Homework 1

Due Friday, February 15, 7pm Chicago time

For the computer problem below (#1), turn in your code, printout of any graphs you make, and an explanation of what you have done. For the math problem below (#2), be sure to show your work.

1. Fisher's linear discriminant

When you work with real high-dimensional data, things get messy. So let's build our intuition by first looking at some simpler artificial data sets.

I have created artificial data having 4 features (4-dimensional data). Each data example is a vector $\mathbf{x} = (x_1, x_2, x_3, x_4)^T$. There are two classes, ω_1 and ω_2.

In real-life problems, we do not know the distributions of the data or their true statistics. But in this case, since I created the data myself, we do know. For each class, the probability density function is a multivariate Gaussian. The mean vectors for the two classes are:

$$\mu_1 = E[\mathbf{x} \mid \omega_1] = \left(0,0,0,0\right)^T$$

$$\mu_2 = E[\mathbf{x} \mid \omega_2] = \left(2,2,0,0\right)^T$$

The covariance matrices for the two classes are identical (both distributions are characterized by "white noise" with variance σ^2):

$$\Sigma_1 = \mathrm{cov}[\mathbf{x} \mid \omega_1] = \sigma^2 \mathbf{I}$$

$$\Sigma_2 = \mathrm{cov}[\mathbf{x} \mid \omega_2] = \sigma^2 \mathbf{I}$$

a) First, try to use your own reasoning (not Fisher's method) to find a good linear classifier for this problem. For this part, you may use what I have told you about the true distributions. As I will explain in class, under this distribution, the data will be a spherical ball of points (a hypersphere, to be precise). Try to imagine these distributions, remembering that the mean vectors represent the centers of these hyperspheres. Try to guess (using geometrical reasoning) what vector **w** would produce good separation of the two classes, when used in a linear model where $y = f(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + w_0$. What would be a logical choice for w_0 ? If σ^2 is increased, will the classifier perform better or worse?

b) Next you will use MATLAB to compute and study the Fisher discriminant for this problem. For this analysis I am giving you two data sets (in files `hmwk1-dataset1.mat` and `hmwk1-dataset2.mat`). In the first data set, the distributions have $\sigma^2 = 1$; in the second data set $\sigma^2 = 4$.

   1. Visualize (graph) the data. When data are more than three-dimensional, you cannot view them directly. One method of displaying high-dimensional data is the "trellis plot" – a matrix of scatter plots, with each scatter plot in the matrix graphing one of the variables against another. For example, the scatter plot in the (2,1) position in the matrix plots x_2

against x_1. You can think of each plot as the projection of the 4D scatter plot onto a different plane, as if we were viewing the clouds of points from different directions (along different coordinate axes). MATLAB has a command to make trellis plots, called `gplotmatrix`. To learn more about this, in MATLAB type `help gplotmatrix`.

Use the following code to display the trellis plots for these data:

```
n=1000;   %This is the number of data points in each class
xx = [x1,x2];   %This concatenates the two classes into one array
yy = [ones(1,n),2*ones(1,n)];   %This assigns class labels to the
                                %data examples
figure; gplotmatrix(xx,[],yy','br');%This draws the trellis plots
```

Looking at the scatter plots, we can see that the two classes have good separation when viewed from certain points of view, but no separation at all in other directions. Relate these graphs to your original thinking in part (a). In which plot(s) do you see the best separation?

2. For each data set provided, compute the Fisher discriminant vector **w** using the eigenvector equation we learned in class:

$$(S_W^{-1}S_B)\mathbf{w} = J(\mathbf{w})\mathbf{w} \ .$$

Remember that the best **w** is the eigenvector corresponding to the largest eigenvalue, because the eigenvalue is the Fisher ratio $J(\mathbf{w})$, which is a kind of signal-to-noise ratio.

To help with this, I am providing you with a MATLAB function I have written (`eigsort.m`) that outputs the eigenvectors in descending order of eigenvalue. Thus, the first eigenvector is the one with the largest eigenvalue.

What **w** did you get for each data set? What Fisher ratio did you find for each data set? The value of σ^2 is different in the two data sets. How did this affect the Fisher ratio? Is this what you expected?

Note: The sign of the eigenvectors you calculate is not important. For example, (1,0,0,0) indicates the same axis as (-1,0,0,0).

3. For each data set, do the following:
   a) Compute $\mathbf{w}^T\mathbf{x}$ for each example data point.
   b) Plot two histograms of $\mathbf{w}^T\mathbf{x}$ (one histogram for each class) on a single graph. The result should be two overlapping Gaussian-shaped functions (but they will be noisy, not smooth like a Gaussian).
   c) How do the plots differ for the two data sets? Which data set shows better separation of the classes?

2. Quadratic discriminant functions

In class we will show that the Bayesian discriminant function is **linear** when there are two classes, obeying multivariate Gaussian distributions with **equal** covariance matrices. Show that the discriminant function is **quadratic** when the covariance matrices are **not equal**.