Data Mining

Lab 4: Clasificarea datelor

Sumar:

- Modele probabiliste
- Clasificatori bazati pe retele neuronale
- Clasificare utilizand vectori support

1. Modelele probabiliste

Permit estimarea probabilității ca o dată să aparțină unei clase ($P(C_k|(a_1,a_2,\ldots,a_n))$). Clasa corespunzătoare datei este cea pentru care probabilitatea este maximă.

Variante:

- Model Bayesian simplu (Naïve Bayes): presupune ca atributele nu sunt correlate (probabilitatea observării unei date este produsul probabilităților de a observa fiecare dintre valorile atributelor).
- Rețea cauzală (Bayesian Net sau Belief network): permite descrierea unui model sub forma unui graf orientat in care sunt specificate relații de cauzalitate

Exercițiul 1:

a) Deschideti fișierul "weather.nominal.arff". Aplicați clasificatorii Naïve Bayes și Bayes Net

b) Utilizați Bayes Network Editor (de la Weka GUI Choser -> Tools) pentru a construi o rețea bayesiană. Etape:
   - Selectare set date: Tools->Set Data
   - Plasare noduri (fiecare atribut, inclusiv cel de clasă va avea asociat un nod) folosind Edit->Add node
   - Specificare relații folosind Edit->Add arc (de exemplu: se plasează arc de la nodul corespunzător clasei la fiecare dintre nodurile associate atributelor)
   - Antrenare rețea (Tools->Learn network (invătarea structurii), Tools->Learn CPT (determinarea tabelelor de probabilitati))
   - Utilizare: se selectează valorile corespunzătoare instanței ce urmează să fie clasificată (folosind click dreapta pe fiecare nod asociat unui atribut și Set evidence); decizia se bazează pe valorile probabilităților corespunzătoare nodului asociat clasei

Obs: detalii privind utilizarea rețelelor bayesiene se găsesc la http://www.cs.waikato.ac.nz/~remco/weka_bn/

2. Rețele neuronale

Etape in proiectarea unei retele neuronale pentru clasificare:

- Stabilirea arhitecturii. In cazul rețelelor multi-nivel (multi-layer perceptrons) se fixează:
  - Număr de unitati de intrare = numarul atributelor
  - Număr de nivele/unitati ascunse – depinde de complexitatea problemei
  - Număr de unități de ieșire:

        - Clasificare binară: o unitate (rezultatul se interpretează folosind o valoare prag) sau două unități (unitatea care produce cea mai mare valoare va indica clasa)
- Stabilirea functiilor de activare. Pentru rețelele antrenate folosind algoritmul Backpropagation:
    - Unitățile ascunse au funcții de activare de tip sigmoidal (functia logistică sau tanh)
    - Unitățile de ieșire au funcții de activare de tip sigmoidal sau funcții liniare
- Alegerea algoritmului de antrenare și a parametrilor acestuia. In cazul algoritmului Backpropagation parametrii uzuali de control sunt:
    - Numărul de epoci de antrenare
    - Rata de învățare
    - Coeficientului termenului de tip „moment”

Implementare Weka: MultilayerPerceptron = rețea feedforward antrenată cu Backpropagation (variant cu moment); arhitectura rețelei este implicit cu un nivel de unități ascunse; numărul de unități

Exercițiul 2: Analiza influentei arhitecturii rețelei asupra performanțelor clasificatorului

a) Deschideti fișierul “breast-w.arff”
b) Antrenați o rețea neuronală folosind setările standard (un nivel ascuns cu număr de unități ascunse cu K=(nr atribute+nr clase)/2 unitati)
c) Comparați performanțele rețelei pentru următoarele valori ale parametrului Hidden Layers (H):
    a. ‘a’ (K=(nr atribute+nr clase)/2)
    b. ‘i’ (K=nr attribute)
    c. ‘o’ (K=nr clase)
    d. ‘t’ (K = nr atribute+nr clase)
    e. 4,2 (două nivele ascunse cu 4, respectiv 2 unități)

    tAnalizați influența ratei de învățare si a coeficientului corespunzător termenului “moment”.
d) Comparați rezultatul obținut folosind MultilayerPerceptron cu cel obținut folosind o rețea de tip RBF – radial basis function (functions->RBF Network)

Obs. Prin activarea opțiunii GUI de la Multilayer Perceptron poate fi vizualizată arhitectura rețelei.

3. Clasificare folosind vectori suport (Support Vector Machines)

Varianta de clasificatori bazati pe vectori suport implementată in Weka folosește un algoritm rapid pentru rezolvarea problemei de optimizare (Sequential Minimal Optimization). In cazul clasificării in M clase problema se transformă în mai multe problem de clasificare binară (corespunzătoare perechilor de clase)

Exercițiul 3:

a) Deschideți in Weka fișierul “breast-w.arff” si utilizati SVM (functions->SMO) ; analizați influența funcției nucleu asupra performanțelor clasificatorului (variante:)
b) Deschideți in Weka fișierul “arrhythmia.arff” și efectuați aceeași prelucrare; incercați să rezolvați aceeași problemă utilizând o rețea neuronală.