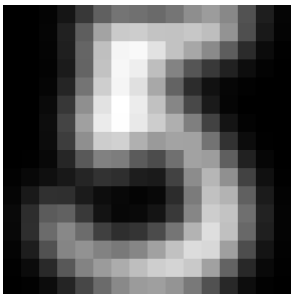
CSC529: Advanced Data Mining
Assignment #3
Total: (60 points plus 10 extra credit points)
Due: Wednesday, February 28, 2018, by midnight

The goal of this assignment is to 1) understand a very popular ensemble of classifiers called random forest; 2) compare the random forest ensemble with a single decision tree classifier, 3) understand non-linear classifiers, 4) compare linear and non-linear classifiers, and 5) understand decision boundaries through support vector machines.

Problem 1: (**Handwriting recognition using support vector machines**) In this problem, you will apply a support vector machine to classify hand-written digits. Download the digit data set from the course documents for week 6. The zip archive contains two text files. The file uspsdata.txt contains a matrix with one digit/data point (= vector of length 256) per row. The 256-vector in each row represents a 16 by 16 image of a handwritten number. The file uspscl.txt contains the corresponding class labels. The data contains two classes, the digits 5 and 6, and the class labels are stored as -1 and +1, respectively. For example, here is the first row, re-arranged as a 16 by16 matrix and plotted as a gray scale image:



Randomly select about 20% of the data and set it aside as a test set.

a. (10 points) Train a linear SVM with soft margin. Vary the soft margin parameter and plot the classification error as a function of the margin parameter. Discuss the results.
b. (15 points) Train a non-linear SVM with soft margin and Gaussian kernel. Vary both the soft margin parameter and the Gaussian kernel bandwidth (sigma) and plot the classification error as a function of the margin parameter and kernel bandwidth.
c. (5 points) After you have selected parameter values for both algorithms (in part a. and part b.), and trained each one with the parameter value you have chosen, compute the classification error on the test set. Report the test set estimates of the error for both cases along with the parameter values you have selected, and compare the two results. Is a linear SVM a good choice for this data, or should we use a non-linear one?
d. (10 extra credit points) For each method (linear and Gaussian), select five correctly and five incorrectly classified samples. Plot these as images like the one above. (Hint: To do so, you will have to write a 256-vector into a 16x16 matrix (function imresize.m) and plot this matrix as an image functions imshow.m and colormap(gray)) Can you see a qualitative difference between correctly classified and misclassified digits? Explain the results.

Problem 2: (E-commerce Customer Identification using ensemble of classifiers)
The data of e-tailer customers is posted under the course documents for week 6. The training data contains 334 variables for a known set of 10,000 customers and non-customers with a ratio of 1:10,

respectively. The test data consists of a set of examples and is drawn from the same distribution as the training set. The feature data is train10000.csv and the label data is train10000_label.csv with corresponding labels for the records in train10000.csv. The test10000.csv is the test data with corresponding labels for the records in test10000-label.csv.

Preprocessing steps to do:

- You may use SPSS, Excel or any other software to do the data preprocessing.
- **Missing values**: Check if there is any missing values inside the dataset, if so, perform an analysis to determine if you can ignore the cases with missing data. If you cannot ignore the missing data, fill in the missing values and explain the method that you used.
- **Normalization**: since the features have very different value ranges, apply a normalization procedure to make the features comparable (be on the same scale).
- **Attribute/Feature selection**: Since there are 334 features in the dataset, it may be useful to use some feature/attribute selection to reduce the dataset before training classifiers. Describe your selected method and explain how it works briefly.
- **Balanced data**: The dataset is a severely unbalanced dataset. You may want to balance the data before training the classifier. Describe your selected method to balance the data.

Here are some other implementation hints that you may want to take into account before running your classification approaches on the preprocessed data:

- If your training data has been applied a set of normalization or feature selection, you need to do the same with test dataset, otherwise the feature values are not consistent, and you will get absurd results on test data.
- Save your preprocessed dataset, then you can test different classifiers without redoing all these preprocessing steps.
- After you read the labels in Matlab, the class label 1/0 maybe regarded as numeric value rather than nominal labels. So you need to make sure that you covert the class label to a nominal type if needed.

Classify the data (where class 1 is the customer) by using the following approaches:

1. Run a decision tree algorithm and analyze how the preprocessing affects the results. Write down the corresponding performance measures for class 1 (customer) for each processing as indicated in Table 1.
2. Repeat 1) but using Random Forests.
3. Use your best classifiers you trained in 1) and 2) to predict the class labels for the test dataset and present the results as indicated in Table 2. Compare and analyze the results.
4. Describe the preprocessing methods you used in the above experiments (missing value estimation, normalization, attribute selection) and the classification parameters (for the decision trees and random forests).

Table 1. Comparing performances of classifiers on training set

Training Results	Sensitivity/precision	Specificity/Recall
Raw data		
Raw data with balanced data		
Normalize attributes		
Feature selection		

Table 2. Comparing performances of classifiers on testing set

Testing Results	Sensitivity/precision	Specificity/Recall
Raw data		
Raw data with balanced data		
Normalize attributes		
Feature selection		