7. FURTHER APPLICATIONS

7.1. COMPUTATIONAL ASPECTS

To evaluate the small sample approximations developed in the previous chapters we require the use of numerical software. The most critical aspect involves the solution for the saddlepoint, $\alpha(\mathbf{t})$. In the most general case, this involves solving the system

$$\int \psi_j(x,\mathbf{t}) \exp\left\{\sum_{i=1}^{p} \alpha_j(\mathbf{t})\psi_j(x,\mathbf{t})\right\} f(x)dx = 0 \quad j = 1,\cdots,p. \tag{7.1}$$

This is a non-linear system of equations which require an integration to evaluate the left hand side. Given the values for α, the computation of $c(\mathbf{t})$, $\Sigma(\mathbf{t})$, $A(\mathbf{t})$ also involve the evaluation of integrals. In all the examples for which we have done computations, x has been univariate. In principle, there is no difficulty with multivariate x, but the problems of integration become severe as the dimension of x increases.

In terms of computational effort, it is the solution of (7.1) which is the dominant feature. In the problems involving location/scale, the functions involved in (7.1) are continuous and piecewise differentiable and the resulting $\alpha_j(\mathbf{t})$'s have been smooth. This smoothness is very helpful in solving (7.1) over a grid of t since we are able to get good initial guesses for α based on values at adjacent grid points. For one dimensional problems, we have used the NAG (Numerical Algorithm Group) subroutine C05AJF. This procedure works well given a good initial guess and iterates using a secant method. The algorithm solves a sequence of problems $h(\alpha) - \theta_r h(\alpha_0)$ where $1 = \theta_0 > \theta_1 > \theta_2 \cdots > \theta_m = 0$ where α_0 is the initial guess. For each θ_r, a robust secant iteration is used based on the solution from earlier problems. If bounds are available on the solution α, a routine such as C05ADF may be used. In the case of location/scale problems α is two-dimensional. For the problem of Huber's Proposal 2, we used the NAG subroutines C05NBF or C05NCF. Both routines are suitable for the situation where the derivatives with respect to α are not provided. These routines are similar to the IMSL routines HYBRD and HYBRD1. It is necessary to choose initial estimates carefully using information from other grid points. It is possible to evaluate the derivatives of α and use a more reliable routine. However the derivatives are somewhat complicated and it is not clear that the extra coding required is worth the effort.

The numerical integration has been carried out using Gaussian quadrature. The particular NAG subroutine used is D01FBF. From experience, it appears that a 32 point quadrature procedure is necessary to get reliable results in (7.1) for t in the tail of the distribution.

The evaluation of tail areas has been carried out using the Lugannani-Rice approximation (6.3). The error in that approximation is about the same order as that resulting from a numerical integration of $f_n(\mathbf{t})$.

To give a sense of the computing time involved, we did several runs on a SUN4, 0S (4.2 Berkeley Unix) using the NAG subroutines mentioned above. The first two cases involved computation of $\alpha(t)$, $c(t)$, $s(t)$ for the mean, $\psi(x,t) = (x-t)$ for a uniform density on $[-1,1]$ and an extreme density. The results are as follows.

Uniform:	$t = 0[.005].99$	CPU time = 2.5 seconds
Extreme:	$t = -9[.1]2.5$	CPU time = 5.9 seconds

In a two dimensional problem, we solved for $\alpha_1(t)$ and $\alpha_2(t)$ in the case of the extreme density. For a total of 110 grid points (t_1 varying, t_2 fixed), the CPU time was 8.2 seconds.