KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Instrukcja współfinansowana przez Unię Europejską
w ramach Europejskiego Funduszu Społecznego
w projekcie

*„Innowacyjna dydaktyka bez ograniczeń
– zintegrowany rozwój Politechniki Łódzkiej – zarządzanie Uczelnią,
nowoczesna oferta edukacyjna i wzmacniania zdolności
do zatrudniania osób niepełnosprawnych”*

Instrukcja jest dystrybuowana bezpłatnie

Instrukcja do laboratorium, część 1

dr inż. Artur Klepaczko

Eksploracja danych

Zadanie nr 13 – Studia podyplomowe „Przetwarzanie i analiza obrazów biomedycznych”

Politechnika Łódzka
Instytut Elektroniki

90-924 Łódź, ul. Żeromskiego 116,
tel. 042 631 28 83
www.kapitalludzki.p.lodz.pl

Spis treści

1 Wstęp

Celem ćwiczenia jest zapoznanie się ze strukturą oraz sposobem pracy z programem Weka[1], zawierającym zestaw narzędzi do eksploracji danych. Program wyposażony jest w rozbudowany i jednocześnie prosty w obsłudze graficzny interfejs użytkownika. Dzięki temu możliwe jest rozwiązywanie szeregu zadań m.in. z zakresu klasyfikacji, regresji, transformacji oraz grupowania danych. Zamieszczony w punkcie 3. przykład pozwoli zapoznać się z typowym przebiegiem pracy z programem.

W ostatniej części instrukcji znajduje się specyfikacja zadań do samodzielnego rozwiązania w trakcie zajęć laboratoryjnych. W zadaniach tych należy zastosować różne metody klasyfikacji danych opartych na regułach lub drzewach decyzyjnych. Zbiory danych, których będą one dotyczyły pochodzą z ogólnodostępnego repozytorium danych wzorcowych – *UCI Machine Learning Repository*[2].

Oczekiwane efekty kształcenia:

→ Znajomość podstawowych zasad pracy z programem Weka.

→ Umiejętność rozwiązywania podstawowych zadań klasyfikacji danych za pomocą reguł i drzew decyzyjnych.

2 Wprowadzenie do programu Weka

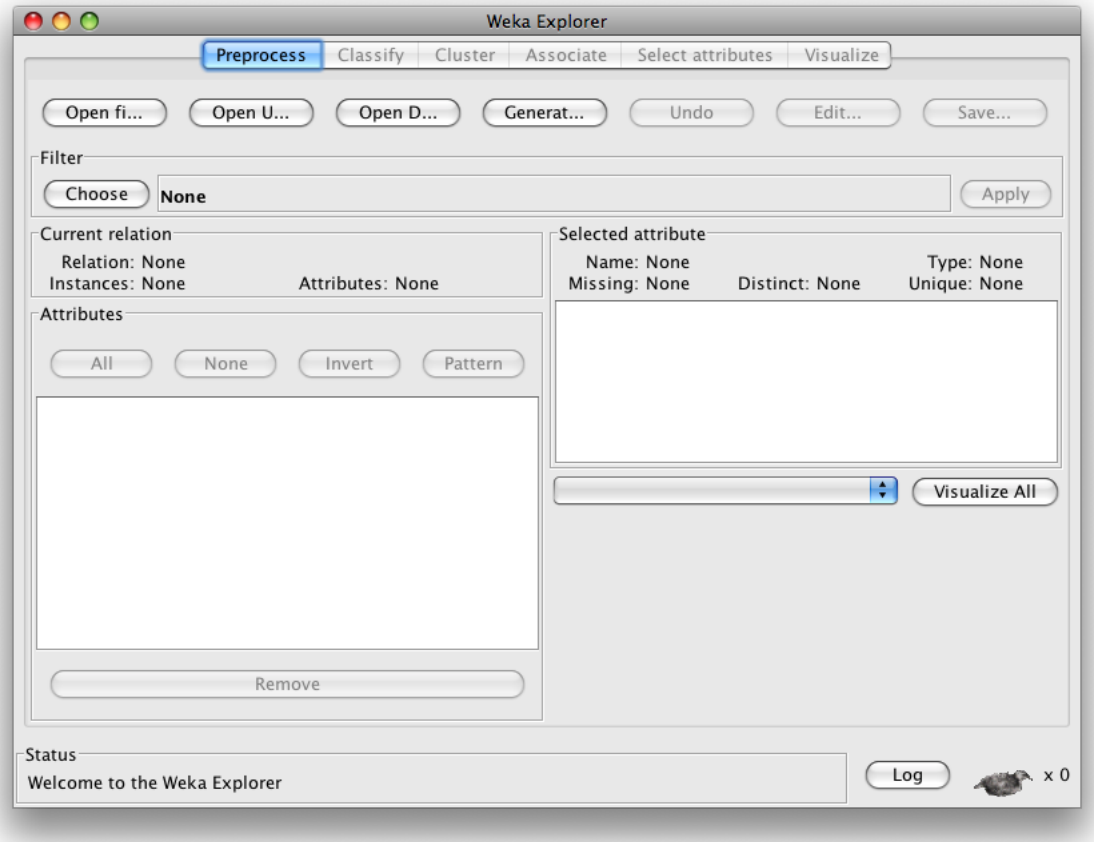
2.1 Charakterystyka programu

Program *Weka* powstał w Uniwersytecie Waikato w Nowej Zelandii. Słowo *weka* pochodzi od wyrażenia *Waikato Environment for Knowledge Analysis* i jednocześnie stanowi nazwę nielota (zob. rys. 1) o wyjątkowo wścipskiej naturze, spotykanego wyłącznie na nowozelandzkich wyspach. Program zawiera bogaty zestaw narzędzi do eksploracji danych, w tym do ich wizualizacji, klasyfikacji, grupowania oraz wielorakiego filtrowania. Pozwala także na automatyzację całego procesu przetwarzania danych obejmującego ich wczytanie (z pliku bądź też ze zdalnego serwera bazodanowego), wstępną transformację, właściwe przetwarzanie, przetwarzanie końcowe oraz ocenę uzyskanych wyników.

Rysunek 1: Weka – skrzyżowanie kiwi z kurą

Należy zauważyć, że program Weka jest rozpowszechniany łącznie z pełną biblioteką kodów źródłowych na zasadach licencji GNU GPL[3]. Dzięki temu możliwe jest korzystanie z

[1] Program można pobrać bezpłatnie ze strony http://www.cs.waikato.ac.nz/ml/weka/.
[2] Repozytorium jest dostępne bezpłatnie w Internecie pod adresem http://archive.ics.uci.edu/ml/.
[3] *General Public License*.



Rysunek 2: Interfejs modułu *Explorer* w programie Weka

gotowych i sprawdzonych implementacji szeregu złożonych algorytmów nie tylko poprzez graficzny interfejs użytkownika lecz także z poziomu własnych programów. Jedynym ograniczeniem może być tutaj język programowania — programista chcący wykorzystać bibliotekę Weka powinien umieć posługiwać się Javą.

2.2 Podstawowe moduły programu

Weka składa się z trzech głównych modułów programowych: *Explorer*, *Experimenter* oraz *KnowledgeFlow*. W tej części laboratorium wykorzystywany będzie tylko pierwszy z nich, umożliwiający rozwiązywanie pojedycznych i niekoniecznie powiązanych ze sobą zadań eksploracji danych. Jednocześnie wprowadza on najprostrzy sposób pracy z programem, a przy tym nie ogranicza zakresu odkryć, jakich można dokonać dla analizowanego zbioru danych.

Rysunek 2 przedstawia główne okno modułu *Explorera* tuż po jego otwarciu. Jak widać, zawiera ono sześć zakładek, z których każda umożliwia wykonanie odmiennych zadań eksploracji danych:

```
% Title: Iris Plants Database
@RELATION iris
@ATTRIBUTE sepallength   NUMERIC
@ATTRIBUTE sepalwidth   NUMERIC
@ATTRIBUTE petallength   NUMERIC
@ATTRIBUTE petalwidth   NUMERIC
@ATTRIBUTE class  {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1,3.5,1.4,0.2,Iris-setosa

                    ⋮
```

Rysunek 3: Fragment pliku danych w formacie *ARFF*

- **Preprocess** – wczytywanie i przetwarzanie wstępne danych,
- **Classify** – klasyfikacja,
- **Cluster** – analiza skupień,
- **Associate** – indukcja reguł asocjacyjnych,
- **Select attributes** – selekcja cech znaczących,
- **Visualize** – wizualizacja danych.

Na początku pracy z programem jedynie pierwsza zakładka pozostaje aktywna. Dostęp do pozostałych zakładek stanie się możliwy po wczytaniu pliku z danymi. Plik ten musi być zgodny z formatem wymaganym przez środowisko Weka (zob. pkt 2.3). Do wczytywania danych służy przede wszytskim przycisk *Open file...* Ponadto dane można pobrać ze zdalnego komputera (*Open URL...*) lub serwera bazodanowego (*Open Database...*). Z kolei za pomocą przycisku *Generate...* można wygenerować dane syntetyczne.

2.3 Format danych wejściowych

Dane wejściowe dla programu Weka muszą być przygotowane w specjalnym pliku w formacie *ARFF* (*Attribute Relation File Format*). Plik ten (obowiązuje rozszerzenie *.arff) jest zwykłym plikiem tekstowym, ale o określonej strukturze (zob. rys. 3).

Po pierwsze, znak % na początku wiersza informuje o tym, że następujący po nim tekst stanowi komentarz. Najczęściej jest to opis zbioru danych (nazwa, źródło pochodzenia, odniesienia do literatury) lub poszczególnych atrybutów użytych do opisu obiektów, których określony zbiór danych dotyczy.

Słowa kluczowe z punktu widzenia programu Weka wprowadzone są za pomocą znaku @ (z wyjątkiem typu atrybutów). W przykładzie pokazanym na rysunku występują trzy takie wyrażenia:

@RELATION – deklaracja nazwy zbioru danych,

@ATTRIBUTE – nazwa i typ atrybutu,

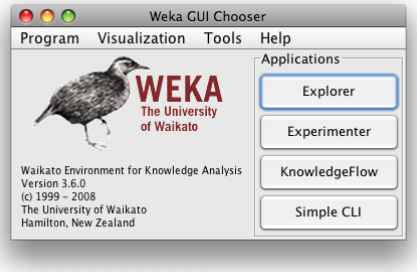
@DATA – początek sekcji, w której znajdują się wektory danych.

Zdefiniowane są dwa zasadnicze typy atrybutów: 1) ciągłe, deklarowane za pomocą słowa kluczowego *NUMERIC* oraz 2) nominalne, dla których należy określić zbiór możliwych wartości objętych nawiasami klamrowymi. Należy zauważyć, że etykieta klasy wektora danych również stanowi atrybut typu nominalnego.

Wektory danych znajdują się w osobnych wierszach następujących po słowie kluczowym @Data. Wartości poszczególnych atrybutów rozdzielone są przecinkami, przy czym ich uszeregowanie musi być zgodne z kolejnością deklarowania.

3 Praca z programem Weka

3.1 Uruchomienie programu

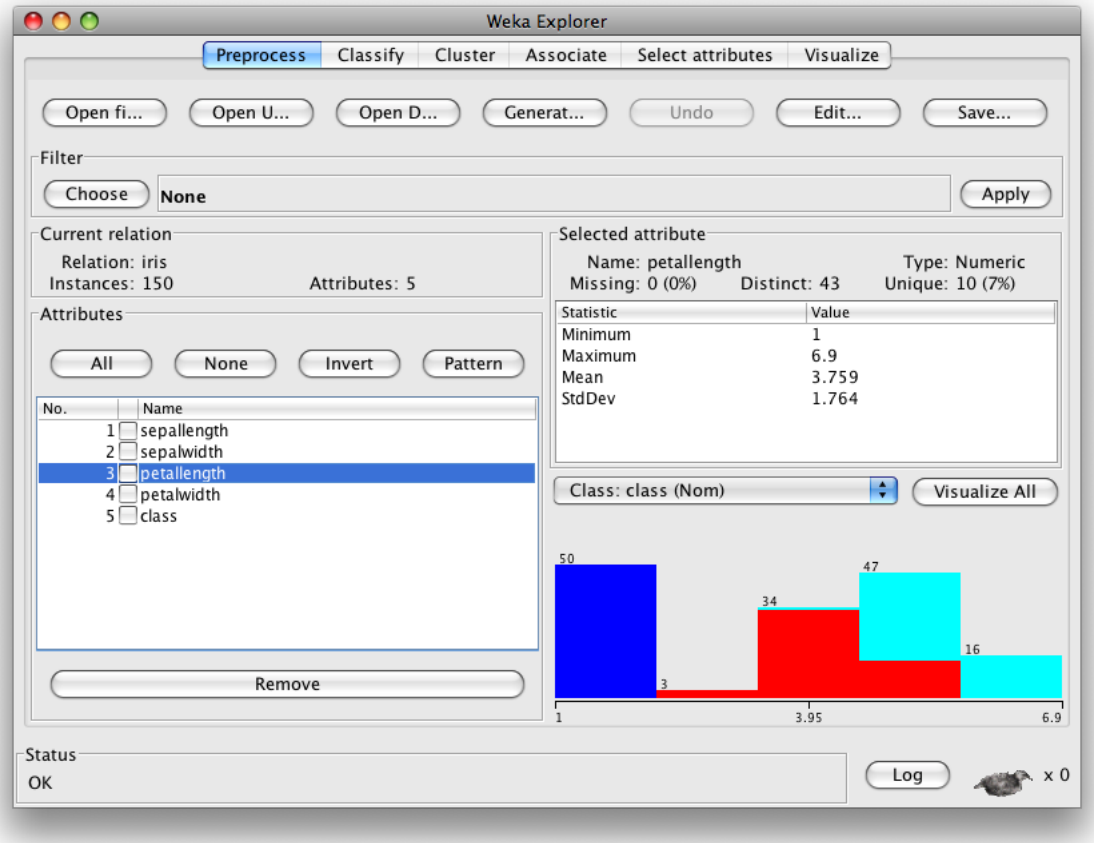


Rysunek 4: Uruchamianie modułów programu Weka

Aby uruchomić program Weka należy odnaleźć na dysku lokalnym komputera folder, w którym został on zainstalowany. W systemie operacyjnym Microsoft Windows XP lub Vista ścieżka dostępu do niego może wyglądać następująco: `C:\Program Files\Weka-3-6`. W folderze tym należy dwukrotnie kliknąć na pliku o nazwie Weka 3-6. Czynność ta spowoduje pojawienie się głównego okna środowiska (*Weka GUI Chooser*), z poziomu którego można uruchamiać jego poszczególne moduły (rys. 4). Wybór przycisku *Explorer* daje dostęp do modułu programu omówionego w pkt 2.2.

W przykładzie omówionym poniżej zaprezentowany zostanie typowy schemat pracy z programem Weka. Zadaniu klasyfikacji podlegać tu będzie zbiór danych rzeczywistych o nazwie *Iris*[4]. Podobnie jak kilka innych zbiorów danych pochodzących z repozytorium UCI (patrz rozdz. 1) jest on dostarczany razem z całym środowiskiem Weka w formacie *ARFF*, a znajduje się w katalogu `$\Data\`), gdzie znak `$` należy zastąpić ścieżką dostępu do folderu z zainstalowanym programem.

[4]http://archive.ics.uci.edu/ml/datasets/Iris

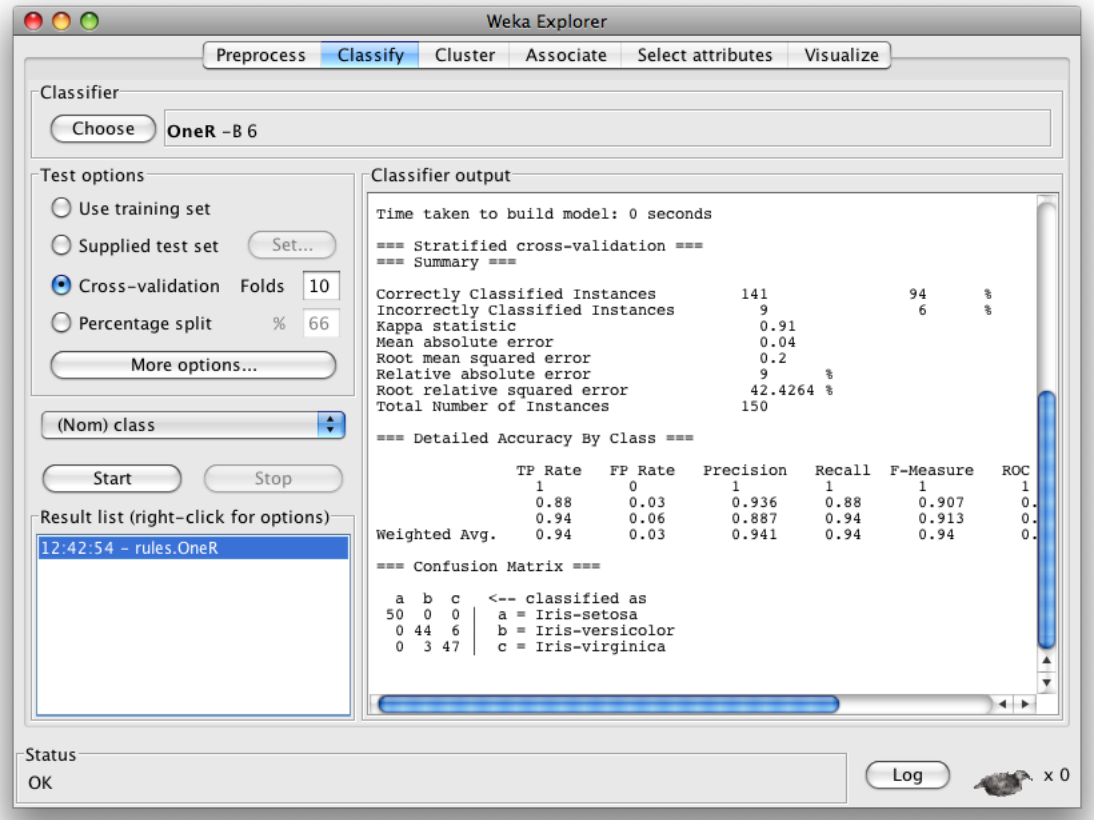


Rysunek 5: Widok modułu *Explorera* po wczytaniu zbioru danych *Iris*

3.2 Podgląd danych

Bezpośrednio po wczytaniu pliku *Iris.arff* aktywne stają się wszystkie zakładki w oknie modułu *Explorera* (rys. 5). Jednocześnie z lewej strony okna wyświetla się lista wszystkich atrybutów określonych dla wczytanego zbioru. W omawianym przykładzie są to cztery cechy numeryczne (*sepallength*, *sepalwidth*, *petallength* i *petalwidth*) oraz atrybut klasy (*class*).

W panelu znajdującym się po prawej stronie okna można natomiast zapoznać się ze szczegółowymi statystykami dotyczącymi wybranego atrybutu (m.in. wartość min., maks., średnia oraz odchylenie standardowe). Można także na wykresie słupkowym, umieszczonym u dołu okna, podejrzeć rozkład wartości danej cechy w poszczególnych klasach. Każda klasa oznaczona jest odrębnym kolorem. Wysokości słupków są proporcjonalne do liczby wektorów danych, dla których wybrana cecha przyjmuje wartości z określonego przedziału. Na podstawie tego histogramu można z grubsza oszacować przydatność pojedycznych atrybutów do dyskryminacji klas. Przycisk *Visualize All* pozwala z kolei wyświetlić rozproszenie wektorów danych we wszystkich możliwych dwuwymiarowych przestrzeniach cech z analizowanego zbioru. Dostęp do tych wykresów możliwy jest także z poziomu zakładki *Visualize*.



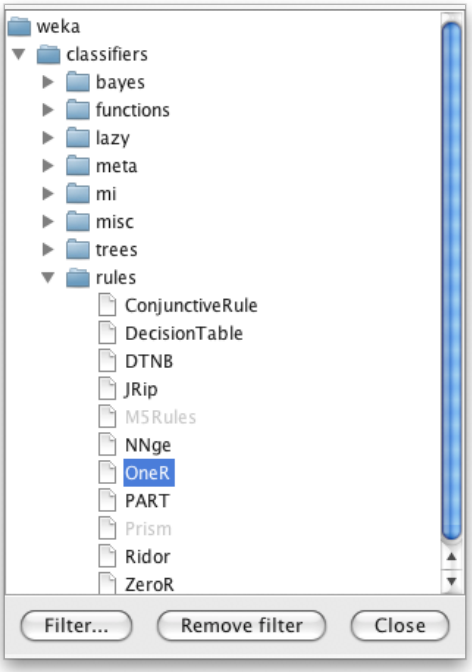
Rysunek 6: Widok zakładki *Classify* w module *Explorera*

3.3 Wykonanie klasyfikacji

Aby przeprowadzić dowolną procedurę klasyfikacyjną należy przełączyć się do zakładki *Classify* (rys. 6). U góry okna znajduje się przycisk *Choose*, służacy do wyboru algorytmu, za pomocą którego zadanie klasyfiakacji zostanie rozwiązane. Wszystkie dostępne algorytmy pogrupowane są w kategorie odpowiadające odmiennym podejściom do klasyfikacji (rys. 7). Nazwa wybranego algorytmu pojawia się w polu tekstowym, sąsiadującym z przyciskiem *Choose*. Pojedyńcze kliknięcie na tym polu otwiera okno dialogowe, w którym można zmieniać parametry wykonania algorytmu.

Po wybraniu algorytmu należy wskazać na jedną z czterech możliwych opcji przeprowadzenia testu nauczonego modelu klasyfkatora:

- *Use training set* – test wykonany będzie na całym zbiorze biorącym udział w uczeniu.
- *Supplied test set* – weryfikacja klasyfikatora zostanie przeprowadzona na dostępnym oddzielnie zbiorze testowym (należy wskazać jego lokalizację).



Rysunek 7: Algorytmy klasyfikacji w programie Weka

- *Cross-validation* – błąd klasyfikatora zostanie oszacowany przy użyciu metody n-krotnej walidacji krzyżowej (należy określić krotność, która domyślnie jest równa 10).
- *Percentage split* – zbiór danych zostanie podzielony na dwie części, z których jedna zostanie wykorzystana do uczenia, a druga do testowania klasyfikatora (należy określić procentowo, jaka część ma zostać przydzielona do fazy treningu, domyślnie 66%).

Uruchomienie procedury klasyfikacji odbywa się po naciśnięciu przycisku *Start*. Wynik obejmujący wytrenowany model klasyfikatora oraz oszacowanie błędu (a także innych miar oceny skuteczności klasyfikacji) zostanie umieszczony po zakończeniu obliczeń w panelu znajdującym się po prawej stronie. Rezultaty kolejnych klasyfikacji są zapamiętywane w buforze programu i można je wyświetlić poprzez wybór odpowiedniej pozycji z listy ukazanej w lewym dolnym rogu okna.

4 Zadania do rozwiązania

4.1 Indukcja reguł decyzyjnych

Zadanie 1

- Za pomocą przycisku *Open file...* wczytaj zbiór danych znajdujący się w pliku *Iris.arff* (katalog `C:\Program Files\Weka-3-6\Data`).

- Za pomocą przycisku *Choose* wybierz algorytm klasyfikacji oparty o regułę jednego atrybutu (*OneR*).
- Jako rodzaj testu klasyfikatora, wskaż opcję *Use training set*.
- Wykonaj klasyfikację ponownie, tym razem przeprowadzając test przy użyciu metody 10-krotnej walidacji krzyżowej (opcja *Cross-validation*).
- Porównaj błędy uzyskane w przypadku obu testów. Które oszacowanie jest bardziej wiarygodne?

Zadanie 2

- Za pomocą przycisku *Choose* wybierz algorytm klasyfikacji *JRip*, w którym zaimplementowano wariant metody przycinania reguł *IREP*.
- Naciśnij jednokrotnie lewym przyciskiem myszy na polu z nazwą wybranego algorytmu klasyfikacji, aby uzyskać dostęp do szczegółowych parametrów wywołania metody. W nowo otwartym oknie dialogowym zmień wartość parametru *usePruning* na *False* (wyłącz przycinanie).
- Wykonaj klasyfikację dwukrotnie – raz testując klasyfikator przy użyciu całego zbioru danych treningowych, drugi raz z użyciem metody walidacji krzyżowej.
- Zmień wartość parametru *usePruning* z powrotem na *True* i powtórz operację z poprzedniego punktu.
- Porównaj błędy klasyfikacji uzyskane dla poszczególnych ustawień parametrów uczenia i testowania. Czy przycinanie reguł daje jakąkolwiek korzyść?

Zadanie 3

- Za pomocą przycisku *Open URL* wczytaj plik z danymi znajdujący się na zdalnym serwerze pod adresem: `http:\\eletel.p.lodz.pl\aklepaczko\open\sonar.arff`.
- Wykonaj klasyfikację dla wczytanego zbioru danych przy użyciu czterech metod:
  - *OneR*,
  - *JRip* z przycinaniem,
  - *JRip* bez przycinania,
  - *Ridor* – konstrukcja reguły z wyjątkami.
- Porównaj uzyskane wyniki uczenia zarówno pod względem poprawności klasyfikacji, jak i stopnia skomplikowalności modelu.

4.2 Indukcja drzew decyzyjnych

Zadanie 4

- Za pomocą przycisku *Choose* wybierz algorytm klasyfikacji *J48*, w którym zaimplementowano metodę konstruowania drzew decyzyjnych z przycinaniem.
- Przeprowadź klasyfikację z oszacowaniem błędu rzeczywistego za pomocą metody walidacji krzyżowej.
- Naciśnij jednokrotnie lewym przyciskiem myszy na polu z nazwą wybranego algorytmu klasyfikacji, aby uzyskać dostęp do szczegółowych parametrów wywołania metody. W nowo otwartym oknie dialogowym zmień wartość parametru *reducedErrorPruning* na *True* (przycinanie drzewa redukujące błąd).
- Ponownie przeprowadź klasyfikację.
- W panelu znajdującym się na dole po lewej stronie okna (*Result list*) wybierz pierwszy rezultat klasyfikacji i kliknij na nim prawym klawiszem myszy, aby uzyskać dostęp do opcji szczegółowych. W otwartym menu kontekstowym wybierz polecenie *Visualize tree*. Powtórz tę operację dla wyniku drugiego doświadczenia. Jeżeli któreś z uzyskanych drzew nie jest dobrze widoczne w swoim oknie, należy okno to powiększyć a następnie dopasować rysunek drzewa do wymiarów okna (polecenie *Fit to Screen* w menu kontekstowym).
- Porównaj błędy klasyfkacji i rozmiar drzew decyzyjnych otrzymanych w obu przypadkach.

Zadanie 5

- Za pomocą przycisku *Open file...* wczytaj zbiór danych znajdujący się w pliku *segment-challenge.arff* (katalog `C:\Program Files\Weka-3-6\Data`).
- Za pomocą przycisku *Choose* wybierz algorytm klasyfikacji *UserClassifier*, który pozwala na w sposób interaktywny samodzielne konstruować drzewa decyzyjne.
- Zaznacz opcję *Use training set* jako metodę testowania.
- Uruchom procedurę budowania drzewa naciskając przycisk *Start*. Nowo otwarte okno zawiera dwie zakładki – *Tree Visualizer* oraz *Data Visualizer*. W pierwszej z nich widoczne jest drzewo na danym etapie konstruowania, w drugiej przedstawione są wektory danych we wskazanej dwuwymiarowej przestrzeni cech w zaznaczonym węźle drzewa.
- Skonstruuj własne drzewo decyzyjne przy użyciu narzędzi zaznaczania i przycisku *Submit* dostępnych w zakładce *Tree Visualizer*.

Łódź 2009

Zredagowane w LaTeX-u