Laboratorium 11

Autorzy: Adam Kuzdraliński, Marek Miśkiewicz
Nazwa pliku: imie_nazwisko_11_bio.pdf
Typ ćwiczenia: jednotygodniowe
Cel: Celem ćwiczenia jest zapoznanie studentów z możliwościami zastosowania uczenia maszynowego w analizie danych biologicznych z wykorzystaniem programowania wizualnego

1. Jak zbudować swój pierwszy model klasyfikacji za pomocą programowania wizualnego. Na początek zadziałamy odtwórczo:
   a. W narzędziu Orange, w oknie “Welcome to Orange” otwórz przykłady (examples), a następnie “Classification tree”
   b. Przyjrzyj się wizualizacji i staraj się zrozumieć jak w tym przypadku działa algorytm
   c. Jaką nazwę ma plik, z którego pochodzą analizowane dane (rozwiń menu podręczne przy “File” i kliknij “Open”) [Task 1]. Co ten plik zawiera - opisz krótko ten dataset [Task 2].
   d. Jaką metodę klasyfikacji danych zastosowano tutaj [Task 3].
   e. Kliknij dwukrotnie “Classification Tree Viewer”, a następnie krótko zinterpretuj widok dla tych danych [Task 4].
   f. ile maksymalnie poziomów ma drzewo dla tych danych (zobacz parametr “Depth) [Task 5].
   g. pobierz to drzewo i umieść w raporcie [Task 6]
   h. obejrzyj wizualizację wykonaną na wykresie “Scatter Plot”, zaznacz “Show color regions”. Na drzewie “Classification Tree Viewer” zaznacz tylko gałęzie zawierające “Iris virginica”, a następnie umieść wykres “Scatter Plot” z zaznaczonymi rekordami tylko dla “Iris virginica” w raporcie [Task 7]

2. Otwórz narzędzie “Orange”, utwórz nowy projekt i wczytaj dataset iris.tab (Data->File):
   a. upewnij się, że wszystkie kolumny, których nazwa zaczyna się od “sepal” lub “petal” mają rolę “feature”, a kolumna iris - “Target”. Na koniec kliknij “Apply”.
   b. utwórz kafelek “Data table” żeby móc obejrzeć te dane w tabeli. Ile rekordów zawiera ten dataset? [Task 8]
   c. od ikony File poprowadź nową gałąź i na jej końcu umieść “Test and Score” (możesz wyszukać wśród dostępnych opcji). Następnie od “Test and Score” poprowadź nową gałąź i na jej końcu umieść “Logistic regression”
   d. kliknij dwa razy w “Test and Score” i odczytaj parametry modelu. Czy model dobrze przewiduje gatunek, uzasadnij [Task 9]
   e. Podobnie jak “Logistic regression” poprowadź nową gałąź od “Test and Score” i wybierz “Tree” - jest to algorytm będący prekursorem innego, często wykorzystywanego w ML alorytmu tj. random forest.
   f. Podobnie jak “Logistic regression” oraz “Tree” przetestyj dane z algorytmem “SVM”
   g. od “Test and Score” poprowadź nową gałąź, na jej końcu umieść “Confusion Matrix” i odpowiedz na pytanie: który model radzi sobie najlepiej i czy różnice między modelami są duże? [Task 10]
   h. od “Confusion Matrix” poprowadź nową gałąź i na jej końcu umieść “Data Table”. Zaznacz w “Confusion Matrix” 5 rekordów, które zostały źle zidentyfikowane z użyciem algorytmu “Logistic regression” dzięki czemu wyświetlą się w “Data Table” i stamtąd skopiuj je w formie tabeli jako odpowiedź na to zadanie [Task 11].

"Train dataset" (zbiór treningowy) to zestaw danych, który jest używany do uczenia modelu w uczeniu maszynowym. Model jest trenowany na tym zbiorze danych, aby nauczyć się, jak przewidywać wyniki na podstawie określonych cech.

"Validation dataset" (zbiór walidacyjny) to zestaw danych, który jest używany do oceny jakości modelu w trakcie jego uczenia. Model jest testowany na tym zbiorze danych, aby zobaczyć, jak dobrze radzi sobie z przewidywaniem wyników na danych, które nie zostały użyte do uczenia.

"Test dataset" (zbiór testowy) to zestaw danych, który jest używany do ostatecznej oceny jakości modelu. Model jest testowany na tym zbiorze danych, który nie był używany ani do uczenia, ani do walidacji, aby zobaczyć, jak dobrze radzi sobie z przewidywaniem wyników na nowych danych.

3. W narzędziu “Orange” utwórz nowy projekt, następnie:
   a. wczytaj dataset iris.tab (Data->File) i zastosuj ustawienia z punktu 2a
   b. od ikony File poprowadź nową gałąź i na jej końcu umieść “Data sampler”. W dokumentacji Orange znajdź odpowiedź na pytanie do czego służy “Data sampler”, wklej ją jako odpowiedź na ten task [Task 12]
   c. w “Data sampler” ustaw “Fixed proportion of data” na 75%. 75% będzie użyte do trenowania, a 25% do testowania.
   d. od ikony “Data sampler” poprowadź nową gałąź i na jej końcu umieść “Test and Score”
   e. dwukrotnie kliknij w linię łączącą “Data sampler” z “Test and Score”. “Data sample” i “Data” powinna łączyć linia, podobnie połącz “Remaining data” z “Test data”.
   f. wejdź do “Test and score” i zaznacz “Test on test data”
   g. do “Test and score” podłącz dwa modele tj. “Logistic regression” oraz “Tree”, na podobnej zasadzie jak było to robione w punkcie drugim.
   h. spójrz na wyniki w “Test and score” i wyjaśnij skrótowo jak działa ten algorytm [Task 13].

4. Teraz spróbujemy w inny sposób:
   a. utwórz nowy obiekt “File” i wczytaj wczytaj dataset iris.tab z takimi ustawieniami jak poprzednio
   b. utwórz drugi obiekt “File” i wczytaj plik iris_test.tab (credits to Hussam Hourani) z takimi ustawieniami jak plik iris.tab
   c. Oba pliki połącz z nowym obiektem “Test and Score”, w którym zaznacz “Test on test data”.
   d. dwukrotnie kliknij w linię łączącą “Test and Score” z obiektem “File” zawierającym zbiór danych iris.tab i dopilnuj żeby linia łączyła “Data” z “Data”, podobnie dwukrotnie kliknij w linię łączącą “Test and Score” z obiektem “File” zawierającym zbiór danych iris_test.tab i dopilnuj żeby linia łączyła “Data” z “Test Data”
   e. podłącz do “Test and Score” modele “Logistic regression” oraz “Tree” i odpowiedz na pytanie jak działa ten algorytm w porównaniu z tymi powyżej z punktu 3, tzn. co go odróżnia [task 14].

5. W narzędziu “Orange” utwórz nowy projekt, następnie:
   a. wczytaj dataset iris.tab (Data->File) i zastosuj ustawienia z punktu 2a
   b. do obiektu “File” podłącz “Test and Score”, do “Test and Score” podłącz “Confusion Matrix”, a do “Confusion Matrix” podłącz “Data table”.
   c. do “Test and Score” podłącz “Logistic regression”. W “Test and Score” pamiętaj, aby zaznaczone było “Cross validation”
   d. W “Confusion matrix” zobacz ile rekordów zostało źle sklasyfikowanych, podaj liczbę [task 15].

e. analizowane dane przetestuj również modelami: “Tree”, “Random forest”, “Neural network”, “Naive bayes”, “kNN”, “Gradient boosting”, “Adaboost”. Na podstawie parametru CA z “Test and Score” określ, który z modeli najgorzej sobie radzi z klasyfikacją danych (uwaga! ten parametr nie zawsze jest najlepszym wskaźnikiem jakości modelu. W tym przypadku jednak upraszczamy i przyjmujemy, że jest) [task 16].

> Analiza ROC (Receiver Operating Characteristic) to technika wykorzystywana w uczeniu maszynowym do oceny jakości modelu klasyfikacyjnego. Jest to wykres, który porównuje zdolność modelu do prawidłowego wykrywania prawdziwie pozytywnych przypadków (czułość) z jego zdolnością do generowania fałszywie pozytywnych wyników.
> W prostych słowach, analiza ROC pokazuje, jak dobrze model potrafi rozróżniać między dwiema klasami, na przykład czy pacjent jest chory, czy zdrowy.
> Im bliżej krzywej ROC jest do lewego górnego rogu wykresu, tym lepiej model radzi sobie z klasyfikacją. Idealny model miałby czułość równą 1 i specyficzność równą 1, co oznaczałoby, że prawidłowo klasyfikuje wszystkie przypadki.
> Często używana metryka do oceny jakości modelu na podstawie krzywej ROC to pole pod krzywą ROC (AUC-ROC). Wartość AUC-ROC wynosi od 0 do 1, gdzie 1 oznacza doskonałą klasyfikację, a 0,5 oznacza klasyfikację na poziomie losowym. Im większa wartość AUC-ROC, tym lepiej model radzi sobie z klasyfikacją.

f. do “Test and Score” podłącz “ROC analysis” i odpowiedz na pytania: dla którego gatunku wszystkie modele okazały się idealnie (jest tylko jeden wyjątek) klasyfikować dane [task 17],
g. w “Confusion Matrix” sprawdź jaką liczbę błędów popełnił najsłabszy model. Podaj tą liczbę w raporcie [task 18].

6. W narzędziu “Orange” utwórz nowy projekt, następnie:
   a. wczytaj dataset iris.tab (Data->File) i zastosuj ustawienia z punktu 2a
   b. tak jak we wcześniejszych metodykach podłącz do tych danych modele “Logistic regression” oraz “Tree” (pamiętaj o “Test and score”). Do “Test and score” podłącz “Confusion matrix”.
   c. Utwórz obiekt “Predictions” i połącz go z “Logistic regression” oraz “Tree”
   d. Utwórz obiekt “File” i załaduj do niego plik iris_predict.tab (credits to Hussam Hourani) i połącz z obiektem “Predictions”
   e. połącz obiekt “Logistic regression” oraz “Tree” z obiektem “File”, który zawiera zbiór danych iris.tab. Linie łączące te obiekty z obiektem “File” powinny znajdować się z drugiej strony względem innych linii, które już są połączone z “Logistic regression” oraz “Tree”
   f. zobacz w “Predictions” do jakich gatunków zostały sklasyfikowane analizowane rekordy z bazy iris_predict.tab, podaj te gatunki [task 19].
   g. w “Predictions” wybierz “Show probabilities for: Classes known to the model” i określ z jakim prawdopodobieństwem dokonana została klasyfikacja do każdego z gatunków [task 20].
   h. podłącz dodatkowy model tj. “Neural network” i odpowiedz na pytanie czy klasyfikacja z użyciem tego modelu różni się [task 21].
   i. sprawdź inne znane Ci modele i wklej do raportu tabelę predykcji (do “predictions” podłącz “data table”, a do “data table” podłącz “save data”, wygeneruj plik csv i wklej go do raportu) [task 22].