STA 410/2102, Spring 2002 — Assignment #3

Due at **start** of class on April 2. Worth 18% of the final mark.

Note that this assignment is to be done by each student individually. You may discuss it in general terms with other students, but the work you hand in should be your own.

This assignment concerns the same model as the last assignment, in which response variables, y_i, with values of -1 and $+1$ are modeled in terms of associated explanatory variables, x_i, as follows:

$$P(y_i \,|\, x_i) \;\; = \;\; \alpha/2 \;+\; (1-\alpha) g(y_i(\beta_0 + \beta_1 x_i))$$

where $g(z) = 1/(1+\exp(-z))$.

For this assignment, you will find maximum likelihood estimates for this model using the EM algorithm. This requires rephrasing the model so that there is some "missing data". We introduce unobserved binary variables u_i for each case, and define the joint distribution of y_i and u_i for a given x_i as follows:

$$P(y_i, u_i \,|\, x_i) \;\; = \;\; \begin{cases} \alpha/2 & \text{if } u_i = 1 \\ (1-\alpha) g(y_i(\beta_0 + \beta_1 x_i)) & \text{if } u_i = 0 \end{cases}$$

Summing over the two possible values for u_i results in the original model for y_i given x_i.

The EM algorithm for this model consists of alternating E and M steps. In the E step, you should find the conditional distribution for the u_i given the x_i, the y_i, and the current estimates for the model parameters, α, β_0, and β_1. In the M step, you should update the parameters to the values that maximize (or at least improve, see below) the expected value of the log likelihood based on y_i and u_i jointly, averaging with respect to the distribution for u_i found in the E step.

You should start by working out the details of the E and M steps based on the general form of the EM algorithm. (Don't just guess at what they should be by analogy with other models.) For the E step, you should end up with a fairly simple formula for the probabilities of each of the u_i being 0 or 1. (Note that the u_i are independent of each other, so this is all you need.) For the M step, you should find a fairly simple formula for the new value of α that maximizes the expected log likelihood for the complete data, but the values of β_0 and β_1 that maximize this will have to be found numerically, by maximizing an expression that you derive.

You should then write a program to find the maximum likelihood estimates using EM. You should do the numerical maximization with respect to β_0 and β_1 (but *not* α) using the built-in `nlm` function in R. You will need to supply `nlm` with a function for evaluating minus the expected complete-data log likelihood.

You should investigate how robust the EM algorithm is compared to the Newton iteration you did for the previous assignment. You should also investigate how many iterations are needed for EM to get accurate results. You can set the number of iterations manually, rather than implementing some automatic stopping rule. You should print out the parameter estimates and the actual log likelihood (using your function from the previous assignment) at each iteration, so that you can see what's happening.

You should also evaluate how well the EM algorithm works if the M step does not do a full optimization, but rather does just a single iteration of the Newton-like method used by `nlm` (which you can do by giving `nlm` an argument of `iterlim=1`). Compare both the number of iterations needed for an accurate answer and amount of computer time needed (found using the `system.time` function).

You should test your programs on at least the two data sets used for assignment 2, found on CQUEST and the stats and CS computers in `/u/radford/ass2-a` and `/u/radford/ass2-b`. These files contain values for x_i and y_i one per line, suitable for reading using `read.table`. (These files are also available from the course web page.)

You should hand in your derivations of the formulas needed for the E and M steps, a listing of your program (suitably documented and formatted with consistent indentation), the output of your tests, and your discussion of the results.