) \ \dots \ a_n(x) \ c(x)$$

$$y = a_1(y) \ a_2(y) \ \dots \ a_n(y) \ c(y)$$

zdefiniujemy podstawowe metryki,

Metryka Manhattan przedstawia się następująco,

$$d(x, y) = \sum_{i=1}^n |a_i(x) - a_i(y)|$$

Metryka Euklidesowa szczególnym przypadkiem metryki Minkowskiego,

$$d(x, y) = \sqrt{(a_1(x) - a_1(y))^2 + (a_2(x) - a_2(y))^2 + \dots + (a_n(x) - a_n(y))^2}$$

czyli zapisując ogólnie:

$$d(x, y) = \sqrt{\sum_{i=1}^n (a_i(x) - a_i(y))^2}$$

Metryka Canberra jest postaci,

$$d(x, y) = \sum_{i=1}^n \left| \frac{a_i(x) - a_i(y)}{a_i(x) + a_i(y)} \right|$$

Metryka Czebyszewa określana jest wzorem,

$$d(x, y) = \max(|a_i(x) - a_i(y)|), \text{ dla } i = 1, 2, \dots, n$$

Bezwzględny współczynnik korelacji Pearsona może być używany w poniższy sposób,

$$d(x, y) = 1 - |r_{x,y}|$$

$$r_{x,y} = \frac{1}{n} * \sum_{i=1}^n \left(\frac{a_i(x) - \bar{x}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (a_i(x) - \bar{x})^2}} \right) \left(\frac{a_i(y) - \bar{y}}{\sqrt{\frac{1}{n} \sum_{i=1}^n (a_i(y) - \bar{y})^2}} \right)$$

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n a_i(x), \bar{y} = \frac{1}{n} \sum_{i=1}^n a_i(y)$$

Procedura algorytmu k-NN z równym uwzględnianiem klas decyzyjnych

- Wczytujemy system testowy (X, A, c) i treningowy (Y, A, c) , gdzie X, Y to odpowiednio uniwersum obiektów testowych i treningowych, $A = (a_1, a_2, \dots, a_n)$ jest zbiorem atrybutów warunkowych, $c \in D = \{c_1, c_2, \dots, c_m\}$ jest atrybutem decyzyjnym.
- Ustalamy metrykę d liczenia odległości między obiektami, oraz liczbę najbliższych sąsiadów decydujących o klasyfikacji k ,
- Klasyfikujemy wszystkie obiekty testowe za pomocą k najbliższych obiektów, każdej z klas systemu treningowego, (decyzję przekazuje klasa, której obiekty są najbliższe testowego w sensie metryki d),
- Po zakończeniu klasyfikacji, tworzymy Macierz Predykcji, zawierającą informacje o jakości klasyfikacji systemu testowego X :

Parametry mówiące o jakości przeprowadzonej klasyfikacji, które należy umieścić w raporcie klasyfikacji (w Macierzy Predykcji) są definiowane następująco:

Dla $\Lambda_{c \in D}$

$$acc_c = \frac{\text{liczba obiektów poprawnie sklasyfikowanych w klasie decyzyjnej } c}{\text{liczba obiektów chwyconych w klasie } c}$$

$$cov_c = \frac{\text{liczba obiektów chwyconych w klasie } c}{\text{liczba obiektów klasy } c}$$

$$TPR_c = \frac{x}{x + \text{liczba obiektów z pozostałych klas błędnie trafiających do klasy } c}$$

przyjmujemy, że $x = \text{liczba obiektów poprawnie sklasyfikowanych w klasie decyzyjnym } c$

Ostatecznie wyliczamy wartości globalne, które umieszczamy pod Macierzą Predykcji,

$$acc_{global} = \frac{\text{ilość obiektów poprawnie sklasyfikowanych w całym systemie TST}}{\text{ilość obiektów chwyconych w systemie TST}}$$

$$cov_{global} = \frac{\text{ilość obiektów chwyconych w całym systemie TST}}{\text{ilość obiektów systemu TST}}$$

Przykładowa klasyfikacja 2-NN Wczytujemy system testowy postaci,

Tabela 1: System Testowy (X, A, c)

	a_1	a_2	a_3	a_4	c
x_1	2	4	2	1	4
x_2	1	2	1	1	2
x_3	9	7	10	7	4
x_4	4	4	10	10	2

oraz system treningowy

Tabela 2: System Treningowy (Y, A, c)

	a_1	a_2	a_3	a_4	c
y_1	1	3	1	1	2
y_2	10	3	2	1	2
y_3	2	3	1	1	2
y_4	10	9	7	1	4
y_5	3	5	2	2	4
y_6	2	3	1	1	4

Ustalmy $k=2$ i d jako metrykę Euklidesa

Metryka Euklidesa działa następująco, dla obiektów

$$x = a_1(x) \ a_2(x) \ \dots \ a_n(x) \ c(x)$$

$$y = a_1(y) \ a_2(y) \ \dots \ a_n(y) \ c(y)$$

$$d(x, y) = \sqrt{(a_1(x) - a_1(y))^2 + (a_2(x) - a_2(y))^2 + \dots + (a_n(x) - a_n(y))^2}$$

czyli zapisując ogólnie:

$$d(x, y) = \sqrt{\sum_{i=1}^n (a_i(x) - a_i(y))^2}$$

Przechodzimy do klasyfikacji obiektów testowych:

Dla x_1 2 4 2 1 4

$$d(x_1, y_1) = \sqrt{(2-1)^2 + (4-3)^2 + (2-1)^2 + (1-1)^2} = \sqrt{3}$$

$$d(x_1, y_2) = \sqrt{65}$$

$$d(x_1, y_3) = \sqrt{2}$$

$$d(x_1, y_4) = \sqrt{114}$$

$$d(x_1, y_5) = \sqrt{3}$$

$$d(x_1, y_6) = \sqrt{2}$$

Dwóch najbliższych sąsiadów obiektu testowego x_1 w klasie decyzyjnej 2 to y_3, y_1

Klasa 2 głosuje z mocą $\sqrt{2} + \sqrt{3}$

Najbliższymi sąsiadami x_1 w klasie decyzyjnej 4 są y_6, y_5

Klasa 4 głosuje z mocą $\sqrt{2} + \sqrt{3}$

$$\sqrt{2} + \sqrt{3} = \sqrt{2} + \sqrt{3}$$

Stąd obiekt x_1 nie jest chwytny, nie jesteśmy w stanie powiedzieć, która klasa jest bliżej w sensie dwóch najbliższych sąsiadów.

Dla x_2 1 2 1 1 2

$$d(x_2, y_1) = 1$$

$$d(x_2, y_2) = \sqrt{84}$$

$$d(x_2, y_3) = \sqrt{2}$$

$$d(x_2, y_4) = \sqrt{166}$$

$$d(x_2, y_5) = \sqrt{15}$$

$$d(x_2, y_6) = \sqrt{2}$$

Klasa 2 głosuje z mocą $1 + \sqrt{2}$

Klasa 4 głosuje z mocą $\sqrt{2} + \sqrt{15}$

$$1 + \sqrt{2} < \sqrt{2} + \sqrt{15}$$

Obiekt x_2 otrzymuje decyzję 2, jest poprawnie sklasyfikowany.

Dla x_3 9 7 10 7 4

$$d(x_3, y_1) = \sqrt{197}$$

$$d(x_3, y_2) = \sqrt{117}$$

$$d(x_3, y_3) = \sqrt{182}$$

$$d(x_3, y_4) = \sqrt{50}$$

$$d(x_3, y_5) = \sqrt{129}$$

$$d(x_3, y_6) = \sqrt{182}$$

Klasa 2 głosuje z mocą $\sqrt{117} + \sqrt{182}$

Klasa 4 głosuje z mocą $\sqrt{50} + \sqrt{129}$

$$\sqrt{50} + \sqrt{129} < \sqrt{117} + \sqrt{182}$$

Obiekt x_3 otrzymuje decyzję 4, jest poprawnie sklasyfikowany.

Dla x_4 4 4 10 10 2

$$d(x_4, y_1) = \sqrt{172}$$

$$d(x_4, y_2) = \sqrt{182}$$

$$d(x_4, y_3) = \sqrt{167}$$

$$d(x_4, y_4) = \sqrt{151}$$

$$d(x_4, y_5) = \sqrt{130}$$

$$d(x_4, y_6) = \sqrt{167}$$

Klasa 2 głosuje z mocą $\sqrt{167} + \sqrt{172}$

Klasa 4 głosuje z mocą $\sqrt{130} + \sqrt{151}$

$$\sqrt{130} + \sqrt{151} < \sqrt{167} + \sqrt{172}$$

Obiekt x_4 otrzymuje decyzję 4, jest błędnie sklasyfikowany.

Podsumowując klasyfikację:

Obiekt x_1 nie jest chwytny

Obiekt x_2 otrzymuje decyzję 2, jest poprawnie sklasyfikowany

Obiekt x_3 otrzymuje decyzję 4, jest poprawnie sklasyfikowany

Obiekt x_4 otrzymuje decyzję 4, jest błędnie sklasyfikowany.

Macierz Predykcji, stanowiąca raport z klasyfikacji możemy zobaczyć w Tab. 3.

Tabela 3: Macierz Predykcji

	2	4	<i>No. of obj.</i>	<i>Accuracy</i>	<i>Coverage</i>
2	1	1	2	0.5	1.0
4	0	1	2	1.0	0.5
<i>True Positive Rate</i>	1.0	0.5			