Assignment 1: Probabilistic Classification

January 28, 2010

In this assignment, you will develop a number of classifiers for handwritten digits. You will use the supervised learning paradigm, in which you are given some examples of two different classes: handwritten 3's and handwritten 5's.

You will be expected to develop classifiers, using the "training samples", that classify the "test samples" as well as possible. **Important:** You will *not* be graded on how well your classifier works on the test data, but rather, on whether you implement the classifiers as described.

- Download the file digits.mat from web page. Use the command 'load digits.mat' to load it into matlab. Type 'whos' to see the variables that are defined in it. You should see four variables named train_threes, train_fives, test_threes, and test_fives. They are each a group of 50 images stored in a three-dimensional array. Try plotting a few of the images using imagesc to make sure they appear as you expect.

- You will be using two features to classify each digit. You can make up whatever features you like, but start by choosing properties of the images that give a range of different values. Examples would be:

  - the height of the digit in pixels
  - the width of the digit in pixels
  - the number of border pixels, i.e. the number of white pixels that are next to black pixels
  - the "left-right" weight of the digit. This could be defined as the percentage of white pixels that are to the right of the center of the bounding box of the digit.
  - any feature you want!

- Next, write two functions called estFeatDistsX.m ("estimate feature distributions"), where X is a name for each feature. For example, one of your functions might be called estFeatDistsHeight.m. This function should take all of the training data as an argument, and a number k of bins as another argument. It should return 4 things: the minimum value of the feature over the training set, the maximum value of the feature over the training set, and two probability distributions. Each probability distribution (one for the class of threes, one for the class of

fives) should give the estimated probability that the feature for that class falls in one of k bins between the minimum and maximum value for the feature. For example, if k is 5, then the third element of the probability distribution should be the frequency with which the feature had a value between 40 percent and 60 percent of its possible range. Obviously, each probability distribution should sum to 1.

- Now write a function called estJointFeatDists.m ("estimate joint feature distribution") which takes as arguments the training data and a number of bins k and returns a matrix holding the joint distribution of both variables as estimated by the frequency of the joint features in the training sets.

- Using these ranges and probability distributions for each feature for each class, you will implement a variety of classifiers. They should include:

  - A classifier using the first feature only. Try various numbers of bins, but at least 2 bins and 10 bins.
  - Do the same thing for your other feature.
  - A classifier that uses both features with 2 bins each, assuming the features are independent.
  - A classifier that uses both features with 2 bins each, assuming the features are not independent.
  - Repeat the last two classifiers using 10 bins each.

You should turn in all of your Matlab code, and a file in which you discuss your results for the various experiments and the relationship among them.