CS6220 Data Mining Techniques – Spring 2015
Assignment 2

Submission Instructions

- **Your program must run on CCIS machines in WVH lab**
- **Create a README file, with simple, clear instructions on how to compile and run your code**
- **Zip all your files (code, README, written answers, etc.) in a zip file named** $\{firstname\}\_\{lastname\}\_CS6220\_HW2.zip$ **and upload it to Blackboard**

1. Decision Tree

   Table 1 below contains a small training set. Each line includes an individual's education, occupation choice, years of experience, and an indication of salary. Your task is to create a complete decision tree including the number of low's & high's , entropy at each step and the information gain for each feature examined at each node in the tree.

Instance	Education Level	Career	Years of Experience	Salary
1	High School	Management	Less than 3	Low
2	High School	Management	3 to 10	Low
3	College	Management	Less than 3	High
4	College	Service	More than 10	Low
5	High School	Service	3 to 10	Low
6	College	Service	3 to 10	High
7	College	Management	More than 10	High
8	College	Service	Less than 3	Low
9	High School	Management	More than 10	High
10	High School	Service	More than 10	Low

Table 1: Decision Tree Training Data

Please turn in a diagram similar to:

Top 6,4, .97
Education gain = <to be calculated>
    1. High School 4,1, <to be calculated>
        Experience gain = <to be calculated>
    Etc.
Etc.

Prune the tree you obtained using the validation data given in Table 2. Show your work.

Instance	Education Level	Career	Years of Experience	Salary
1	High School	Management	More than 10	High
2	College	Management	Less than 3	Low
3	College	Service	3 to 10	Low

Table 2: Validation Data

2. Implement the KNN classifier using C/C++, Java or Python. Your implementation should accept two data files as input (both are posted with the assignment):

   (a) A **training** dataset with class labels

   (b) A **test** dataset without class labels (i.e. only contains feature vectors)

   the training data contains examples of three different types of Iris (a type of flower), with each example having a class label of either *versicolor, virginica* or *setosa*. Each example has four (numeric) features: Sepal Length, Sepal Width, Petal Length and Petal Width. Your classifier must examine each unlabeled example in the **test** set and classify it as one of the three classes of Iris. The classification will be based on an *unweighted* vote of its k nearest examples in the **training** set. You should measure all distances using regular Euclidean distance:

$$d(x, y) = \sqrt{\sum_i (x_i - y_i)^2}$$

   If two or more classes receive the same (winning) number of votes, break the tie by choosing the class with the minimum total distance from the test point to its voting examples.

   Program output

   Output our program in the same format as the test file with the KNN predicted labels appended at the end for $k = 1, 3, 5, 7, 9$ (in this order).

   For exampke, if $\mathbf{x_i} = (6.7, 3.1, 4.4, 1.4)$ is of class *setosa* when $k = 1, 3, 5$ but predicts class *versicolor* when $k = 7, 9$, then out output line for x should be:

   6.7, 3.1, 4.4, 1.4, $\mathbf{x_i}$, setosa, setosa, setosa, versicolor, versicolor

3. SVM using Weka

   For this exercise, we apply SVM with several different kernels and hyper-parameter choices to the **veh-prime.arff** file provided with the assignment. Import this file into Weka (free download from http://www.cs.waikato.ac.nz/ml/weka/) and then select the SMO classifier found under classifiers/function. Use 10 fold cross-validation. You can make kerrnel and hyper-parameter choices by clicking on "SMO ..." appearing next to Choose.

   You will make 5 runs of the algorithms. Select PolyKernel with exponent option 1, 2, and 4. Then select RBFKernel with gamma set to 0.01 and 1.0. For each run record the number of correctly and incorrectly classified instances. Explain why some of the choices do not work well.

4. Kernels

   Assume that $x = (x_1, x_2)$ is a two dimensional vector and we have a function K defined as $K(x, z) = x_1 * z_1 + x_1 * e^{z_2} + z_1 * e^{x_2} + e^{x_2+z_2}$. Prove that K is a kernel.

5. Dual Form

The dual form convers a problem of finding an d-dimensional vector w into a problem of finding an n-dimensional vector α by finding α_j's satisfying $w = \sum_i \alpha_i y_i x_i$. Note: here x_i is the i^{th} example not the i^{th} feature. Given the data:

- 1, 0, 1 with label = 1
- 1, 1, 1 with label = -1
- 1, 0, 0 with label = 1
- 1, 1, 0 with label = 1

Assume we know that $\alpha = (-0.8, 1, 6.4, -1.9)$. Find the associated w vector and classify the test item (1, 0.8, 1) using both the primal and dual form.