CSE 426/326 Pattern Recognition --- Spring 2015 --- Prof. Baird

Final Choices: Exam, Project, Scholarly Report, Etc

You may choose to take a three-hour closed-book Final Exam during Exam Period.

Or, as an alternative, you may choose to carry out a Project, including any of these:
(a) a scholarly review report on some topic in the pattern recognition R&D literature; or
(b) a theoretical (mathematical) analysis of a challenging problem; or
(c) some programming/experimental project such as one of the following suggested character-image classification projects. (If you choose a scholarly review report, I can offer you a selection of technical articles; some to my office hour---Weds 12 noon – 1 PM, in PL 380 to discuss this. Also, you may propose variations on these projects; for example, if you would like to work with different data sets---*e.g.* many more samples per class, and/or many more classes---feel free to ask me: I may be able to provide them.)

Please tell me the Final Choice you have made, no later than **Thursday April 9.**

Character-Image Classifier Projects

Given a training set A of 100 samples/class and three test sets B, C, & D of 100 samples/class (chosen from the list of data sets below), do:

- Reposition each sample character image by centering it within a 16x16 pixel array (with equal white-space margins top/bottom and left/right; rarely, a character-image may be larger than 16 pixels in width or height---in this case, simply trim off the extra rows or columns roughly equally from left-&-right or top-&-bottom).

- Extract, from each centered sample, twenty normalized central moment features (as discussed in HW#5, but extended beyond the ten described there).

- Train four classifiers using the data from set A, each using a different classification method given in the list below.

- Test each of the four classifiers on sets A, B, C, & D; note the best error rate 'E' achieved by any of these four classifiers, averaged over the three test sets B, C, & D.

- Explore new features which will allow one of the four

classifiers to cut E by at least a factor of two (that is, to drop the error in half at least). You are free to invent any sort of feature that you wish.

- In addition to trying new features, design and implement a fifth classifier, different from the four above, which will cut E at least in half. You are free to try any trainable classifier technology in the PR literature (e.g. SVMs, ANNs, CARTs), including any discussed in DHS. You are free to find and use any pre-existing software for training and testing this fifth classifier.

Choice of data sets:

I) Lowercase character classes: a, c, e, m, n, o, r, u, v, x (10 classes) in a single typeface 'Times Roman', under a range of image qualities. Filenames: C-I/
A-a.txt B-a.txt C-a.txt D-a.txt ...
… & similarly for c, e, m, n, ...

II) Uppercase character classes: B, C, D, E, I, J, O, R, U, V (10 classes) in a single typeface 'Times Roman', under a range of image qualities. Filenames: C-II/
A-B.txt B-B.txt C-B.txt D-B.txt ...
… & for C, D, E, I, ..

III) Punctuation classes: . , - : ; ! ? / ( ) (10 classes) in a single typeface 'Times Roman', under a range of image qualities) (NOTE: periods often shrink to single pixels, which lead to singular covariance matrices; if you can't find a way to survive the resulting numerical instabilities, feel free to omit 'pe'.) Filenames: C-III/
A-pe.txt B-pe.txt C-pe.txt D-pe.txt (for 'period')
... & 'cm' (comma), 'da', 'co', 'sc', 'ex', 'qm', 'sl', 'lp', 'rp'

IIII) Typeface classes: Times Roman, *Times Italic*, Courier, etc (10 classes) of the character 'e', under a range of image qualities. Filenames: C-IV/
A-TR.txt B-TR.txt C-TR.txt D-TR.txt (Times Roman)
... & similarly for:
TI Times Italic
TB Times Bold
HR Helvetica Roman

HR Helvetica Roman
HI Helvetica Italic
HB Helvetica Bold
CR Courier Roman
CI Courier Italic
CB Courier Bold
AR AvantGarde Roman

V) Image Quality classes: nearly ideal, slightly blurred, highly blurred, etc (10 classes) of the character 'e' in typeface 'Times Roman'. Filenames: C-V/
A-ideal.txt B-ideal.txt C-ideal.txt … for nearly ideal (undegraded) images,
... & similarly for:
blur1 blur2 fat thin noisy tall wide rot1 rot2

These data sets are posted in CourseSite's `Final Project Assignments & Data` folder. Each data set is in a separate *.tar archive file: C-I.tar, C-II.tar, etc. Pick one of the data sets (I – V) above, and tell the Instructor which one you chose.

With each of the following four methods, use Data Set A of 100 samples/class as the *set of prototypes*, and classify the remaining three sets. Print a **summary table of the error rates**, with one row for each method, and one column for each data set. Print out all three confusion tables for Methods 1 and 5 only.

1. *Moment-space minimum-distance classifier*: each sample is assigned to the category whose class mean, in 20-dimensional "moment space," is nearest to it under Euclidian distance. (This is the same as a Bayes classifier assuming *identity* covariance matrices for all the ten classes, and results in a linear classifier.)

2. *Moment-space classifier with identical covariances*: use the average covariance matrix (averaged over all classes, *i.e.* compute the covariance matrix for each class separately, then average these matrices component-wise across all ten classes). (This is DHS's Case 2, also resulting in a linear classifier.) Note that this will require computing (among other things) the inverse of an 20x20 symmetric matrix: feel free to use any pre-existing software tools you can find to compute these.

3. 1*NN in moment space*, under the Euclidian (Minkowski L2) metric. Note that you are allowed to replace the squared distance with an

equivalent linear expression. Decide ties by lexical precedence (*i.e.* choose the class that comes first in the arbitrary given order of classes).

4. 5*NN in moment space*, under the Euclidian (Minkowski L2) metric, breaking ties lexically.

Report Format. At the start of the report, summarize your error rates in a well-formatted table of the error rates, with one row for each method. Here's a hypothetical example:

SUMMARY TABLE

Total error counts on four classifiers

Test set: Method	A	B	C	D
1	1			
2	0			
3	0			
4				

Next, print the confusion tables, one for each Method. Here's an example:

CONFUSION TABLE

Method 1 – Twenty Moments, Identity Covariance Matrix

(trained on A, tested on A)

Classified as:	a	c	e	m	n	o	r	s	x	**z**	**Error Type II**
True class											
a	9				1						**1**
c		8							2		**2**
e			8						1	1	**2**
m				9						1	**1**
n					10						**0**
o						10					**0**
r							10				**0**
s								10			**0**
x									7	3	**3**
z										10	0
Error Type II	0	0	0	0	0	0	1	0	0	3	**5**	**9**

Each entry represents the number of samples (an integer) of a particular class classified in a particular way, with the actual class corresponding to the row and the decision corresponding to the column. In the bottom right corner, print the total number of errors. Type I Error for "a" is the number of "a"s that are misclassified.
Type II Error for "a" is the number of times a character is misclassified as "a".

There will be 6 **confusion tables, one for each Classification Method. Formatting may be a lot of work, but it's important that the results of experiments be readily understood.**

Please attach your program listing at the end.