# CSC 340 Test 1

28 January 2013

Due: 11 February 2013

Name: Julian Paolo Usero

## Instructions:

This is a take home test. You may use your book and software that you have developed yourself for this course. If you find it convenient to do so, you may also use a plotting tool such as Excel, SAS, or gnuPlot to render graphics. You ***may not*** use commercial software or statistical tools, such as those included in Excel or SAS, or MatLab, to generate the results you report. You may not use the built-in matrix or vector operations of any programming language to find matrix inverses or determinants, or to perform matrix or vector operations such as addition, subtraction, multiplication, or scalar multiplication. ***Your report must arise from algorithms that you yourself have implemented.*** (You may use commercial software only to check your work.) ***Again, the computational results required for this test must arise from your own personal implementation of the required algorithms in a programming language such as Java, C, or C++.*** You are required to do your own work in implementing algorithms, creating supporting code, and applying your software to these problems. You must submit your source code together with the examination results.

The expectation and requirement is that you will write your findings as you would document the results of laboratory experiments. In order to assure that all questions receive complete responses, many students have found it useful to embed their responses in this test document itself. ***In order to receive credit for any given problem, you must also be prepared to provide a live demonstration of your implementation of the associated algorithms and to explain any component of the source code that you provide.*** Your submission may contain a mix of program output, program generated charts, or handwritten work, as appropriate.

## Due:

The findings and your source code are due by 11:59 PM, 11 February 2013.

## Problems:

Consider the data posted to the Web in the file “2013 Test 1 data.txt”, which represent information describing the objects in the two classes illustrated in the chart below.

For questions 1-8, suppose that the objects in the two classes have been measured with respect to two parameters, x and y, and the measurements are multivariate normally distributed for each dimension for each class.

1. Find the *mean* vectors **m**1 and **m**2 for each of the classes. (Note: **m**1 and **m**2 are each nx1 vectors where n=2.) This problem requires the computation of a sum of nx1 vectors, and computing a scalar multiple, 1/k, of the sum for each class, where k is the number of objects in each class.

**----1 mean vectors of the classes 1----**

**Class 1 Mean Vector: 2.67399036405 0.328465665525**

**Class 2 Mean Vector: 0.755607564725 2.3790660250749993**

1. Find the covariance matrices ****1 and ****2 for the classes. This problem requires multiplying each nx1 vector in a class by its 1xn transpose, accumulating the sum of the nxn products for the class, and multiplying the sum for the class by a scalar, 1/k.

**----2 covariance matrices of the classes 2----**

**Class 1 covariance:**

**1.17 0.27**

**0.27 3.81**

**Class 2 covariance:**

**0.88 -0.17**

**-0.17 1.0**

1. Find and report the ***determinants*** of the covariance matrices |****1| and |****2| for the classes. This problem exploits the determinant finder derived from the Gauss reduction algorithm.

**----3 determinats of the covariances 3----**

**Determinant 1: 4.3804651628899824**

**Determinant 2: 0.8500412578751957**

1. Find and report the ***inverses*** of the covariance matrices ****1-1 and ****2-1for the classes. This problem uses the inverse finder algorithm derived from the Gauss-Jordan reduction algorithm.

**----4 inverse of the covariances 4----**

**Inverse 1:**

**0.87 -0.06**

**-0.06 0.27**

**Inverse 2:**

**1.18 0.2**

* 1. **1.03**

1. Find and report the discriminant functions g1(**x**) and g2(**x**) for the classes? Report these with the right-hand side of each equation in matrix form. Refer to the classifier lecture notes for a detailed example.

1. Into which classes would you place the points **m**1 and **m**2? Use your matrix tools to evaluate the discriminant functions to support your choices.

**----6 mean vectors 6----**

**discriminant results: g1: -0.7385774601612112 g2:-3.4688571508191246**

**mean of class 1 should be in class 1**

**discriminant results: g1: -3.1399572344489797 g2:0.08123519599951878**

**mean of class 2 should be in class 2**

1. Use your personally implemented matrix manipulation tools to determine how many classification errors occur when you apply the discriminant functions to the example data?
   1. List the misclassified points separately for each class and provide the values of both discriminant functions for each point. (It might be a good idea to use a table to organize this response. The table would contain on row and one column for each class. In the table, the entry in row j and column k should report the number of objects in class j that the classifier indicates should be in class k.)

**----7 classification errors 7----**

**--a--**

**set one miss-classified**

**x y g1 g2**

**0.79 1.6 -2.63 -0.23**

**1.65 3.1 -2.39 -0.79**

**1.94 1.65 -1.26 -0.85**

**1.65 1.84 -1.59 -0.45**

**total: 4**

**set two miss-classified**

**x y g1 g2**

**1.93 0.28 -0.88 -1.91**

**3.95 1.56 -0.91 -1.7**

**3.87 0.55 -2.47 -2.53**

**total: 3**

* 1. How many examples are correctly identified for each class?

**--b--**

**class 1: 36/40 correctly identified**

**class 1: 37/40 correctly identified**

1. Estimate and plot the boundary contour generated by the classifier.
   1. Notice that along the exact boundary contour g1(**x**) = g2(**x**) so that for an estimation of the boundary, | g1(**x**) - g2(**x**)| ≈ 0.0.
   2. Subdivide the area of interest into a grid and evaluate the magnitude of the difference of discriminant function values, |g1(**x**) - g2(**x**)| at the grid points.
   3. Choose a small value ε.
   4. When | g1(**x**) - g2(**x**)| < ε, report (so that you can plot) a boundary marker point.
2. Linear systems:
   1. If one a solution exists, use your implementation of Gauss-Jordan Elimination Algorithm to ***estimate the*** ***solution*** for the following linear system:  
      .

**1.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.65**

**0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.0 -12.74**

**0.0 0.0 1.0 0.0 0.0 0.0 0.0 0.0 0.93**

**-0.0 -0.0 -0.0 1.0 0.0 0.0 0.0 0.0 6.16**

**0.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 11.27**

**0.0 0.0 0.0 0.0 0.0 1.0 0.0 0.0 5.35**

**-0.0 -0.0 -0.0 -0.0 -0.0 -0.0 1.0 0.0 -5.67**

**0.0 0.0 0.0 0.0 0.0 0.0 0.0 1.0 -5.26**

* 1. What is the determinant of the coefficient matrix A?

**2.8971995E7**

* 1. If they exist, what are
     1. The inverse of the coefficient matrix A-1,

**-0.02 -0.05 0.0 -0.0 0.06 0.04 0.01 -0.0**

**0.16 -0.06 0.2 -0.08 -0.04 -0.33 0.2 0.12**

**-0.01 0.03 -0.01 -0.01 0.02 0.04 -0.01 0.01**

**-0.07 0.03 -0.06 0.07 0.07 0.23 -0.13 -0.13**

**-0.06 0.1 -0.03 0.1 0.18 0.52 -0.42 -0.12**

**-0.06 0.05 0.06 0.07 0.03 0.16 -0.16 -0.01**

**0.03 0.0 0.02 0.03 -0.04 -0.2 0.12 0.1**

**0.07 0.01 0.21 -0.05 0.16 -0.06 -0.11 0.0**

* + 1. The determinant of A-1, and

**3.451609045217634E-8**

1. The product of the determinants of A and A-1?

**1.0000000000000007**

* 1. If A-1 exists, check your system solution results by performing the appropriate matrix multiplication and reporting the results.

1. If it exists, what is the ***condition number*** for the coefficient matrix for the system given in problem 9?

**2361.73**