Zastosowania informatyki w medycynie
Projekt: Zasady Zaliczenia

Prowadzący: Dr inż. Paweł Ksieniewicz [Grupa TN19]

1 Projekt – informacje i zasady zaliczenia

1.1 Cel projektu

Celem projektu jest nabycie umiejętności zastosowania wybranych algorytmów klasyfikacji nadzorowanej w praktycznym zadaniu diagnostyki medycznej wraz z selekcją cech i eksperymentalną oceną skuteczności algorytmu na danych rzeczywistych oraz sprawdzenie, jak jakość klasyfikacji zależy od liczby atrybutów wykorzystanych w konstrukcji modelu.

1.2 Zespoły projektowe

Zespoły projektowe powinny składać się z **dwóch** osób. Skład zespołu projektowego **musi** być zgłoszony prowadzącemu. Członkowie zespołu ponoszą **wspólną** odpowiedzialność za całość projektu.

1.3 Język programowania

Projekt powinien zostać wykonany w języku programowania **Python** z wykorzystaniem biblioteki uczenia maszynowego *scikit-learn*[1].

1.4 Etapy realizacji projektu

I Zapoznanie się z algorytmami diagnostycznymi (algorytmami klasyfikacji), określonymi w temacie.

II Zapoznanie się z materiałem empirycznym – zdefiniowaanie problemu rozpoznawania (klasyfikacji) – określenie liczby i znaczenia klas, liczby i znaczenia cech oraz charakteru cech (ciągłe, dyskretne, itd.).

III Wyznaczenie rankingu cech pod względem ich przydatności do klasyfikacji, korzystając z dowolnej miary (kryterium) jakości cech stosowanych w selekcji cech z grupy metod zwanych filtrami (ang. *Univariate Filter Methods*), np. *Kolmogorov–Smirnov test* – polecany ze względu na łatwość obliczeń.

IV Zaplanowanie badań eksperymentalnych dla następujących założeń:
   a Ewaluacja wykorzystanego klasyfikatora z wykorzystaniem 5 razy powtarzanej metody 2-krotnej walidacji krzyżowej (ang. *Cross-validation*). Jakość klasyfikacji (poprawność diagnozy) należy mierzyć częstością poprawnych rozpoznań (diagnoz) na zbiorze testującym.

[1] `https://scikit-learn.org`

b Przygotowanie środowiska eksperymentalnego oraz algorytmu diagnostycznego (klasyfikacji), tak aby można było przeprowadzić badania eksperymentalne dla przedstawionych parametrów:
   - Sztuczne sieci neuronowe (MLP) – sieć jednokierunkowa z 1 warstwą ukrytą dla 3 różnych liczb neuronów w warstwie ukrytej oraz dla uczenia metodą propagacji wstecznej z momentum i bez momentum.
   - K-NN – dla 3 różnych wartości k (1, 5, 10) oraz dla 2 różnych miar odległości (w tym euklidesowej).
   - Naiwny algorytm Bayesa (zakładamy, że cechy są niezależne) – algorytm dla 0-1 funkcji strat. Prawdopodobieństwa *a priori* oraz prawdopodobieństwa cech w poszczególnych klasach (dla cech dyskretnych) szacujemy (estymujemy) ze zbioru uczącego metodą częstotliwościową i/lub warunkowe gęstości (ang. *Density estimation*) cech w klasach (dla cech ciągłych) szacujemy metodą histogramu, metodą empirycznej dystrybuanty lub metodami jądrowymi (ang. *Kernel methods*). **W przypadku wybrania tematu wykorzystującego naiwny algorytm Bayesa, należy samodzielnie zaimplementować poprawny klasyfikator *scikit-learn* zgodnie z powyższymi wytycznymi. Gaussian Naive Bayes z *scikit-learn* nie będzie wystarczający.**

c Badania należy przeprowadzić dla różnej liczby cech (poczynając od jednej - najlepszej wg. wyznaczonego rankingu, a następnie dokładać kolejno po jednej (również według wyznaczonego rankingu) tak długo, aż zostanie znaleziona najlepsza liczba cech. Dodawanie cech powinno poprawiać jakość klasyfikacji, ale do pewnego momentu – dalsze dodawanie cech jakość pogorszy. Trzeba znaleźć optimum. Jeżeli cech jest mało (< 7), to przeprowadzić badania dla wszystkich cech.

d Dla każdego pojedynczego eksperymentu (pojedynczy eksperyment to doświadczalne wyznaczenie jakości klasyfikacji dla danego algorytmu, danych wartości parametrów algorytmu i dla danej liczby cech) należy przedstawić wyniki (jakości klasyfikacji) w formie uśrednionej (względem 5 powtórzeń metody 2-krotnej walidacji krzyżowej). Dodatkowo, dla najlepszego przypadku należy przedstawić macierz konfuzji (pomyłek) (ang. *Confusion matrix*).

V Zrealizowanie badań eksperymentalnych według przedstawionych w punkcie IV założeń.

VI Przeanalizowanie uzyskanych wyników, przeprowadzenie dyskusji oraz przedstawienie wniosków.

1.5 Warunki zaliczenia projektu

1. Stawienie się (w pełym składzie) na zajęciach/konsultacjach i omówienie z prowadzącym każdego z kamieni milowych projektu (najpóźniej w dniu wskazanym jako deadline) po jego uprzednim wydruku.
2. Sporządzenie sprawozdania z wykonanego projektu zawierającego:

a Opis problemu medycznego jako zadania klasyfikacji (liczba klas i ich medyczny sens, liczba cech i ich charakterystyka (znaczenie, czy ciągłe, czy dyskretne), liczba danych w dostępnym zbiorze).
b Przedstawienie zastosowanego algorytmu selekcji cech – forma opisu algorytmu: patrz punkt następny.
c Przedstawienie stosowanego algorytmu: dla algorytmów minimalno-odległościowych i naiwnego algorytmu bayesowskiego w formie, w jakiej się algorytmy przedstawia (schemat blokowy, pseudokod, formuły matematyczne – wszystko powinno być precyzyjne, aby można było z opisu utworzyć kod), dla sztucznych sieci neuronowych opisać precyzyjnie strukturę stosowanej sieci (liczba warstw, liczba neuronów), funkcję aktywacji, parametry metody uczenia propagacji wstecznej.
d Opis środowiska programistycznego – krótko.
e Przedstawienie planu eksperymentu oraz jego wyników (w formie syntetycznej: tabela lub wykres + macierz konfuzji).
f Dyskusję otrzymanych wyników – czy są zauważalne jakieś prawidłowości, czy można sformułować jakieś wnioski, itp.
g Wykorzystaną literaturę.

W przypadku niespełnienia chociaż jednego z tych warunków projekt uznaje się za niezaliczony.

Kamienie milowe:

I Wprowadzenie do projektu, wybór tematów oraz grup projektowych (pierwsze zajęcia).
II Opis problemu medycznego jako zadania klasyfikacji oraz wyznaczenie rankingu cech pod względem ich przydatności do klasyfikacji (28 kwietnia).
III Implementacja środowiska eksperymentowania (26 maja).
IV Przedstawienie wstępnych wyników eksperymentów i próba sformułowania pierwszych wniosków (9 czerwca).
V Oddanie finalnej wersji projektu (23 czerwca):

1.6 Plagiaty

Prace oddawane przez studentów są sprawdzane pod kątem wykrywania plagiatów. W przypadku stwierdzenia plagiatu grupy projektowe, których prace noszą znamiona plagiatu, otrzymują za projekt ocenę niedostateczną. Stwierdzenie plagiatu w jednej z części projektu jest jednoznaczne z uznaniem całego projektu za plagiat. Nie jest możliwa poprawa. Innymi słowy – popełnienie plagiatu skutkuje niezaliczeniem przedmiotu.

1.7 Ocena końcowa – zasady

Na ocenę końcową składają się następujące elementy:

1. Poprawność realizacji projektu (50% oceny końcowej):

   - Poprawne wyznaczenie rankingu cech.
   - Zaplanowanie i przeprowadzenie badań eksperymentalnych według właściwych założeń.
   - Właściwa implementacja środowiska eksperymentowania.
2. Jakość sporządzonej dokumentacji – Sekcja 1.5 punkt 2. (40% oceny końcowej)
3. Przygotowany kod / środowisko eksperymentowania (10% oceny końcowej)
   - Znajomość przygotowanego kodu.
   - Komentarze w kodzie.

Procenty uzyskane za każdy z elementów zostaną zsumowane, a ocena końcowa będzie wystawiana według poniższej skali:

Uzyskany %	**Ocena**
$> 60\%$	2.0
$[60; 70]$	3,0
$(70; 80]$	3,5
$(80; 90]$	4,0
$(90; 95]$	4,5
$(95; 100]$	5,0

Poszczególne kamienie milowe nie są oceniane osobno, ale jakość ich przygotowania ma wpływ na ocenę końcową.

Uwaga! Demonstracja rażącego braku wiedzy na temat któregokolwiek z elementów projektu (przeprowadzonych badań, sprawozdania lub zaprezentowanego kodu) wiąże się z otrzymaniem oceny niedostatecznej.

2 Lista tematów projektowych

Komputerowe wspomaganie diagnozowania ...

1. ... białaczek u dzieci z wykorzystaniem sztucznych sieci neuronowych.
2. ... białaczek u dzieci z wykorzystaniem naiwnego algorytmu bayesowskiego.
3. ... białaczek u dzieci z wykorzystaniem algorytmu k-NN.
4. ... choroby niedokrwiennej u dzieci z wykorzystaniem sztucznych sieci neuronowych.
5. ... choroby niedokrwiennej u dzieci z wykorzystaniem naiwnego algorytmu bayesowskiego.
6. ... choroby niedokrwiennej u dzieci z wykorzystaniem algorytmu k-NN.
7. ... stanów ostrego brzucha z wykorzystaniem sztucznych sieci neuronowych.
8. ... stanów ostrego brzucha z wykorzystaniem naiwnego algorytmu bayesowskiego.
9. ... stanów ostrego brzucha z wykorzystaniem algorytmu k-NN.
10. ... zawałów z wykorzystaniem sztucznych sieci neuronowych.
11. ... zawałów z wykorzystaniem naiwnego algorytmu bayesowskiego.
12. ... zawałów z wykorzystaniem algorytmu k-NN.

13. ... nowotworów piersi z wykorzystaniem sztucznych sieci neuronowych.
14. ... nowotworów piersi z wykorzystaniem naiwnego algorytmu bayesowskiego.
15. ... nowotworów piersi z wykorzystaniem algorytmu k-NN.
16. ... chorób tarczycy z wykorzystaniem sztucznych sieci neuronowych.
17. ... chorób tarczycy z wykorzystaniem naiwnego algorytmu bayesowskiego.
18. ... chorób tarczycy z wykorzystaniem algorytmu k-NN.
19. ... ostrego zapalenia dróg moczowych z wykorzystaniem naiwnego algorytmu bayesowskiego.
20. ... ostrego zapalenia dróg moczowych z wykorzystaniem algorytmu k-NN.
21. ... ostrego zapalenia dróg moczowych z wykorzystaniem sztucznych sieci neuronowych
22. ... zapalenia wątroby z wykorzystaniem naiwnego algorytmu bayesowskiego.
23. ... zapalenia wątroby z wykorzystaniem algorytmu k-NN.
24. ... zapalenia wątroby z wykorzystaniem sztucznych sieci neuronowych.
25. ... chorób wątroby z wykorzystaniem naiwnego algorytmu bayesowskiego.
26. ... chorób wątroby z wykorzystaniem algorytmu k-NN.
27. ... chorób wątroby z wykorzystaniem sztucznych sieci neuronowych,

2.1 Dane do projektów:

1. Tematy 1–12 — dane na serwerze[2].
2. Pozostałe tematy — dane na *UCI Machine Learning repository* [3]
   a Tematy 13–15 — Breast Cancer Wisconsin (Original).
   b Tematy 16–18 — Thyroid Disease.
   c Tematy 19–21 — Acute Inflammations.
   d Tematy 22–24 — Hepatitis.
   e Tematy 25–27 — ILPD (Indian Liver Patient Dataset).

3 Literatura

1. Marek Kurzyński, Rozpoznawanie obiektów – metody statystyczne, Oficyna Wyd. Politechniki Wrocławskiej, Wrocław 1998.
2. Stanisław Bielawski, Modele farmakokinetyczne, WKiŁ, Warszawa 1989.
3. W. Sobczak, W. Malina, Metody selekcji i redukcji informacji, WNT, Warszawa 1988.
4. J. Ćwik, J. Mielniczuk, Statystyczne systemy uczące się. Ćwiczenia w oparciu o pakiet R, Oficyna Wyd. Pol. Warszawskiej, Warszawa 2009.
5. M. Kurzyński, Metody sztucznej inteligencji dla inżynierów, PWSZ Legnica 2009.
6. K. Krawiec, J. Stefanowski, Uczenie maszynowe i sieci neuronowe, Wydawnictwo Pol.Poznańskiej, Poznań 2004.

[2] `http://156.17.43.89/zbiory-ziwm.zip`
[3] `https://archive.ics.uci.edu/ml/datasets.php`