Машинно обучение

Лабораторно упражнение № 10

Инструментални средства за машинно обучение

Упражнението демонстрира работа с инструменталното средство за машинно обучение Weka. Целта е въведение в графичния интерфейс на програмата Weka, с нейна помощ да се класифицират данни чрез различни алгоритми, да се визуализират резултатите, да се сравни точността на обучението.

Weka (Waikato Environment for Knowledge Analysis) е софтуер написан на езика Java и създаден в университета Waikato в Нова Зеландия, разпространяващ се под лиценз GNU General Public License version 21 (GPLv2). Програмата съдържа колекция от алгоритми за машинно обучение на задачи за извличане на данни. Алгоритмите могат да бъдат приложени директно към набор от данни или да бъдат извикани от Java код. Weka съдържа инструменти за предварителна обработка на данни, класификация, регресия, групиране, правила за асоцииране и визуализация[1].

1. Въведение в графичния интерфейс на Weka

Фигура 1

След стартиране на програмата WEKA се предоставя избор на приложения. (Фигура 1)

- Explorer - среда за изследване на данни с WEKA;
- Experimenter - среда за провеждане на експерименти и сравнителни статистически тестове между схеми за обучение;

[1] https://www.cs.waikato.ac.nz/ml/weka/

- KnowledgeFlow - среда поддържаща същата функционалност, като Explorer, но с интерфейс "плъзгане и пускане", предоставяща постепенно обучение;
- Workbench - среда, която съчетава всички интерфейси на GUI в един интерфейс;
- SimpleCLI - интерфейс на команден ред, посредством който се изпълняват директно WEKA команди.

2. Графичния интерфейс Explorer

- Preprocess – предварителна обработка на данни
- Classify – обучаване и тестване на алгоритми за класификация и регресия
- Cluster – клъстеризация на данни
- Associate – асоцииране чрез правила на данни
- Select attributes - избор на атрибути
- Visualize – визуализация

3. Набор от данни

Набор от данни (dataset), е еквивалент на двуизмерна електронна таблица или таблица от база данни. В WEKA се изпълнява от класа weka.core.Instances. Наборът от данни е колекция от примери, всяка от класовете weka.core.Instance. Всеки инстанция се състои от няколко атрибута, всеки от които може да бъде номинален, цифров (реален или цял), дата, релационен или низ.

Външното представяне на клас Instances е ARFF (Attribute-Relation File Format) файла, който е ASCII текстов файл. В папка data (подпапка на Weka), са предоставени текстови файлове с разширение arff.

Следващите редове са част от файла iris.arff, който се намира в папка data. Това е известен набор от данни, създаден през 1998 година от R.A. Fisher, използван за класифициране цветовете на ириса между три вида (setosa , versicolor или virginica). Наборът съдържа 150 броя инстанции, по 50 от всеки клас.

% 1. Title: Iris Plants Database

% 2. Sources:

% (a) Creator: R.A. Fisher

% (b) Donor: Michael Marshall (MARSHALL%PLU@io.arc.nasa.gov)

% (c) Date: July, 1988

% …

% 9. Class Distribution: 33.3% for each of 3 classes.

@RELATION iris

@ATTRIBUTE sepallength REAL

@ATTRIBUTE sepalwidth REAL

@ATTRIBUTE petallength REAL

@ATTRIBUTE petalwidth REAL

@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA

5.1,3.5,1.4,0.2,Iris-setosa

4.9,3.0,1.4,0.2,Iris-setosa

…

7.0,3.2,4.7,1.4,Iris-versicolor

6.4,3.2,4.5,1.5,Iris-versicolor

…

6.3,3.3,6.0,2.5,Iris-virginica

5.8,2.7,5.1,1.9,Iris-virginica

…

Редовете започващи с %, са коментари, съдържащи обикновено информация за авторите и съдържанието на набора от данни. Секция @RELATION дава наименованието на задачата. Следва секция @ATTRIBUTE, която описва признаците: техните названия и типове. В секция @DATA са предоствани самите данни.

4. Задачи

Задача 1: Класифициране на цветовете на ириса.

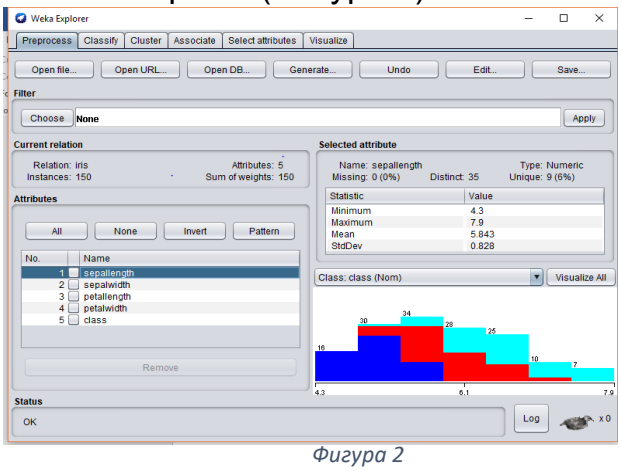
Последователност от действия:

1. Стартиране на WEKA
2. Стартиране на Preprocess

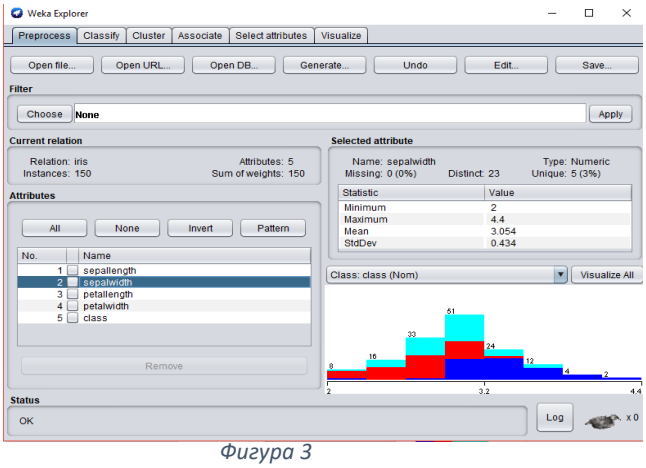
3. Open file: iris.arff

4. Последователно изберете всеки от атрибутите: sepalenghth, sepalwidth, petallength, petalwidth, class и разгледайте получените хистограми. (Фигура 2, Фигура 3, Фигура 4, Фигура 5, Фигура 6)

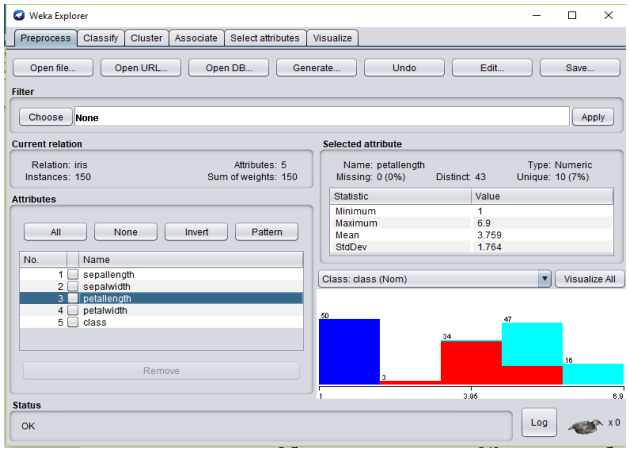
5. Натиснете бутона Visualize All за да визуализирате всички хистограми.(Фигура 7)



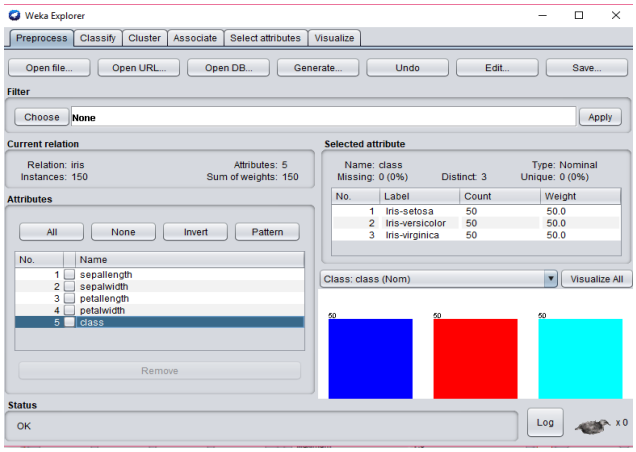
Фигура 2



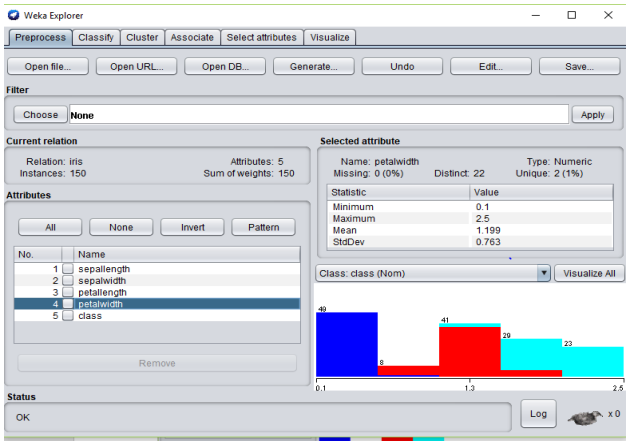
Фигура 3



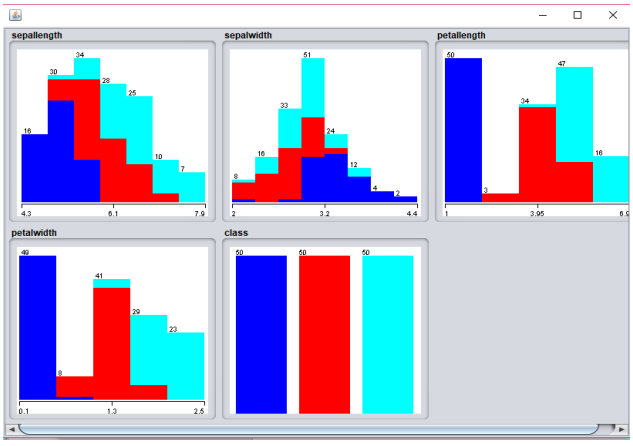
Фигура 4



Фигура 5



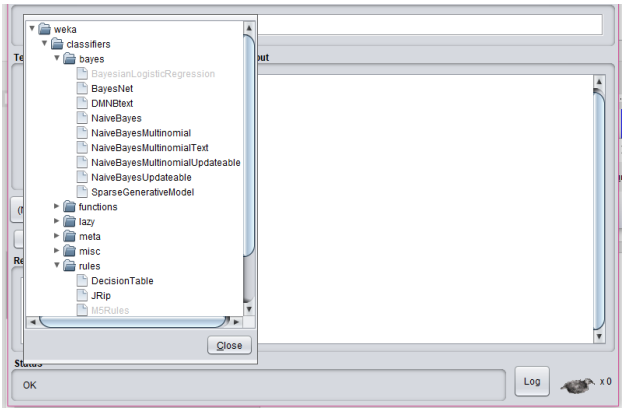
Фигура 6



Фигура 7

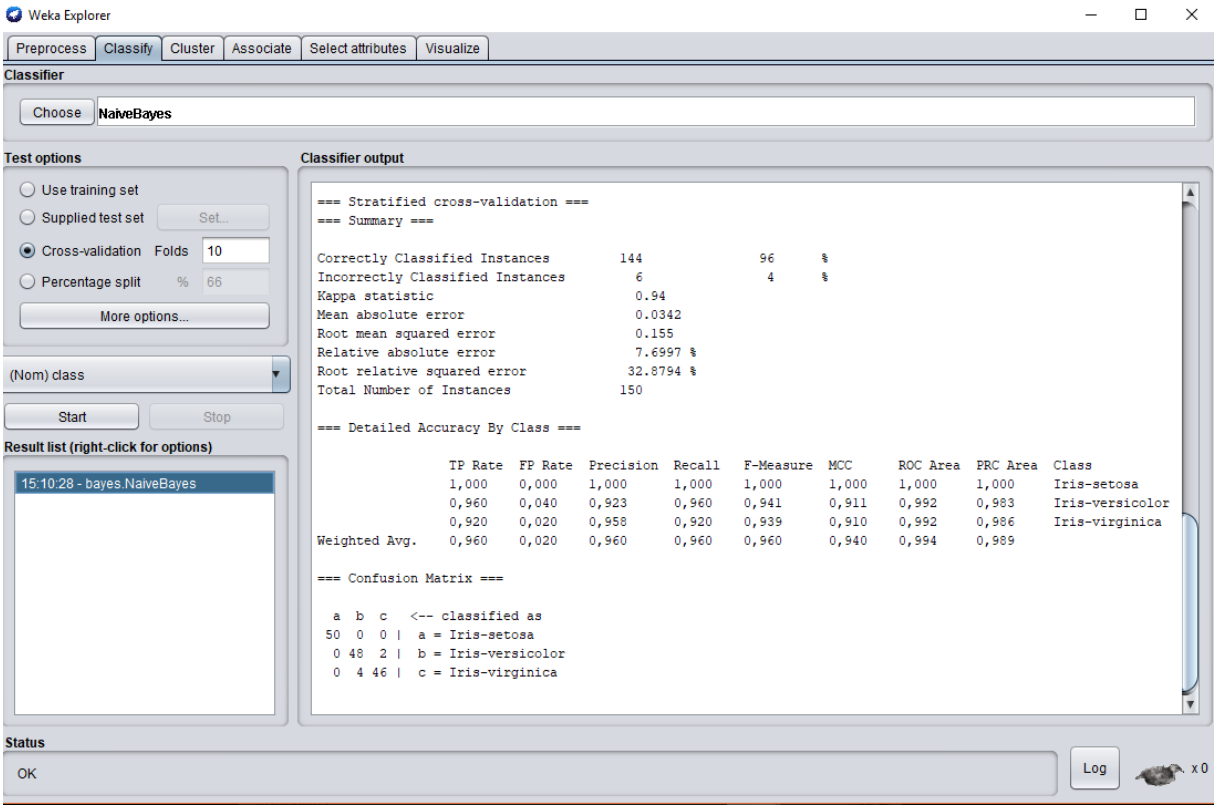
6. Стартиране на Classify

7. Choose -> bayes->NaiveBayes (Фигура 8)



Фигура 8

8. Натиснете бутона Start (Фигура 9)



Фигура 9

Резултатите показват (Фигура 9), че правилно са класифицирани 144 екземпляра от възможни 150. Видно от матрица: Confusion Matrix, 2 екземпляра от клас Iris – versicolor са класифицирани като Iris - virginica, а 4 екземпляра от Iris - virginica са некласафицирани неправилно като Iris – versicolor.

Класификационният модел разпределя съответния обект в един от възможните класове вярно (True) или невярно (False). При този модел са възможни четири варианта на класификация:

- True Positive (TP) – действителен клас Positive и вярно класифициран като клас Positive;

$$TP\ Rate = \frac{TP}{TP+FN} \quad (1)$$

- True Negative (TN) – действителен е клас Negative и вярно класифициран като клас Negative;

$$TN\ Rate = \frac{TN}{TN+FP} \quad (2)$$

- False Positive (FP) - действителен е клас Negative, а невярно класифициран като клас Positive;

$$FP\ Rate = \frac{FP}{FP+TN} \quad (3)$$

- False Negative (FN) - действителен е клас Positive, а невярно класифициран като клас Negative.

$$FN\ Rate = \frac{FN}{FN+TP} \quad (4)$$

Прецизност (Precision)

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

Recall

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

Мярката F-measure

$$F\ measure = \frac{2}{{}^{1}/_{Recall} + {}^{1}/_{Precision}} \quad (7)$$

Точност (Accuracy)

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (8)$$

Матрица на грешките (Confusion matrix) – квадратна матрица чиито редове и колони, отговарят на класовете. Редовете на матрицата представляват действителните екземпляри от съответния клас, а колоните класовете, получени след прилагането на класификационния модел върху тестовите данни.

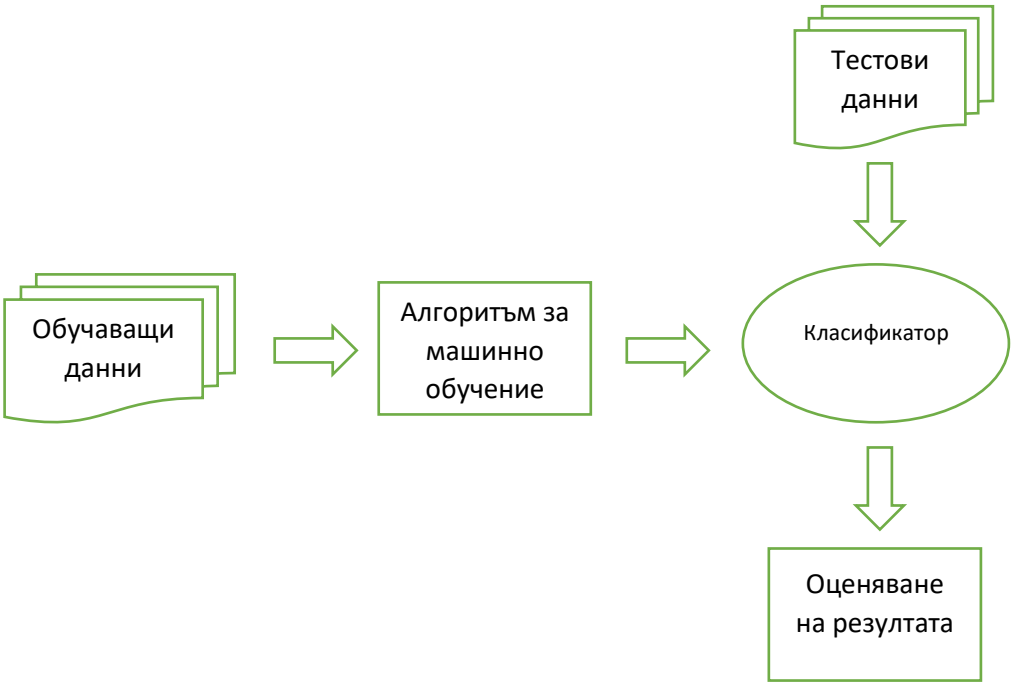
9. Продължете класифицирането с различни алгоритми:
    a. NaiveBayesMultinomial
    b. lazy -> IBK – това е Метода на К най-близкия съсед
    c. functions -> SMO – това е Метода на опорните вектори
    d. tree ->J48 (вивуализирайте дървото като щракнете с десен бутон на мишката върху J48 и изберете Visualize tree)
    e. тествайте и с други различни алгоритми
10. Сравнете получените резултати.

Задача 2: Разпознаване на изображения

Използвайте файл segment-challenge.arff, представляващ комплект от данни от 1500 копия, разделени в 7 класа с 19 атрибута. Класифицирайте

с различни алгоритми, като променяте някои от характеристиките, като например: броя K на Метода на K най-близкия съсед, или ядрото на Метода на опорните вектори, или броя клони на дърво. За промяна на характеристиките се щраква с ляв бутон на мишката върху вече избрания алгоритъм, чието име след избора се е появило вдясно на бутона Choose.

Процесът на класифициране е представен на Фигура 10.



Фигура 10

Тествайте като използвате крос валидация (Cross –validation), при който данните се разделят на 10 части (10 е число по подразбиране и може да се променя), една десета се използват за тестови данни, а 9/10 за обучителни, след това се взема втора десета за тестови данни, а новите 9/10 за обучителни, като този процес се повтаря 10 пъти. Полученият резултат е осреднен.

Тествайте чрез Supplied test set, като за тестови данни изберете файла segment-test.arff. Обикновено един набор от данни се разделя на две части – обучителни и тестови. Обучителните данни може да са 2/3 от целия набор, а тестовите - 1/3 (съотношението може да е 10:1 или 4:3).

Тествайте и чрез Percentage split, като за обучителни данни изберете 66%, които са зададени по подразбиране, като останалите 34% ще са тестови.

Сравнете получените резултати.

5. Други известни инструментални средства

- Библиотека за машинно зрение OpenCV - http://opencv.org/
- Система за статистически изчисления R - http://www.r-project.org/

- Система за статистически изчисления RapidMiner - https://rapidminer.com/
- Пакет за решаване на задачи на машинното обучение и анализ на данни Orange - http://orange.biolab.si/

6. **Еталонни данни за изследователи:** UCI Machine Learning Repository
http://archive.ics.uci.edu/ml/