STA 410/2102, Spring 2002 — Assignment #2

Due at **start** of class on March 19. 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.

The conditional distribution for a dichotomous response variable given one or more predictor variables is often modeled using *logistic regression*. If the two possible values for a response variable, y_i, are coded as -1 and $+1$, the model can be expressed as follows:

$$P(y_i \,|\, x_i) \quad = \quad 1 \,/\, (1 + \exp(-y_i(\beta_0 - \beta_1 x_i))) \quad = \quad g(y_i(\beta_0 - \beta_1 x_i)$$

where x_i is the value of the predictor variable for this case (we'll assume there is only one), β_0 and β_1 are the regression coefficients, and $g(z) = 1/(1 + \exp(-z))$ is the logistic function.

A problem with this model is that for a case where $|x_i|$ is very large, the probability of the response being $+1$ will be very close to either 0 or 1, unless β_1 is very close to zero. This may not be realistic. For example, suppose we are modeling whether or not a machine fails ($y_i = +1$ for failure, -1 for no failure) in terms of the temperature at which it is operated (x_i). Even if failure is strongly related to temperature, so that we expect β_1 to be far from zero, we nevertheless expect that even machines operated at a good temperature will sometimes fail.

A possible solution to this problem is to use a model in which the probability of a response being $+1$ can never be less than $\alpha/2$ or greater than $1-\alpha/2$. This model can be expressed as follows:

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

Estimates for α, β_0, and β_1 based on data $(x_1, y_1), \ldots, (x_n, y_n)$ might be found by maximum likelihood, using the following likelihood function:

$$L(\alpha, \beta_0, \beta_1) \quad = \quad \prod_{i=1}^{n} \Big[\alpha/2 \;+\; (1-\alpha) g(y_i(\beta_0 - \beta_1 x_i)) \Big]$$

Standard errors for the estimates can be based on the observed information. Since this model is not standard, finding maximum likelihood estimates and their standard errors may require special programming.

Your task is to solve this problem using Newton iteration. One problem with this approach is that α is restricted to the interval $[0, 1]$, which Newton iteration might not stay within. To solve this, you should use a parameter γ instead of α, replacing α by $g(\gamma)$. The range for γ is unrestricted. This solves the problem of invalid values, but may create a problem of a different sort if the maximum likelihood estimate for γ is at $\pm\infty$. (Note that it is also possible for estimates of β_0 or β_1 to be at $\pm\infty$.)

Newton iteration is not guaranteed to converge. After implementing the plain Newton method, you should investigate how much of a problem this is. You should then implement the following method that aims to overcome this problem: After computing the new point found by an iteration of the Newton method, check to see whether the likelihood at this new point is actually less than the likelihood at the current point. If it is, don't move to this new point. Instead, find a new point by adding the gradient of the likelihood at the current point to the current point, and check the likelihood at this new point. If it also is less than the likelihood of the current point, try adding half the gradient, and so forth (halving the distance moved at each stage), until a point where the likelihood is greater than or equal to the current likelihood is reached. You should investigate whether this modification increases reliability.

You should investigate how many iterations are needed to get accurate results. You can set the number of iterations manually, rather than implementing some automatic stopping rule. You should also investigate whether multiple local maxima of the likelihood can exist, as well as whether local minima and/or saddle points are a problem, and try to find starting values for the iteration that avoid any such problems.

You should test your programs on at least the two data sets 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 listings of your programs, suitably documented and formatted with consistent indentation, along with the output of your tests, and your discussion of the results.