



Inżynieria wiedzy i uczenie maszynowe

Konspekt zajęć laboratoryjnych
prowadzonych w Katedrze Informatyki
Studia Drugiego Stopnia
Drugi rok

Bartłomiej Śnieżyński

Laboratorium nr 5

Temat

System Weka – Moduł Knowledge Flow

Cel

Celem zajęć jest zapoznanie się z modułem Knowledge Flow w systemie Weka.

Wymagane wiadomości wstępne z wykładu

Problem klasyfikacji

Konfiguracja komputera

Podczas laboratorium wykorzystywany będzie system Weka.

Linki

<http://www.cs.waikato.ac.nz/ml/weka/>

<http://archive.ics.uci.edu/ml/>

Uwaga! Jeśli są problemy z dostępem do strony UCI można użyć proxy, np.

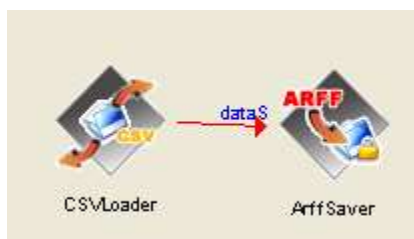
<https://www.proxysite.com>

Plan laboratorium

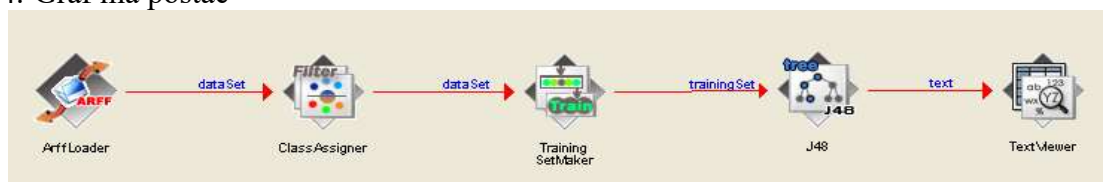
1. Uruchomić system Weka
2. Uruchomić moduł Knowledge Flow
3. Zbudować graf konwertujący pliki .data do formatu ARFF
 - 3.1. Wybrać zestaw danych z [UCI](#) (w przypadku problemów z dostępem por. str. 2) lub zbudować swój własny w formacie CSV
 - 3.2. Pierwszy wiersz powinien zawierać nazwy atrybutów, np.:

	car.csv	test_car.arff
1	buying,maint,doors,persons,lug_boot,safety,class	
2	vhigh,vhigh,2,2,small,low,unacc	
3	vhigh,vhigh,2,2,small,med,unacc	
4	vhigh,vhigh,2,2,small,high,unacc	
5	vhigh,vhigh,2,2,med,low,unacc	
6	vhigh,vhigh,2,2,med,med,unacc	
7	vhigh,vhigh,2,2,med,high,unacc	
8	vhigh,vhigh,2,2,big,low,unacc	
9	vhigh,vhigh,2,2,big,med,unacc	
10	vhigh,vhigh,2,2,big,high,unacc	
11	vhigh,vhigh,2,4,small,low,unacc	

- 3.3. Wstawić wierzchołki CSVLoader i ArffSaver
- 3.4. Połączyć CSVLoader z ArffSaver (PKlik na CSVLoader/dataSet, LKlik na ArffSaver)
- 3.5. Graf ma postać:



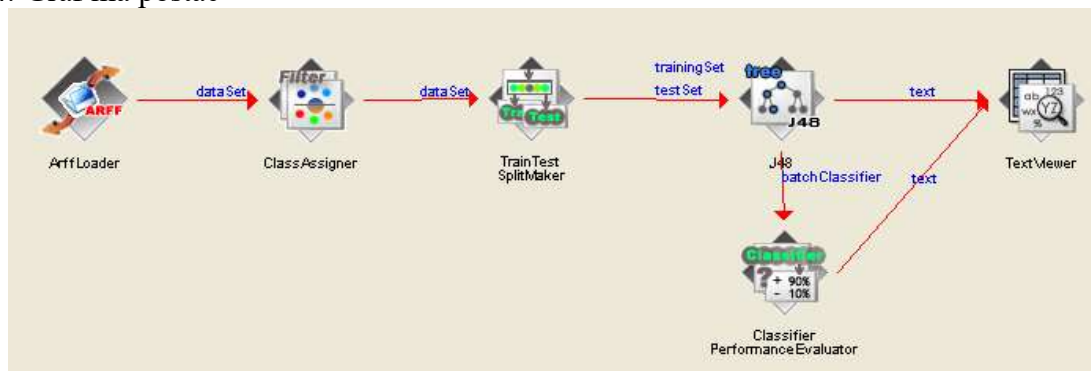
- 3.6. Oba wierzchołki należy skonfigurować (PKlik/Configure)
- 3.7. Uruchomić CSVLoader (PKlik/Start loading)
- 3.8. Oglądnąć plik wynikowy
4. Zbudować graf tworzący drzewo decyzyjne przy użyciu algorytmu J48
 - 4.1. Po załadowaniu danych trzeba wstawić Filtr ClassAssigner, który wskazuje atrybut oznaczający kategorię.
 - 4.2. Za nim wstawiamy wierzchołek TrainingSetMaker typu Evaluation przygotowujący zestaw danych treningowych i podpinamy go do klasyfikatora J48.
 - 4.3. Wynik tekstowy podpinamy do TextViewer.
 - 4.4. Graf ma postać



- 4.5. Uruchamiamy przetwarzanie
- 4.6. Oglądamy rezultaty (PKlik na TextViewer/Show results)
5. Przerobić powyższy graf tak by przetestować utworzony klasyfikator
 - 5.1. Zastąpić TrainingSetMaker wierzchołkiem TrainTestSplitMaker.
 - 5.2. Oba wyjścia połączyć z klasyfikatorem

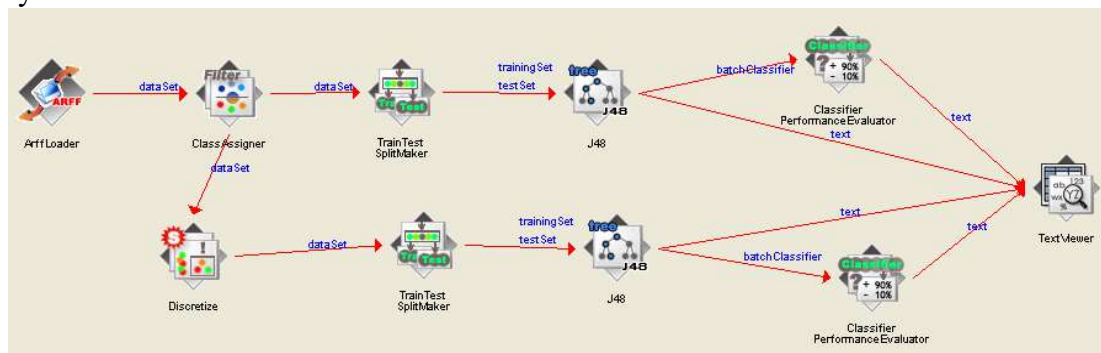
5.3. Dodać ClassifierPerformanceEvaluator i podpiąć do niego batchClassifier z J48, a wyjście tekstowe skierować do TextViewer

5.4. Graf ma postać



5.5. Uruchomić i oglądać wyniki

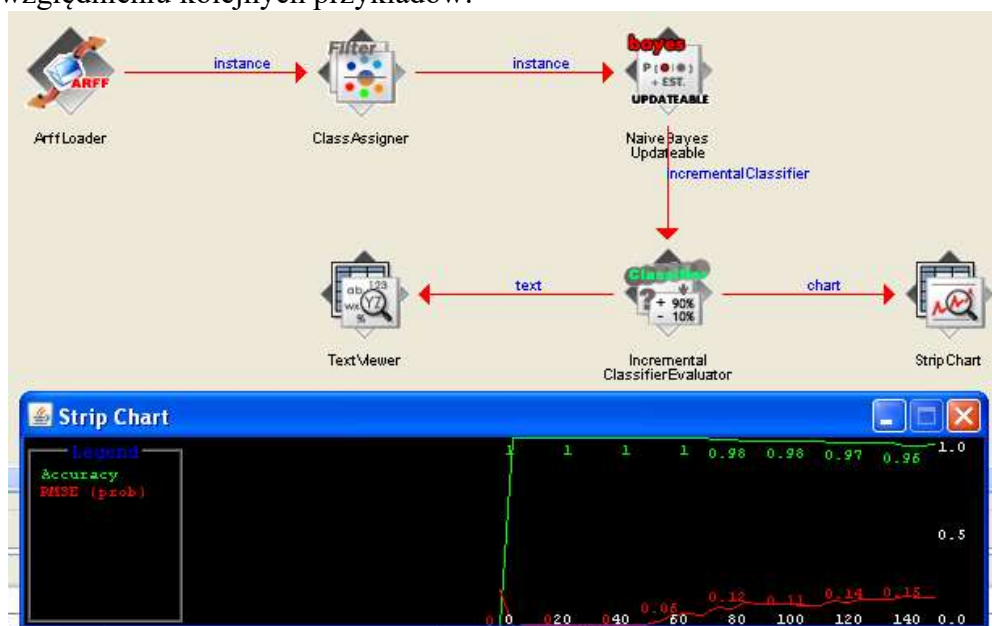
6. Porównać powyższy klasyfikator z klasyfikatorem działającym na zdyskretyzowanych danych:



7. Dodać wizualizację graficzną klasyfikatorów i porównać oba nauczone klasyfikatory.

8. Porównać efektywność algorytmów J48, Naive Bayes, JRipp

9. Użyć uczenia inkrementacyjnego i zwizualizować jak zmienia się efektywność klasyfikatora po uwzględnieniu kolejnych przykładów:



10. Sprawdzić wpływ parametrów wierzchołków IncrementalClassifierEvaluator i StripChart na wygląd wykresu.