ECE-497/BME-491: Applied Biomedical Signal Processing

Laptop Day #12

Due at the end of class, February 15, 2007

Today we have the following goals

- estimating the pdf of a sample
- using the likelyhood ratio to make decisions

1) Download *laptop12.m*. This routine generates data, uses the data to get sample statistics and estimated pdfs, construct the likelyhood ratio, generate new data to test, and record the results with the new data. There are 9 arguments to this program:

- *m_s*, the mean for the pdf for the sick
- *s_s*, the standard deviation of the pdf for the sick
- *K_s*, the number of bins to use in generating the pdf for the sick
- *m_n*, the mean for the pdf for the normal
- *s_n*, the standard deviation of the pdf for the normal
- *K_n*, the number of bins to use in generating the pdf for the normal
- *p*, the probability that a patient undergoing the test is normal
- *N_pdf*, the number of data points to generate for estimating the pdfs
- *N_pts*, the number of patients to test the likelyhood ratio against

The first thing the program does is to generate data with the appropriate statistics. This data is to be used for the remainder of the program. We cannot assume we actually know what we just did!

The next thing the program does is to plot histograms of the data generated and estimate the mean and standard deviations of the generated data. This implicitly assumes the data has a Gaussian pdf. If it did not have this pdf, a different formula may need to be used.

Next, a function describing the pdf is generated. This is an *implicit function*, and is pretty easy to use. If I wanted the pdf of the sick patients evaluated at z, I would just type *pdf_s(z)* and the function would return the result. It also works if z is an array.

Now we need to estimate the pdf for the sick patients, and we will compare this with the known form of the pdf using the measured statistical parameters (μ_s and σ_s). To generate the pdf, we proceed as follows:

- determine the minimum (a) and maximum (b) range of the data

- break the range into K_s equally spaced segments $dz = [d_1, ..., d_{K_s}]$)
- determine the number n_i of sample points that fall within bins centered at these points (use the *hist* command)
- determine the pdf by normalizing

$$pdf = \frac{n_i}{N_{pdf}} \frac{k_s - 1}{b - a}$$

Plot the estimated pdf (plot pdf vs. dz)

3) Repeat step (2) for the distribution for the normal patients.

4) Compute the likelyhood ratio for the data from normal patients we want to test against. Determine the number of correct and incorrect diagnoses.

5) Compute the likelyhood ratio for the data from sick patients we want to test against. Determine the number of correct and incorrect diagnoses.

If you run the program as

```
laptop12(2,1,50,3,2,50,0.4,10000,10000);
```

You should get results like those shown in Figure 1, though it will not be exactly the same.

6) Vary the probability p, what happens as p is changed?

7) Keep p fixed, and vary *cns* and/or *csn*, the costs of choosing incorrectly. What is the effect of changing these parameters?

8) Assume $p = 0.9$ and we want to be able to detect 90% of the sick people while not claiming all of the normal people are ill. Determine how to do this and turn in your plot.

Turn in your code

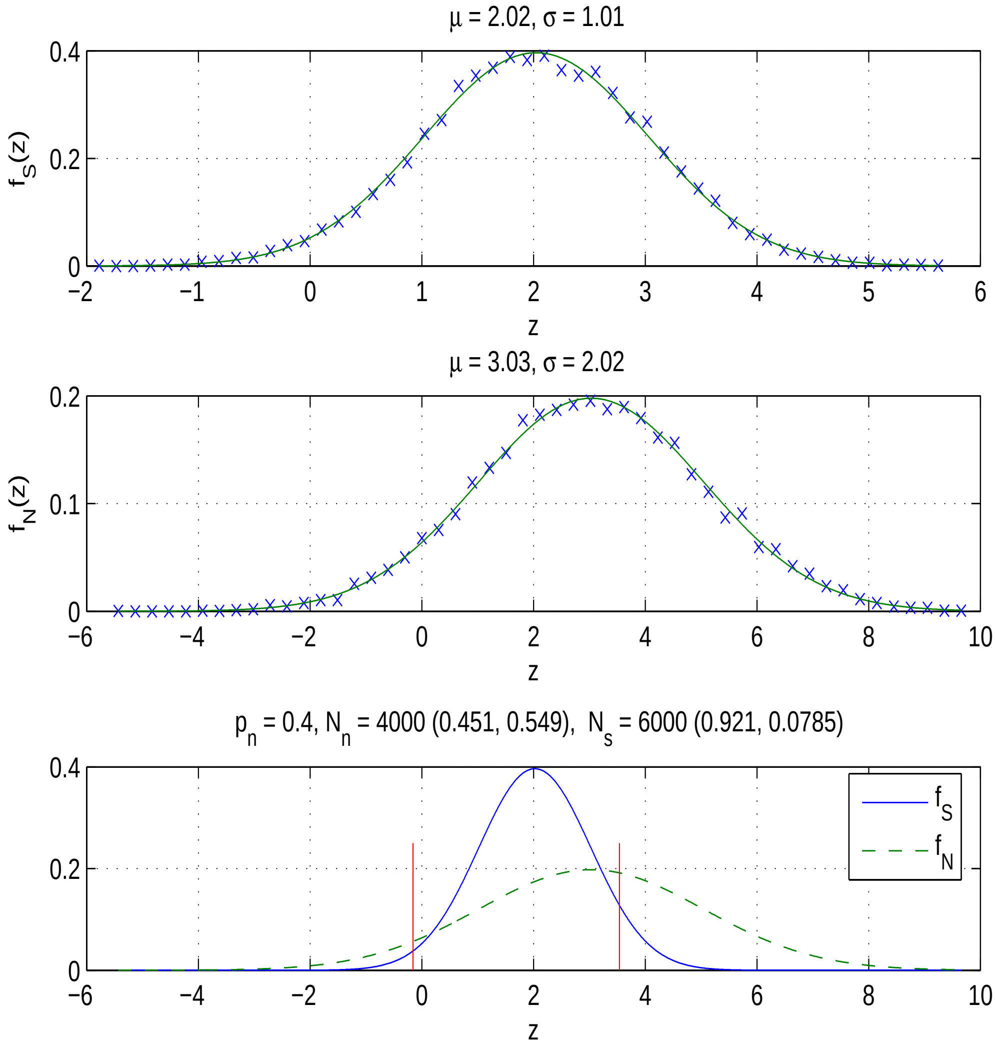

Figure 1: Density functions and results for example in part 5.