CS659 Spring 2015

Homework #2

Due March 17th, in-class.

1. For this problem, use the Breast Cancer Dataset that you will find here. You will use the following algorithms to classify this data (use Cross-validation):
    a. Decision Stump
    b. J48 Unpruned
    c. J48 Pruned
    d. K-Nearest neighbors

    Report your findings (measures, confusion matrix) in each case. Compare them. What is causing most of the errors in the Decision Stump? What is causing most of the errors in the Unpruned Decision Tree? What happened when you allowed the tree to get pruned?

2. For this problem, use the churning data. Observe the “defector” attribute: what do you notice?
    a. Using J48 (pruned) and cross-validation produce a classification model for this data. What is noticeable in the data? Is this model good?
    b. Using undersampling, oversampling, and Smote (separately) balance the data and reapply J48 (pruned) to the data. What do you notice? Report your findings.
3. For this problem use the train_mnist_clean_best.arff dataset. This comes from a collection of 60,000 train and 10,000 test samples of handwritten digits. The samples are partitioned (nearly) evenly across the 10 different digit classes {0, 1, . . . , 9}. Each sample is a 28×28 pixel image containing one digit. The digits are scaled to fit a 20×20 pixel box, which is then positioned in the 28 × 28 image by placing its center of mass to the center of the image. For further details on how the digits are normalized and centered to create the final images refer to the MNIST webpage (http://yann.lecun.com/exdb/mnist/). The images are grayscale, with each pixel taking values in {0, 1, . . . , 255}, where 0 corresponds to black (weakest intensity) and 255 corresponds to white (strongest intensity). Therefore, our dataset is a N × 784 dimensional matrix where each dimension corresponds to a pixel from the image and N is the number of images. To save you time, we have cleaned the data and selected the best attributes.
    a. Assess the “usefulness” of the features using the attribute evaluator InfoGainAttributeEval (in the Select Attributes tab). How does this “attribute evaluator” work? Go to the Visualize tab and look atthe plots of the attributes against the class variable for the highest ranked attributes and the lowest ranked ones. What do you notice? Remove all the attributes that have 0 information gain. Is it safe to perform this operation? Save the reduced dataset.
    b. Perform PCA on the dataset and retain 90% of the variance. Save the reduced dataset.
    c. Install the lib-SVM implementation in your Weka package, using the instructions found here). Be sure to start Weka using the command “java -classpath "%CLASSPATH%;weka.jar;libsvm.jar" weka.gui.GUIChooser” from the console.
    d. For each of the datasets (the one we provide, the ones you obtained in Steps a. and b.) use SVM to build a classifier. Retain 20% of the data for test purposes (not to be used in the building of the classifier). Use cross-validation on the rest of the set to find the best parameters for SVM (grid search). Report your results.