Homework 2
CS 436/580L: Introduction to Machine Learning

Instructor: Arti Ramesh

Instructions

1. You can use either C/C++, Java or Python to implement your algorithms.
2. **Your implementations should compile on remote.cs.binghamton.edu.**
3. Make sure remote.cs.binghamton.edu has the packages that you require before starting to implement.
4. This homework requires you to **implement** post-pruning in Decision Trees and Naive Bayes. Using existing packages for post-pruning or naive bayes is not allowed.
5. Your homework should contain the following components:
   (a) README.txt file with detailed instructions on how to compile and run the code.
   (b) Code source files
   (c) Type-written document containing the results on the datasets.
6. Submit the homework as a **single zip file:** $firstname\_lastname\_hw2.zip$.

1 Post-pruning in Decision Trees (45 points)

- Implement the post pruning algorithm given below as Algorithm 1 (See also Mitchell, Chapter 3).
- Once we compile your code, we should be able to run it from the command line. Your program should take as input the following six arguments:

```
.\program <L> <K> <training-set> <validation-set> <test-set> <to-print>
L: integer (used in the post-pruning algorithm)
K: integer (used in the post-pruning algorithm)
to-print:{yes,no}
```

Algorithm 1: Post Pruning

```
Input: An integer L and an integer K
Output: A post-pruned Decision Tree
begin
    Build a decision tree using all the training data. Call it $D$;
    Let $D_{Best} = D$;
    for $i = 1$ to $L$ do
        Copy the tree $D$ into a new tree $D'$;
        $M$ = a random number between 1 and $K$;
        for $j = 1$ to $M$ do
            Let $N$ denote the number of non-leaf nodes in the decision
            tree $D'$. Order the nodes in $D'$ from 1 to $N$;
            $P$ = a random number between 1 and $N$;
            Replace the subtree rooted at $P$ in $D'$ by a leaf node. Assign
            the majority class of the subset of the data at $P$ to the leaf
            node.;
            /* For instance, if the subset of the data at $P$
               contains 10 examples with $class = 0$ and 15
               examples with $class = 1$, replace $P$ by $class = 1$   */
        end
        Evaluate the accuracy of $D'$ on the validation set;
        /* accuracy = percentage of correctly classified
           examples                                                  */
        if $D'$ is more accurate than $D_{Best}$ then
            $D_{Best} = D'$;
        end
    end
    return $D_{Best}$;
end
```

It should output the accuracies on the test set for decision trees constructed using the two heuristics as well as the accuracies for their post-pruned versions for the given values of L and K. If to-print equals yes, it should print the decision tree in the format described above to the standard output.

- On the two datasets available on myCourses:
  Choose 10 suitable values for L and K (not 10 values for each, just 10 combinations). For each of them, report the accuracies for the post-pruned decision trees constructed using the both the heuristics (information gain and variance impurity) on both test datasets.

2 Naive Bayes for Text Classification

In this question, you will implement and evaluate Naive Bayes for text classification.

0 Points Download the spam/ham (ham is not spam) dataset available on myCourses. The data set is divided into two sets: training set and test set. The dataset was used in the Metsis et al. paper [1]. Each set has two directories: spam and ham. All files in the spam folders are spam messages and all files in the ham folder are legitimate (non spam) messages.

40 points Implement the multinomial Naive Bayes algorithm for text classification described here: `http://nlp.stanford.edu/IR-book/pdf/13bayes.pdf` (see Figure 13.2). Note that the algorithm uses add-one laplace smoothing. Make sure that you do all the calculations in log-scale to avoid underflow. Use your algorithm to learn from the training set and report accuracy on the test set.

Extra Credit 20 points Improve your Naive Bayes by throwing away (i.e., filtering out) stop words such as “the” “of” and “for” from all the documents. A list of stop words can be found here: `http://www.ranks.nl/resources/stopwords.html`. Report accuracy for Naive Bayes for this filtered set. Does the accuracy improve? Explain why the accuracy improves or why it does not?

What to Turn in

- Your code
- **(5 points)** README file for compiling and executing your code.
- **(10 points)** A detailed write up that contains:
  1. The accuracy obtained on the test set for different values of L and K for the post-pruned version of decision tree.
  2. The accuracy on the test set for Naive Bayes algorithm.

References

[1] V. Metsis, I. Androutsopoulos and G. Paliouras, “Spam Filtering with Naive Bayes - Which Naive Bayes?”. Proceedings of the 3rd Conference on Email and Anti-Spam (CEAS 2006), Mountain View, CA, USA, 2006.