COMP4702/COMP7703 - Machine Learning

Prac 8 – Support Vector Machines

Aims:

- To complement lecture material in understanding the operation of SVMs.
- To gain experience with simulating these techniques in software.
- To experimentally compare a range of classifiers studied in this course.
- To produce some assessable work for this subject.

Procedure:

Support Vector Machines

In Chapter 10 of the Alpaydin text, Support Vector Machines are introduced in the context of linear discriminative models. Weka contains a couple of SVM implementations, the basic version being "SMO" under "classifiers->functions".

SMO stands for Sequential Minimal Optimization and refers to a particular algorithm for training SVMs. If you left-click on this model you will see that there are several user parameters – the default settings implement a linear support vector machine (i.e. a linear kernel function) as described in lectures. The key "user" parameters in an SVM are:

- **C** – the penalization factor/regularization factor, which controls the influence of errors (in terms of the maximal margin hyperplane) on the overall objective function for training.
- The kernel function and any associated parameters. In WEKA, you can use either a polynomial kernel or a radial basis function (RBF) kernel. A linear kernel function is a special case of a polynomial with the order of the polynomial equal to 1.

Note also that this implementation of an SVM solves multiclass (>2) classification problems by training a two class SVM on each distinct pair of classes (see lecture notes on pairwise separation and Section 10.4 of the Alpaydin book). The implementation also normalizes each attribute in the data set before training.

→ **Q1:** Using the default settings, train an SVM on the Diabetes (from Prac 4), Sonar and Ionosphere datasets (available on the course web page, details below and available from the UCI machine learning repository). Use the default training/test set percentage split in Weka. Record your results (classification rate and confusion matrix).

→ **Q2:** Select one of the datasets from Q1 (your choice!). Using this dataset, experiment with the SVM kernel function by trying at least 3 different parameter value settings for both polynomial and RBF kernels (in addition to your answer to Q1). Record your results as in Q1.

→ **Q3:** Using the same dataset with the parameter/kernel setting that gave the best performance in Q2, experiment by varying the C parameter (which has value 1.0 by default). Try C=0.01, 0.1, 10 and 100. Use Matlab to plot the performance (percentage correct) on the test set.

→ **Q4:** Repeat your experiments from Q3, but this time use 10-fold cross validation to estimate (generalization) performance, rather than the default percentage split option in WEKA. Plot these results on the same set of axes as your results from Q3.

Ionosphere Data set

```
Input features      = 34
No. of classes      = 2
No. of samples= 351
       Training samples    = 264 (Class 1 samples=174, Class 2 samples=89)
       Test samples        =   87 (Class 1 samples=  50, Class 2 samples=37)
```

More information about the data set:

- Source Information:
  Donor: Vince Sigillito (vgs@aplcen.apl.jhu.edu)
  Date: 1989
  Source: Space Physics Group, Applied Physics Laboratory
  Johns Hopkins University, Johns Hopkins Road, Laurel, MD 20723
- Relevant Information:
  This radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. See the paper for more details. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not; their signals pass through the ionosphere.
  Received signals were processed using an autocorrelation function whose arguments are the time of a pulse and the pulse number. There were 17 pulse numbers for the Goose Bay system. Instances in this databse are described by 2 attributes per pulse number, corresponding to the complex values returned by the function resulting from the complex electromagnetic signal.
- Attribute Information:
  All 34 are continuous, as described above
  The 35th attribute is either "good" or "bad" according to the definition summarized above. This is a binary classification task.
- Missing Values: None