STAT 604 Final Exam **Solution** **Fall, 1990 Dr. Newton**

Name: ____________________________

1. (12 points) How is -73.4236 represented as a `real*4`? How about 7,532 as an `integer*2`?

2. (12 points) How would you use Newton-Raphson to find a relative minimum of the function

$$S(\theta)=\sum_{i=1}^{n}(y_i-\theta_1X_{i1}-e^{\theta_2X_{i2}})^2,$$

where $\theta^T=(\theta_1,\theta_2)$ and we observe (y_i,X_{i1},X_{i2}) for $i=1,\ldots,n$. Assume you have a reasonable starting value for the process.

3. (8 points) What is wrong with the following TEX code (in order to generate the displayed equation in the previous question)?

```
$$
S(\theta)=\sum_{i=1}^n(y_i-\theta_1X_{i1}-e^{\theta_2X_{i2})^2,
$$
```

4. (10 points) Write down a recursive algorithm for finding $s_n^2=\sum_{i=1}^{n}(X_i-\bar{X})^2/(n-1)$ in terms of s_{n-1}^2 and X_n.

5. (8 points) What would be the output of the following Fortran code?

```
        dimension x(7,6)
        do 10 i=1,7
        do 10 j=1,6
10      x(i,j)=10*(i-1)+j
        call mprt(x,5,5)
        stop
        end
        subroutine mprt(a,n,m)
        dimension a(n,m)
        write(*,*) a(n,m)
        return
        end
```

6. (12 points) Write down in matrix form the multiple linear regression model, making sure to define the vectors and matrices involved and to specify their dimensions. Give the formula for the least squares estimators of the coefficients of the model, and those of the residual sum of squares and covariance matrix of the least squares estimators. How does one use the Modified Cholesky Decomposition to obtain these three quantities?

7. (14 points) Write an S function (call it `plotdray`) that will plot the pdf of the Rayleigh distribution

$$f(x) = \frac{1}{\alpha^2} x e^{-x^2/(2\alpha^2)}, \qquad x > 0,$$

from its `u1`th to `u2`th quantiles. Write functions `dray` and `qray` to use in the `plotdray` function. The arguments to `plotdray` should be `alpha`, `u1`, and `u2` which should be assigned default values 1.0, 0.01, and 0.99, respectively. You can assume that the user has already called `x11` or `postscript` prior to calling `plotdray`. Nice labels on the plot would be useful but not required.

8. (12 points) What is the relative absolute error in using the composite Simpson's rule (with four intervals) to approximate $\int_0^1 \cos \pi x \, dx$?

9. (12 points) What is the probability that a random point in the unit square is also in the unit circle? Use this to design a simulation that estimates π by generating random points in the unit square. How many points would you have to generate to be 95% sure that your estimate of π is within 0.01 of the true value?

Solutions

1. I made a slight error in describing `real*4`'s in class. I said to move the binary point to in front of the leftmost 1, when in fact it should be moved to the right of the leftmost 1. Thus for the example in class,

$$124.625_D = 64 + 32 + 16 + 8 + 4 + \frac{1}{2} + \frac{1}{8} = 1111100.101_B = 1.111100 \times 2^6,$$

which gives $b_{31} = 0$ since the number is positive, $b_{30} - b_{23} = 10000101$, which is $133_D = 6 + 127$, while $b_{22} - b_0 = 11110000000000000000000$ since you don't need the leading 1. Thus

$$124.625 = 0100\ 0010\ 1111\ 1000\ 0000\ 0000\ 0000\ 0000.$$

Now if we expand 73.4236 in powers of 2 down to 2^{-10}, we get

$$73.4236 \doteq 2^6 + 2^3 + 2^0 + 2^{-2} + 2^{-3} + 2^{-5} + 2^{-6} + 2^{-10},$$

and thus

$$-73.4236_D \doteq 1001001.0110110001_B = 1.0010010110110001 \times 2^6,$$

and $b_{31} = 1$ since the number is negative, $b_{30} - b_{23} = 10000101$ since the biased exponent is $133_D = 6 + 127$, while bit 22 through bit 0 are 00100101101100010000000, giving

$$-73.4236 \doteq 1100\ 0010\ 1001\ 0010\ 1101\ 1000\ 1000\ 0000.$$

Now

$$7532 = 2^{12} + 2^{11} + 2^{10} + 2^8 + 2^6 + 2^5 + 2^3 + 2^2,$$

which since the number is positive, we just do the binary representation

$$7532 = 0001\ 1101\ 0110\ 1100.$$

2. To do the Newton-Raphson procedure we need the gradient vector g and Hessian matrix H whose elements are given by

$$
\begin{aligned}
g_1 &= \frac{\partial S(\theta)}{\partial \theta_1} = -2\sum_{i=1}^{n} X_{i1}(y_i - \theta_1 X_{i1} - e^{\theta_2 X_{i2}}) \\
g_2 &= \frac{\partial S(\theta)}{\partial \theta_2} = -2\sum_{i=1}^{n} X_{i2}e^{\theta_2 X_{i2}}(y_i - \theta_1 X_{i1} - e^{\theta_2 X_{i2}}) \\
H_{11} &= \frac{\partial^2 S(\theta)}{\partial \theta_1^2} = 2\sum_{i=1}^{n} X_{i1}^2 \\
H_{12} &= H_{21} = \frac{\partial^2 S(\theta)}{\partial \theta_1 \partial \theta_2} = 2\sum_{i=1}^{n} X_{i1}X_{i2}e^{\theta_2 X_{i2}} \\
H_{22} &= \frac{\partial^2 S(\theta)}{\partial \theta_2^2} = 2\sum_{i=1}^{n} X_{i2}^2 e^{\theta_2 X_{i2}}\left[e^{\theta_2 X_{i2}} - (y_i - \theta_1 X_{i1} - e^{\theta_2 X_{i2}})\right]
\end{aligned}
$$

Then the iteration is of the form $\theta_{k+1} = \theta_k - H_k^{-1} g_k$, where g_k and H_k are g and H evaluated at θ_k.

3. The right curly bracket is missing that should close the exponent on the `e`.

4. Perhaps the most useful recursive formula is

$$s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(X_n - \bar{X}_{n-1})^2,$$

which we can get by writing

$$
\begin{aligned}
(n-1)s_n^2 &= \sum_{i=1}^{n}(X_i - \bar{X}_n)^2 = \sum_{i=1}^{n}\left[(X_i - \bar{X}_{n-1}) + (\bar{X}_{n-1} - \bar{X}_n)\right]^2 \\
&= \sum_{i=1}^{n-1}(X_i - \bar{X}_{n-1})^2 + (X_n - \bar{X}_{n-1})^2 + n(\bar{X}_{n-1} - \bar{X}_n)^2 + 2(\bar{X}_{n-1} - \bar{X}_n)\underbrace{\sum_{i=1}^{n}(X_i - \bar{X}_{n-1})}_{=0+(X_n - \bar{X}_{n-1})} \\
&= (n-2)s_{n-1}^2 + (X_n - \bar{X}_{n-1})^2 + n(\bar{X}_{n-1} - \bar{X}_n)^2 + 2(\bar{X}_{n-1} - \bar{X}_n)(X_n - \bar{X}_{n-1}).
\end{aligned}
$$

But $X_n - \bar{X}_{n-1} = -n(\bar{X}_{n-1} - \bar{X}_n)$ and so

$$
\begin{aligned}
(n-1)s_n^2 &= (n-2)s_{n-1}^2 + n^2(\bar{X}_{n-1} - \bar{X}_n)^2 + n(\bar{X}_{n-1} - \bar{X}_n)^2 - 2n(\bar{X}_{n-1} - \bar{X}_n)^2 \\
&= (n-2)s_n^2 + (n^2 - n)(\bar{X}_{n-1} - \bar{X}_n)^2,
\end{aligned}
$$

which means

$$s_n^2 = \frac{n-2}{n-1}s_{n-1}^2 + n(\bar{X}_{n-1} - \bar{X}_n)^2 = \frac{n-2}{n-1}s_{n-1}^2 + \frac{1}{n}(X_n - \bar{X}_{n-1})^2.$$

5. In the main program,

$$x = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 11 & 12 & 13 & 14 & 15 & 16 \\ 21 & 22 & 23 & 24 & 25 & 26 \\ 31 & 32 & 33 & 34 & 35 & 36 \\ 41 & 42 & 43 & 44 & 45 & 46 \\ 51 & 52 & 53 & 54 & 55 & 56 \\ 61 & 62 & 63 & 64 & 65 & 66 \end{bmatrix},$$

but since matrices are passed to subroutines in Fortran by columns and the dimension in the subroutine is five by five, we have

$$a = \begin{bmatrix} 1 & 51 & 32 & 13 & 63 \\ 11 & 61 & 42 & 23 & 4 \\ 21 & 2 & 52 & 33 & 14 \\ 31 & 12 & 62 & 43 & 24 \\ 41 & 22 & 3 & 53 & 34 \end{bmatrix},$$

and thus the program would print out the (5,5) element of a which is 34.

6. The multiple linear regression model can be written in matrix form as

$$y = X\beta + \epsilon,$$

where y is an $(n \times 1)$ vector of the values of the dependent variable for the n "experimental units" being studied, X is an $(n \times m)$ matrix whose ith row consists of the values of the m independent variables for the ith unit, β is a vector containing the true values of the m coefficients of the model, and ϵ is the $(n \times 1)$ vector containing random errors for the n units. The ϵ's are usually assumed to be iid $N(0, \sigma^2)$ random variables. The quantities most often desired for such a model are the least squares estimates $\hat{\beta}$ of the coefficients and their standard errors

$$\hat{\beta} = (X^T X)^{-1} X^T y, \qquad se(\hat{\beta}_j) = \sqrt{\hat{V}_{jj}},$$

where

$$\hat{V} = \hat{\sigma}^2 (X^T X)^{-1}, \qquad \hat{\sigma}^2 = \frac{RSS}{n-m},$$

and $RSS = e^T e$, where $e = y - X\hat{\beta}$. If we find the Modified Cholesky Decomposition $X^T X = LDL^T$ of $X^T X$, we can write the normal equations $X^T X \hat{\beta} = X^T y$ as

$$LDL^T \hat{\beta} = X^T y,$$

and we can solve for $\hat{\beta}$ by first solving the lower triangular system $L\hat{\theta} = X^T y$, then the diagonal system $Dz = \hat{\theta}$, and finally the upper triangular system $L^T \hat{\beta} = z$. We can get $(X^T X)^{-1} = L^{-T} D^{-1} L^{-1}$ easily since it is easy to find the inverse of a triangular matrix. Finally, obtaining RSS is still best done by $e^T e$.

7. To be able to plot the Rayleigh pdf for arbitrary quantiles we need the quantile function. The cdf is given by

$$F(x) = \int_0^x \frac{1}{\alpha^2} y e^{-y^2/2\alpha^2}\, dy = 1 - e^{-x^2/2\alpha^2},$$

which gives (letting $q = Q(u)$) $F(q) = 1 - e^{-q^2/2\alpha^2}$, which is u if

$$Q(u) = [-2\alpha^2 \log(1-u)]^{1/2}.$$

The code given below will graph the pdf.

```
plotdray <- function(alpha=1,u1=.01,u2=.99)
#-----------------------------------------------------------
#
#   Function to plot the pdf of the Rayleigh distribution
#   having parameter alpha from its u1 th to u2 th percentiles.
#
#------------------------------------------------------------
{
   x <- seq(qray(u1,alpha),qray(u2,alpha),length=101)
   plot(x,dray(x,alpha),xlab="x",ylab="f(x)",type="l",
        main=paste("Rayleigh (",round(alpha,2),") pdf "))

   invisible()  # This makes it so plotdray returns nothing

}
dray <- function(x,alpha=1)
#-----------------------------------------------------------
#
#   Function to calculate the pdf of the Rayleigh distribution
#   having parameter alpha
#
#------------------------------------------------------------
{
   return(exp(-2*log(alpha)+log(x)-x^2/(2*alpha^2)))
}
qray <- function(u,alpha=1)
#-----------------------------------------------------------
#
#   Function to calculate the quantile function of the Rayleigh
#   distribution having parameter alpha from its u1 th to u2 th
#   percentiles.
#
#------------------------------------------------------------
{
   return(sqrt(-2*alpha^2*log(1-u)))
}
```

8. The exact value of the integral can be obtained as

$$\int_0^1 \cos \pi x \, dx = -\frac{\sin \pi x}{\pi}\Big|_{x=0}^{x=1} = 0,$$

while the composite Simpson's Rule gives (with $h = 1/4$),

$$\begin{aligned} \int_0^1 \cos \pi x \, dx &\doteq \frac{h}{3}\left[f(0) + 4f(1/4) + 2f(1/2) + 4f(3/4) + f(1)\right] \\ &= \frac{1}{12}\left[1 + 4f(1/4) + 0 - 4f(1/4) - 1\right] \\ &= 0, \end{aligned}$$

and so there is no error.

9. The probability that a randomly generated point in the unit square is also in the unit circle is just the ratio of the areas of the circle and the square which is $p = \pi/4$. Thus to estimate p, we can generate n points in the random square (each of which can be generated as (V_1, V_2) where $V_i = 2(U_i - .5)$ and U_1

and U_2 are iid U(0,1) random variables) and let X be the number of the points also in the unit circle. Thus X is a Binomial random variable with n trials and success probability p. Thus $\hat{p} = X/n$, and a natural estimator of π is $\hat{\pi} = 4\hat{p}$, which asymptotically has the normal distribution with mean π and variance

$$\sigma_{\hat{\pi}}^2 = 16\sigma_{\hat{p}}^2 = 16\frac{p(1-p)}{n}.$$

Now

$$\Pr(|\hat{\pi} - \pi| \leq .01) = \Pr(-.01 \leq \hat{\pi} - \pi \leq .01) = \Pr(-\frac{.01}{\sigma_{\hat{\pi}}} \leq \frac{\hat{\pi} - \pi}{\sigma_{\hat{\pi}}} \leq \frac{.01}{\sigma_{\hat{\pi}}})$$

$$\doteq \Pr(-\frac{.0025}{\sqrt{p(1-p)/n}} \leq Z \leq \frac{.0025}{\sqrt{p(1-p)/n}}),$$

where $Z \sim N(0,1)$. Thus for this probability to be the desired 0.95, we need

$$\frac{.0025}{\sqrt{p(1-p)/n}} = Z_{.025} = 1.96,$$

and solving this for n (recalling that $p = \pi/4$ and rounding n up to the nearest integer) we have $n = 103,599$.