Final Exam
Math 350
Fall 2009

The Final Exam will be in two parts. The first part, dealing with classifiers, is given below. The second part, consisting of one or two questions will be given to you during our meeting on Monday or Tuesday.

This is real data, so we should not expect the classifiers to be perfect. In fact, I am looking for the modeling process more than "the correct solution" (in terms of a perfect classifier).

To help you, there are two script files on our class website that construct an RBF classifier for data representing flowers- Use these to help you.

What to turn in: A nice write up exported to PDF.

Deadlines: Turn in it before you leave for break or by Friday morning of Finals week.

Problem:

Given a collection of photographs of people, construct a classifier based on gender. Here are some specifics:

First, separate the data into a "Training" set and a "Validation" set. For the training set, use 7 (out of 13) randomly chosen photos for each gender for a total of 14 in the training set. We will construct a classifier using the two techniques below.

1. Method 1: Subspace classifier. For this classifier, construct a mean and best basis for each gender. The gender of a new face is defined to be the subspace that best matches the given face. You should run your algorithm several times using different choices for training, and then settle on the best set. For each set, use as few basis vectors as possible.

2. Method 2: RBF Classifier. In this case, we will input a face, and output the gender. Write the RBF script yourself using two centers and Gaussian transfer function. You may run this several times to see if you can find a best width for the two Gaussians, and also with different training sets to see if this value changes. Compare this output to Matlab's built-in RBF routine (use the help file to determine how to build the RBF using only two centers).

Commands that we wrote: `edm`, `rbf1`. Commands that are built in: `newrb`, `sim`

Make some conclusions- How did your algorithms perform? In particular, use a confusion matrix for each method.

The next page describes the data you'll be working with.

Data Format for the Gender Classification

- **Description of the Data:** To load the data, type `load MenAndWomen` The main data file is in the 3-d array `Photos`, and the gender (0 or 1) is in the vector `classes`.

  The data file is $162 \times 149 \times 26$ of unsigned integers, meaning that there are 26 photos, each 162×149. There are 13 men, 13 women. To work with these, we'll have to change the `uint8` format to `double` by typing: `Photos=double(Photos);`

  To view the images, for example the third one, type: `imagesc(Photos(:,:,3))` followed by `colormap(gray);` To make the image look better, you might also use `axis square` and `axis off`

- **Conversions:** To make the data set into a regular matrix (and have each photo as a column), you can use the command:

  ```
  X=reshape(Photos,162*149,26);  %X is 24138 x 26
  ```

  You might verify that you can undo this by typing:

  ```
  Y=reshape(X,[162,149,26]);     %Y is 162 x 149 x 26
  ```

  Once you've verified that Y is the same as `Photos`, be sure and delete Y to save space: `clear Y`

- To separate into genders, you can use two vectors, one for boys and one for girls:

  ```
  Boys=find(classes==1);
  Girls=find(classes==0);
  ```

  so that `X(:,Boys)` will contain all the photos for the males, `X(:,Girls)` will contain all the photos for the females.

- Be sure to use the `subplot` command (See the sample file from the eigenfaces script) to do several plots in the same figure window. To see one of the images (say Face 19) from the original data set, type: `imagesc(Photos(:,:,19));`

- To obtain a random sampling of 20 photos for training, we could write:

  ```
  idx=randperm(26);  %idx is a vector with 1-26 in random order
  TrainingPhotos=X(:,idx(1:20));
  TrainingClasses=classes(idx(1:20));
  ```