

## MIW Mini-projekt 2 – kNN i regresja logistyczna

### Zadanie:

1. (2 punkty) Korzystając ze zbiorów Repozytorium Uczenia Maszynowego UCI (link poniżej) wybierz dane, które poddasz klasyfikacji:
  - dokonaj przeglądu wybranego zbioru,
  - wytypuj kolumny, które wykorzystasz jako parametry oraz etykiety (klasy),
  - wybór poprzyj analizą: dlaczego na podstawie wybranych parametrów/właściwości zaklasyfikujesz dany obiekt do klasy? Nie muszą to być naukowe opracowania poparte szerokimi badaniami; wystarczy zdroworozsądkowe wnioskowanie,
  - napisz uzasadnienie swojego wyboru.Link: <https://archive.ics.uci.edu/ml/datasets.php?format=&task=cla&att=&area=&numAtt=&numIns=&type=&sort=nameUp&view=table>
2. (4 punkty) Zaimplementuj klasyfikator kNN, a ponadto:
  - a. podziel cały zbiór na dwa mniejsze: trenujący i testowy,
  - b. zwizualizuj wyniki klasyfikacji,
  - c. za pomocą zbiorów z punktu (2a) zweryfikuj dokładność klasyfikacji zaimplementowanego algorytmu; miarą dokładności może być procent poprawnie zaklasyfikowanych obiektów w całym zbiorze testowym,
  - d. zaprezentuj działanie algorytmu wykorzystując dwie różne miary odległości: Euklidesową i Manhattan.
3. (4 punkty) Zaimplementuj klasyfikator wykorzystujący regresję logistyczną, a ponadto:
  - a. podziel cały zbiór na dwa mniejsze: trenujący i testowy,
  - b. określ liczbę iteracji działania algorytmu
  - c. zwizualizuj wyniki klasyfikacji
  - d. utwórz macierz błędów (ang. *confusion matrix*) – można posłużyć się gotowymi rozwiązaniami (np. sklearn.metrics)

Przy klasyfikacji wieloklasowej można posłużyć się klasyfikatorami binarnymi, na zasadzie: jedna klasa przeciwko pozostałym.

Uwaga (1): w punktach (2) i (3) wykorzystaj dane wybrane w punkcie (1)

Uwaga (2): zabronione jest posługiwanie się klasyfikatorami z gotowych bibliotek! Uwaga dotyczy tylko klasyfikatorów. Pisząc klasyfikator można natomiast korzystać z materiałów przesłanych przez prowadzącego ćwiczenia oraz przykładów i fragmentów kodu udostępnionych przez wykładowcę. Zewnętrzne biblioteki mogą posłużyć do celów innych niż klasyfikacja (np. operacje na macierzach, wizualizacja itd.).

Uwaga (3): zastosowanie klasyfikatora binarnego w danym zadaniu implementacyjnym (zad. 2 i 3) skutkować będzie odjęciem 1-go punkta za zadanie, czyli w sumie maksymalnie -2 punkty.

Uwaga (4): wybór trywialnego zbioru skutkuje -1 punktem za zadanie 1 (trywialny jest np. zbiór Iris)

Uwaga (5): kod musi być napisany posługując się wyłącznie językiem angielskim, komentarze piszemy w języku polskim

Uwaga (6): za nieczytelny kod będę przyznawał -1 punkt – zgodnie z poprzednimi wytycznymi

Uwaga (7): niechlujne prace (czyli wszystko poza kodem) także zasłużą na -1 punkt

### Forma:

1. Komentowanie kodu jest na ogół złą praktyką. Jednak w tym przypadku wynikowy kod wraz z wartościami wyjściowymi wypisanymi np. na konsolę stanowi dokumentację naszej pracy. Kluczowe elementy kodu i wartości wynikowe proszę skrótkowo, aczkolwiek dostatecznie jasno, opisywać i komentować.

2. Mini-projekt można oddać w formie (do wyboru):
  - a. Wyeksportowanego do html/pdf pliku ipynb. Proszę pamiętać o wszystkich wartościach wynikowych, czyli kod ipynb musi zostać wykonany przed eksportem.
  - b. Sprawozdania w pliku pdf. Sprawozdanie ma posiadać pełny kod wraz z dokumentacją oraz wszystkimi niezbędnymi wartościami wynikowymi.
3. Sprawozdanie należy wysłać na adres e-mail [pprzestrzelski@pjawst.edu.pl](mailto:pprzestrzelski@pjawst.edu.pl):
  - a. nazwa pliku: *NR\_INDEKSU\_Imię\_Nazwisko.ext*, gdzie NR\_INDEKSU to np. S999, a ext to rozszerzenie pliku, czyli html lub pdf – w zależności od wybranego wariantu formy oddania mini-projektu (2a lub 2b),
  - b. tytuł maila: *[MIW] Mini-projekt 2*.

**Termin:**

1. 06.04.2020 r., godzina 15:45 (100 % punktów)
2. 13.04.2020 r., godzina 15:45 (100 % punktów)
3. 20.04.2020 r., godzina 15:45 (50 % punktów)