CAP 6638 **Pattern Recognition** **Spring 2015**

MW 05:00 PM – 06:15 PM in NES 013

Project #1

The objective of this assignment is to extract some continuous-valued features from the scanned digits in our data set. In subsequent assignments we will use them to illustrate algorithms based on normal distributions.

Use the MNIST digit recognition data sets consisting of sample 28X28 (width x height) images of 10 printed digits ('1', '2', '3', '4', '5', '6', '7', '8', '9', '0'). Each digit is viewed as a 2-D binary array. Let **i be the row index** increasing from *top to bottom of each character*, and **j be the column index** increasing from *left to right of the character*. **It is important that everyone use the same coordinate system, otherwise we cannot compare results!** You are requested to test your program on some standard data (in ascii format). For this assignment use **Dataset A.** Dataset A consists of 10 classes; and 10 examples for each class**.** Print out all the characters as an image with the correct aspect ratio to make sure that you are reading the data right, and to identify the character labels. Add coordinate axes so that one can identify pixel (i,j)..
We will use ***k* as the class index**, with **k=1 (not 0) for "1" and k=10 for "0". The examples in each class are also labeled 1 to 10: 1-1, 1-2, … 1-10, 2-1, 2-2, ….**

Definition: The (p,q)th Central Moment of a pattern X is:

$$\mathbf{Mpq = \Sigma\Sigma\ [((i-imean)^p)\ ((j-jmean)^q)\ X(i,j)],} \qquad \mathbf{i=1\ to\ I,\ j=1\ to\ J.}$$

where X(i,j) is the pixel (1 or 0) in the (i,j)th entry of the pattern matrix, I is the number of rows, J is the number of columns, and (imean, jmean) is the centroid of the pattern. For example: $\mathbf{imean=\Sigma\Sigma\ [\ X(i,j)*i]/\Sigma\Sigma\ [\ X(i,j)]}$.
The values of imean and jmean should not be rounded or truncated.

1. For the 10 samples of each class of the training set, compute 8 central moments of the class-conditional sample distributions for each class. The moments vary greatly in magnitude, and some are negative, **so normalize the moments by dividing the moments of each pattern by the overall (all 100 samples) root-mean-square value for that moment (*overall rms*)**. For each class, compute the class mean and class variance of the *normalized* moments. (To compute the variance, divide by N rather than by N-1.)
Print a table **in fixed-point format (rows, columns, and decimal points aligned)** showing the feature values and class-conditional statistics as shown on the next page.

2. Compute the *class-conditional sample covariance matrix* $\boldsymbol{^kC}$ using the 8 moments as the features for each class k in the training set. Divide by N throughout, rather than N-1. Print these matrices for Dataset A. Please label each matrix **unambiguously** as **Cov-Dataset-A-1;... Cov-Dataset-A-0.** You may use any software package for computing the covariance matrices, or write your own. Please make sure that matrices don't span page breaks.

3. Invert each matrix and print $\mathbf{^kC}$, $\mathbf{^kC^{-1}}$ (k = 1, 2, ...,10) as 8x8 arrays.
Check that $\mathbf{^kC\ ^kC^{-1} = I}$. (For debugging, you may want to work with the 2x2 covariance matrices of the first two features.) Please print all matrices in **fixed-point format (rows, columns, and decimal points aligned)** for legibility. Make sure that you normalize your values by the appropriate power of 10 to show 2 or 3 significant digits. Label as **Inverse-Cov-Dataset-A-a**, etc. Show the multiplier (*true value = multiplier x value printed*).

4. Compute and print the *average covariance matrix* ***C*** by averaging the ten class-conditional matrices term by term. Compute its *inverse* $\boldsymbol{C^{-1}}$ and label it **Inverse-Cov-Dataset-A**, etc.

Formatting the data in a sensible manner is an important part of this homework.
The features will have to be computed for other datasets as well, so be sure to document and save your programs.

Submission: Source code and report in Canvas. Your program should accept two arguments: a file name and the path of the dataset A. The program should write the output to the file name given as argument. The program should write the values shown in the next page. Details about the report is given in the subsequent page.

The mean, variance, overall rms, multiplier and the covariance matrix values shown here are random numbers and do not compare this with the values you get for the Dataset A. Store the calculated values, round-off only for printing.

Normalized Central Moments (Dataset-A)

	M_{00}	M_{02}	M_{11}	M_{20}	M_{03}	M_{12}	M_{21}	M_{30}
1-1	0.908	0.307	...					
1-2	0.946	0.318	...					
.								
.								
1-10								
mean(1)	1.002	0.357						
var (1)	0.003	0.001						
2-1	0.687	0.302						
3-2	0.709	0.291						
.								

0-10

mean(0)
var (0)

overall rms 131.09 3797.28

Print the covariance and inverse-covariance matrix for each class and the average covariance and inverse-covariance matrix as shown below:

Covariance matrix of the digit (1): (multiplier: 0.010000)
1.710 2.066 -0.638 1.668 -1.119 3.570 0.575 -2.877
2.066 3.534 -6.549 1.692 -2.120 3.823 0.661 -1.749
……
…….

Inverse covariance matrix of the digit (1): (multiplier: 100.000000)
28.462 -9.163 -0.304 -12.682 6.938 3.727 5.197 5.668
-9.163 10.814 1.931 0.262 -0.581 -1.936 -0.495 -0.233
…….
……

Store the calculated values, round-off only for printing!

Report should include the definitions and formulas for all the operations you performed in the project, and the output of the program as described below.

The mean and variance values shown here are random numbers and do not compare this with the values you get for the Dataset A. Store the calculated values, round-off only for printing.

1. Print the mean and variance of the 8 normalized central moments for all the classes.
2. Print the overall rms for all the 8 normalized central moment.
3. Print the covariance and inverse covariance matrix values as described below.

Normalized Central Moments (Dataset-A)

	M_{00}	M_{02}	M_{11}	M_{20}	M_{03}	M_{12}	M_{21}	M_{30}
mean(1)	1.002	0.357						
var (1)	0.003	0.001						
mean(2)	1.002	0.357						
var (2)	0.003	0.001						
:								
:								
mean(0)								
var (0)								
overall rms	131.09	3797.28						

For each class print the covariance and inverse covariance matrix values at the following index:
Class 1 : [5, 7] , [3, 7]
Class 2 : [6, 7] , [5, 1]
Class 3 : [4, 1] , [4, 2]
Class 4 : [3, 8] , [2, 5]
Class 5 : [3, 5] , [3, 7]
Class 6 : [7, 8] , [4, 6]
Class 7 : [3, 5] , [2, 4]
Class 8 : [8, 5] , [8, 6]
Class 9 : [3, 7] , [2, 5]
Class 10 : [3, 2] , [1, 5]

Print the Average covariance and inverse covariance matrix values at the following index:
[3,5], [6,3]