Examples Guide: Chapters 2 – 6
Ex. 2.1.1: Demonstration of a DSP for Production Functions (C2DSP.doc)

Program Background:

The two production functions considered in this demonstration are relatively simple and have well-known properties. The square-root production function

\[ y = a + b \sqrt{x} + cx + e \]

expresses the quantity produced as a function of the variable input x plus noise (e). The noise term represents random influences on the production process and is assumed to be \( e \sim N(0, \sigma^2) \). For purposes of the demonstration, the variable input is generated as a Uniform(0, 2) pseudorandom variable. The production parameter \( a > 0 \) represents the level of production generated from fixed factors, and the parameters \( b \) and \( c \) determine the marginal impact of the variable input. For production to occur under profit maximization, the marginal product and the change in marginal product should have the following characteristics

\[
\frac{\partial y}{\partial x} = \frac{b}{2 \sqrt{x}} + c > 0 \quad \text{and} \quad \frac{\partial^2 y}{\partial x^2} = -\frac{b}{4} x^{-3/2} < 0
\]

The parameters should satisfy \( b > 0 \) and \( c > -0.5 b x^{-1/2} \) to represent profit maximizing production. In this program, GAUSS allows the user to specify \( 0 < a < 10, \ 0 < b < 10, \) and \(-10 < c < 10 \) in order to examine the characteristics of production outside the profit maximizing region.
The Cobb-Douglas production function

\[ y = K^a L^b V \]

specifies the quantity produced as a function of capital (K) and Labor (L) plus noise (V). The noise term V = exp(e) is assumed to be Lognormal(0, \sigma^2) with mean \( \exp\left(\frac{\sigma^2}{2}\right) \) and variance \( \exp\left(2\sigma^2\right) - \exp\left(\sigma^2\right) \) such that \( e \sim N\left(0, \sigma^2\right) \). For purposes of the demonstration, K and L are generated as independent Uniform(0,2) pseudorandom variables. Further, note that the Cobb-Douglas model is linear in the parameters (a and b) after transformation by the natural logarithm (if you don’t immediately understand this point, glance ahead to section 3.1.2 in the book).

The Cobb-Douglas production parameters a and b determine the relative importance of the inputs in production of the output. For fixed values of L and b, the marginal products of capital and labor are

\[ \frac{\partial y}{\partial K} = aK^{a-1}L^b e > 0 \quad \text{and} \quad \frac{\partial y}{\partial L} = b(b-1)K^aL^{b-1}e < 0 \]

The sufficient conditions for the existence of a solution to the firm’s profit maximization problem are:

\[ \frac{\partial^2 y}{\partial K^2} = a(a-1)K^{a-2}L^b e < 0 \quad \text{and} \quad \frac{\partial^2 y}{\partial L^2} = b(b-1)K^aL^{b-2}e < 0 \]

and

\[ \left(\frac{\partial^2 y}{\partial K^2}\right)\left(\frac{\partial^2 y}{\partial L^2}\right)^2 - \left(\frac{\partial^2 y}{\partial K \partial L}\right)^2 = ab(1-a-b)K^{2a-2}L^{2b-2}e > 0 \]

where

\[ \frac{\partial^2 y}{\partial K \partial L} = ab(a-1)(b-1)K^{a-1}L^{b-1}e \]
These conditions imply that \( 0 < a < 1 \) and \( 0 < b < 1 \) are required for production to occur in the profit maximizing region. The returns to scale of production for the Cobb-Douglas model is the sum of parameters \( a \) and \( b \).

**User Input:**

The user selects a production function (Cobb-Douglas or square-root) and the associated location and scale parameters. If the user prefers not to make these choices, GAUSS randomly selects a model with a set of default parameter values.

**Program Output:**

For the square root production function, GAUSS presents the user with a graph of the simulated input and output values as well as the associated regression function (the signal component). For the Cobb-Douglas production function, the simulated output values are plotted in two dimensions by holding one input fixed (at the sample average of the generated values) and allowing the other to vary. The regression surface (signal) is plotted in three dimensions (capital, labor, output), and the two-dimensional isoquants of the production function are also provided. The user then has the option of continuing with this model, selecting another model, or exiting the program.
Ex. 2.2.2: Analyzing Noise Component

Assumptions (C2Noise.gss)

Program Background:

A number of assumptions are associated with the standard forms of regression models having an additive noise component. In particular, Assumptions 2.1 and 2.2 (see section 2.2 in the book) imply that the noise components satisfy

1. $\mathbb{E}[\varepsilon] = 0$ (the unconditional expected value of the noise vector is the n-dimensional null vector)

2. $\text{cov}(\varepsilon) = \sigma^2 I$ (the noise components are uncorrelated and homoscedastic)

This GAUSS program generates $n = 30$ pseudorandom noise observations and plots the sequence of outcomes. The generated outcomes represent draws from a normal noise process with one of the following characteristics:

1. $\varepsilon \sim N(0, I)$ (following Assumptions 2.1 and 2.2)

2. $\text{cov}(\varepsilon) = I$ but $\mathbb{E}[\varepsilon] = 0.75 \times I$ (non-zero mean)

1. $\varepsilon \sim N(0, \Sigma)$ where $\Sigma$ is a diagonal matrix with elements $\Sigma[i, i] = i^2 / 10$

   (heteroscedastic with increasing variance)

4. $\varepsilon_i = \rho \varepsilon_{i-1} + v_i$ where $v \sim N(0, I)$ and $\rho$ is pseudorandom Uniform(-1,1) (first-order autocorrelated noise components)

5. $\varepsilon \sim N(0, \gamma I)$ for the first 15 observations and $\varepsilon \sim N(0, I)$ for the last 15
observations where $\gamma$ is pseudorandom Uniform(0,2) (heteroscedastic with a one-time shift in variance)

GAUSS randomly selects the true noise process from one of these five candidates each time the program is run.

**User Input:**

After plotting the data, the GAUSS program lists the five possibilities for the true noise. The user selects the alternative that best matches the observed noise components.

**Program Output:**

After the user selects a model for the noise component, GAUSS states the correct form of the noise component model and indicates if the user’s answer was correct or incorrect. The GAUSS program then asks if the game should be continued with another sample, and the user may repeat the game as many times as desired.
Ex. 3.2.1: Likelihood Function for a Simple Regression Model (C3Simple.gss)

Program Background:

The purpose of the program is to illustrate the likelihood function for a simple linear regression model \( y_i = \beta + \varepsilon_i \) where \( \varepsilon_i \sim N(0,1) \). As such, the location parameter \( \beta \) is the only unknown parameter in the model. The normal likelihood function for a random sample of \( n \) observations generated from this DSP is

\[
L(\beta; y) = (2\pi)^{-n/2} \exp \left[ -\frac{1}{2} \sum_{i=1}^{n} (y_i - \beta)^2 \right]
\]

Note that the likelihood function takes identical form to the joint PDF of \( Y \) but is viewed as a function of \( \beta \) (conditional on the observed outcomes, \( y \)). By choosing \( \beta \) to maximize \( L(\beta; y) \), we can compute the maximum likelihood (ML) estimator of \( \beta \). For this problem, the MLE is simply \( \hat{\beta} = \bar{y} \), the sample average of the observations.

User Input:

The user selects the sample size (5 < \( n \) < 500) and the value of the location parameter (-10 < \( \beta \) < 10) for the observations generated from the normal linear regression model.
**Program Output:**

After generating the n pseudorandom outcomes, GAUSS plots $L(\beta; y)$ and reports the true value of $\beta$ (selected by the user) and the estimate $\hat{\beta}$. The user should find that the peak of $L(\beta; y)$ coincides with the ML estimate.
EX. 3.2.2: Sampling Properties of the ML Estimator of $\beta$ (C3Distn.gss)

Program Background:

This program uses a Monte Carlo experiment to simulate the mean, variance, and sampling distribution for the maximum likelihood estimator of the location parameter $\beta$. Although these properties may be analytically determined for the normal linear regression model, this exercise should demonstrate to the user that Monte Carlo experiments can quickly and accurately represent the sampling properties of an estimator.

User Input:

For the experiment, the user is asked to select the number of Monte Carlo experiments (1, 2, or 4) to conduct as well as the sample size (3 < n < 1000) for each case. Then, the program requests the intercept (-10 < $\beta_1$ < 10), slope (-10 < $\beta_2$ < 10), and noise variance (0 < $\sigma^2$ < 5) parameters for a simple linear regression model.

$$y_i = \beta_1 + \beta_2 x_i + \epsilon_i$$

The explanatory variable $x_i$ is generated from the N(10,1) distribution and is held fixed in repeated samples for the Monte Carlo experiment. The user also selects the number of Monte Carlo trials (10 < $m$ < 1000) used in each experiment.
**Program Output:**

Given the selected model parameters, the program generates m sets of data for the Monte Carlo simulation exercise. For each replicated sample, the program computes the least squares estimates of $\beta_1$ and $\beta_2$. For each sample size selected by the user, GAUSS reports the sample average and variance of the Monte Carlo outcomes plus the actual mean and variance of the estimator. The user should find that the sample moments of the Monte Carlo trials approach the actual moments as the sample size (n) and the number of trials (m) increase.

GAUSS then computes the empirical distribution function (EDF) of the Monte Carlo estimates to compare with the CDF of the actual sampling distribution. Let $\hat{b}_{jk}$ denote the ML estimate of the slope parameter $\beta_k$ computed from the jth Monte Carlo sample. Then, the program forms the empirical distribution function (EDF) for $\hat{b}_{jk}$ based on the Monte Carlo estimates

$$\hat{F}_{mk}(z) = \frac{\#\{\hat{b}_{jk} : \hat{b}_{jk} \leq z\}}{m}$$

The notation $\#\{a: a < b\}$ represents the number of elements in the set of a values that satisfy the condition $a < b$. For comparison purposes, the EDF is plotted with the actual normal distribution function for the ML estimator $F(\hat{\beta}_k)$. Given that $\hat{F}_{mk} \xrightarrow{p} F(\hat{\beta}_k)$ as $m \to \infty$ and $n \to \infty$, the user should find that the EDF approaches the normal CDF.
EX. 3.3.1: ML Estimation by the Newton-Raphson Algorithm (C3MLE.gss)

Program Background:

The purpose of the GAUSS program is to demonstrate that the solution to the ML estimation problem computed by numerical optimization methods is effectively identical to the analytical solution derived in Equations (3.2.7) and (3.2.8). The data for the demonstration are generated for a normal linear regression model

\[
y = 10 + 3x_2 - 2x_3 + 4x_4 + e
\]

Here, \(x_2, x_3, x_4,\) and \(e\) are generated as pseudorandom \(N(0,1)\) variables.

To compute the ML estimates of \(\beta\) and \(\sigma^2\), the GAUSS program uses the iterative Newton-Raphson optimization algorithm

\[
\hat{\theta}_1 = \hat{\theta}_0 - \left[ \frac{\partial^2 \ln L(\hat{\theta}_0; y, x)}{\partial \theta \partial \theta'} \right]^{-1} \left[ \frac{\partial \ln L(\hat{\theta}_0; y, x)}{\partial \theta} \right]
\]

As explained later in Chapter 8, section 8.13, the next iterative estimate \(\hat{\theta}_1\) is projected from a quadratic approximation to \(\ln L(\theta; y, x)\) evaluated at the preceding estimate, \(\hat{\theta}_0\).

The algorithm is implemented in the NRaphson() GAUSS procedure (included in the EF library) and requires additional GAUSS procedures for the log-likelihood function (ignoring the constant terms), score function, and hessian matrix, which are based on Equations (3.2.4), (3.2.5), and (3.2.6).
Although the hessian matrix is easily formed for this simple estimation problem, an alternative method based solely on the observed score function may be used to compute the variance estimates. From Equation (3.3.2), we know that

\[
I(\theta) = -E \left[ \frac{\partial^2 \ln L(\theta; y, x)}{\partial \theta \partial \theta'} \right] = E \left[ \frac{\partial \ln L(\theta; y, x)}{\partial \theta} \frac{\partial \ln L(\theta; y, x)}{\partial \theta'} \right]
\]

If the analytical form of the hessian matrix is difficult to compute, we can form an estimate of the information matrix based on the sample average

\[
I_{jk}(\hat{\theta}) = n^{-1} \sum_{i=1}^{n} \frac{\partial \ln L(\hat{\theta}; y_i, x_i)}{\partial \theta_j} \frac{\partial \ln L(\hat{\theta}; y_i, x_i)}{\partial \theta_k}
\]

The estimate is the sample average of the outer product of the observed score function, which is commonly known as the outer-product-of-the-gradient (OPG) estimate. The inverse matrix \( [I(\hat{\theta})]^{-1} \) is a consistent estimator of \( \text{cov}(\hat{\theta}) \). In more complicated ML estimation problems, the OPG procedure is often used because it does not requires computation of the hessian matrix.

**User Input:**

The user selects the sample size \((10 < n < 500)\) for the observations generated from the normal linear regression model.

**Program Output:**

GAUSS first explains the auxiliary procedures used to compute the ML estimates by the
NRaphson() procedure. For the pseudorandom normal sample ($y$), GAUSS reports the parameter estimates for $\beta$ and $\sigma^2$ based on the analytical solution to the ML problem (Equations (3.2.7) and (3.2.8)) and the numerical solution computed by the Newton-Raphson algorithm (including the number of iterations). The program also reports the estimated $\text{cov}(\hat{\theta})$ based on the analytical inverse information matrix (Equation (3.3.4)) and on the hessian matrix reported by the NRaphson() procedure. In both cases, the user should find that the analytical and numerical estimates of the parameters and the estimator variances are identical to several decimal places. Finally, GAUSS compares the estimates of $\text{cov}(\hat{\theta})$ based on the hessian matrix and the outer-product-of-the-gradient (OPG) methods. The user should find that the estimates are similar but distinct, and the observed differences will diminish as the sample size ($n$) increases.
Program Background:

This GAUSS program uses a Monte Carlo experiment to demonstrate the consistency (convergence in probability) of the ML estimator of the location parameter $\beta$ for a simple linear regression model

$$Y_i = \beta_1 + \beta_2 x_i + \epsilon_i.$$  

The noise components $(\epsilon_i)$ are generated as pseudorandom $N(0, \sigma^2)$ outcomes. The explanatory variable $x_i$ is generated from the $N(10,1)$ distribution and held fixed in repeated samples for the Monte Carlo experiment.

Under this regression model, the ML estimator converges in mean square $\hat{\beta} \xrightarrow{ms} \beta$ and in probability $\hat{\beta} \xrightarrow{p} \beta$. By the definition of mean square convergence (section E1.6.3), the elements of the bias vector, $\text{Bias}(\hat{\beta}) = E[\hat{\beta}] - \beta$, and $\text{cov}(\hat{\beta})$ converge to zero as $n \to \infty$. To demonstrate this property, the GAUSS program simulates the bias and variance of the ML estimator across replicated samples. In particular, let $\hat{\beta}_j$ represent the ML estimate of $\beta$ for the jth Monte Carlo trial. The Monte Carlo bias estimate is

$$\left( m^{-1} \sum_{j=1}^{m} \hat{\beta}_j \right) - \beta$$
The Monte Carlo estimate of $\text{cov}(\hat{\beta})$ is

$$(m-1)^{-1} \sum_{j=1}^{m} (\hat{\beta}_j - \bar{\beta}_m)(\hat{\beta}_j - \bar{\beta}_m)' \quad \text{where} \quad \bar{\beta}_m = \frac{1}{m} \sum_{j=1}^{m} \hat{\beta}_j$$

**User Input:**

For the experiment, the user selects the number of Monte Carlo experiments (1, 2, ..., 8) conducted as well as the sample size (10 < n < 1000) for each case. Then, the program asks the user to select values for the intercept (-10 < $\beta_1$ < 10), slope (-10 < $\beta_2$ < 10), and noise variance (0 < $\sigma^2$ < 5) parameters. The user also selects the number of Monte Carlo trials (1 < m < 1000) used in each experiment.

**Program Output:**

Given the specified model parameters, the GAUSS program computes the estimated bias vector and covariance matrix for the ML estimator. For each sample size (n) selected by the user, GAUSS reports the simulated bias and variance for $\beta_1$ and $\beta_2$. The user should find that the simulated values approach zero as $n \to \infty$, which demonstrates but does not prove convergence in probability. Next, GAUSS forms boxplots for the set of m Monte Carlo estimates of $\beta_1$ and $\beta_2$. The user should find that the boxes and whiskers in the plots collapse on the true values of $\beta$ (selected by the user) as $n \to \infty$. 
EX. 3.4.2: Unbiasedness and Consistency of $S^2$ 

(C3S2.gss)

Program Background:

This GAUSS program uses a Monte Carlo experiment to demonstrate the consistency (convergence in probability) of the ML estimator of the scale parameter $\sigma^2$ for a simple linear regression model

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

The noise components ($\epsilon_i$) are generated as pseudorandom $N(0, \sigma^2)$ outcomes. The explanatory variable $x_i$ is generated from the $N(10,1)$ distribution and held fixed in repeated samples for the Monte Carlo experiment.

Under this regression model, the unbiased estimator $S^2$ converges in mean square $S^2 \overset{ms}{\rightarrow} \sigma^2$ and in probability $S^2 \overset{p}{\rightarrow} \sigma^2$. By the definition of mean square convergence (section E1.6.3), the bias, $\text{Bias}(S^2) = E[S^2] - \sigma^2$, and $\text{var}(S^2)$ converge to zero as $n \to \infty$. To demonstrate this property, the GAUSS program simulates the bias and variance of $S^2$ across replicated samples. In particular, let $s_j^2$ represent the ML estimate of $\sigma^2$ for the $j$th Monte Carlo trial. The Monte Carlo bias estimate is

$$\left( m^{-1} \sum_{j=1}^{m} s_j^2 \right) - \sigma^2$$

The Monte Carlo estimate of $\text{var}(S^2)$ is
\[(m-1)^{-1} \sum_{j=1}^{m} \left( s_j^2 - \bar{s}_m^2 \right) \left( s_j^2 - \bar{s}_m^2 \right) \] where \[ \bar{s}_m^2 = m^{-1} \sum_{j=1}^{m} s_j^2 \]

**User Input:**

For the experiment, the user selects the number of Monte Carlo experiments (1, 2, \ldots, 8) conducted as well as the sample size (10 < n < 1000) for each case. Then, the program asks the user to select values for the intercept (-10 < \( \beta_1 < 10 \)), slope (-10 < \( \beta_2 < 10 \)), and noise variance (0 < \( \sigma^2 < 5 \)) parameters. The user also selects the number of Monte Carlo trials (1 < m < 1000) used in each experiment.

**Program Output:**

Given the specified model parameters, the GAUSS program computes the estimated bias and variance for \( S^2 \). For each sample size (n) selected by the user, GAUSS reports the simulated bias and variance. The user should find that the simulated values approach zero as \( n \to \infty \), which demonstrates but does not prove convergence in probability. Next, GAUSS forms boxplots for the set of m Monte Carlo estimates of \( \sigma^2 \). The user should find that the boxes and whiskers in the plots collapse on the true value of \( \sigma^2 \) (selected by the user) as \( n \to \infty \).
Ex. 3.4.3: Sampling Properties of the Unbiased Variance Estimator, $S^2$ (C3CLT.gss)

Program Background:

This GAUSS program uses parametric and nonparametric bootstrap resampling methods to simulate the mean, variance, and sampling distribution for the unbiased estimator, $S^2$. In the parametric case, the assumed normality of the noise component $\varepsilon \sim N(0, \sigma^2 I)$ implies $Y \sim N(x\beta, \sigma^2 I)$ (see section 3.2.1). The unknown parameters in the distribution of $Y$ are replaced with the estimates, $\hat{\beta}$ and $s^2$, and the $n$ bootstrap replicates $y^*$ are drawn from the $N(\hat{x}\hat{\beta}, s^2 I)$ distribution. In the nonparametric case, the sampling process is replicated by drawing $n$ elements $\varepsilon^*$ from the residual vector $\hat{\varepsilon}$ with replacement. Then, the vector of bootstrap outcomes for the dependent variable $y^*$ is formed as $y^* = \hat{x}\hat{b} + \varepsilon^*$ (see section A.3.2 for more details).

Given values for the model parameters (selected by the user), the GAUSS program generates $n$ initial outcomes $y$ and computes the parameter estimates $\hat{b}$ and $s^2$. GAUSS then generates $n$ replicated outcomes $y^*$ of the initial sample by the parametric and nonparametric resampling algorithms. The resampling procedures are repeated $m$ times, and GAUSS uses the unbiased estimator $S^2$ to compute the estimated error variance for each replicated sample. GAUSS then computes the sample mean, sample variance, and empirical distribution function (EDF) for the parametric and nonparametric bootstrap
replicates.

To evaluate the performance of the estimator, recall that the parameter estimates from the initial sample \((y)\) are viewed as the true parameter values in the bootstrap world. Accordingly, GAUSS compares the sample moments and EDF for each set of bootstrap replicates to the estimates based on the in properties of the limiting normal distribution, 

\[ S^2 \sim N\left(\sigma^2, 2n^{-1}\sigma^4\right). \]

That is, the sample average of the bootstrap outcomes is compared to \(s^2\), the sample variance is compared to \(2n^{-1}s^4\), and the EDF is plotted with the CDF of the \(N\left(s^2, 2n^{-1}s^4\right)\) distribution.

**User Input:**

For the demonstration, the user is asked to select the number of bootstrap simulation experiments (1, 2, or 4) to conduct as well as the sample size (10 < n < 1000) and number of bootstrap trials (10 < m < 1000) for each case. Then, the GAUSS program asks the user to select the intercept \((\beta_1)\), slope \((\beta_2)\), and noise variance \((\sigma^2)\) parameters for a simple linear regression model

\[ y_i = \beta_1 + \beta_2 x_i + \varepsilon_i \]

The noise component is pseudorandom \(N\left(0, \sigma^2\right)\), and the explanatory variable \(x_i\) is generated from the \(N(10,1)\) distribution and is held fixed in repeated samples.

**Program Output:**

After GAUSS completes the bootstrap simulation exercises, the sample average and
variance of the parametric and nonparametric outcomes are reported with the estimated mean and variance (computed from the initial sample, y) of the limiting normal distribution, \( S^2 \sim \mathcal{N}(\sigma^2, 2n^{-1}\sigma^4) \). The user should find that the sample moments approach the estimated moments as n and m become large. Then, GAUSS plots the EDF and the estimated limiting normal CDF for each of the selected sample sizes (n). The EDF should approach the normal CDF as n and m become large. The user may repeat the demonstration as many times as desired.
EX. 3.9.1: Comparison of Estimators under Quadratic Risk (C3Jstein.gss)

Program Background:

James and Stein proposed the nonlinear shrinkage estimator
\[ \hat{\beta}^s = \left[ 1 - (k - 2)/\hat{\beta}'\hat{\beta} \right] \hat{\beta} \]

where \( \hat{\beta} \) is the ML estimator of \( \beta \) for the linear regression model. In effect, the James-Stein estimator forms a linear combination of the unrestricted ML estimator \( \hat{\beta} \) and the restricted ML (RML) estimator subject to the constraint \( \beta = 0 \). The weight assigned to the ML estimator in the combination is the shrinkage factor \( 1 - (k - 2)/\hat{\beta}'\hat{\beta} \). Note that the James-Stein estimator is not a convex combination of the two ML estimators because the weight can be negative. To avoid negative shrinkage factors, we may truncate the shrinkage factor at zero, which yields a variant of the James-Stein estimator known as the positive-part Stein rule, \( \beta^*_s = \max(0, 1 - (k - 2)/\hat{\beta}'\hat{\beta}) \hat{\beta} \).

As noted in section 3.9, the quadratic risk functions of the alternatives may be compared for a very simple version of the normal linear regression model in which \( \sigma^2 = 1 \) and the columns of \( x \) are orthogonal such that \( x'x = I_k \). In this case, the quadratic risk of the unrestricted ML estimator is \( \rho(\beta, \hat{\beta}) = k \), which is constant for all values of \( \beta \). The quadratic risk of the restricted ML estimator is simply \( \beta'\beta \), and the risk function is linear when plotted against the squared norm of the true parameter vector (see Figure 3.1 in the
book). The quadratic risk function for $\beta^s$ is 
$$\rho(\beta, \beta^s) = k - (k - 2)E\left[1/\hat{\beta}'\hat{\beta}\right]$$
where $1/\hat{\beta}'\hat{\beta}$ is distributed as an inverted Chi-square($k, \lambda$) random variable with noncentrality parameter $\lambda = \hat{\beta}'\beta/2$. Although the expected value is difficult to compute, Judge and Bock (1978) show that $\rho(\beta, \beta^s) \leq \rho(\hat{\beta}, \hat{\beta}) = k$ for all $\beta$. Further, the risk function of $\beta^s$ is also very difficult to analytically evaluate, but researchers have proven that $\beta^s$ risk-dominates $\beta^s$ such that $\rho(\beta, \beta^s) \leq \rho(\beta, \beta^s)$ for all $\beta$.

Users should also note the method used to generate the orthogonal matrix of explanatory variables, $x$ such that $x'x = I_k$. GAUSS first generates a $(40 \times 4)$ matrix of $N(0,1)$ pseudorandom numbers and uses the svd2() procedure to compute the singular value decomposition (SVD), $x = usv'$. By replacing the singular values in the matrix $s$ with ones, we can form a matrix $x$ that is orthogonal. For more details, refer to the SVD section in the Matrix Review Manual.

**User Input:**

The user selects the number of Monte Carlo simulation trials to be conducted for the experiment ($50 < m < 5000$).

**Program Output:**

GAUSS presents the user with a graph of the simulated quadratic risk functions versus the squared norm of the true parameter vector ($\beta'\beta$) for the four estimators. The user should
find that the plot is similar to the risk functions presented in Figure 3.1 in the book. That is, the risk of the ML estimator is roughly constant across the parameter space, and the quadratic risk of the restricted ML estimator increases linearly from zero. The risk function for the James-Stein estimator is everywhere below the ML risk, and the risk function for the positive-part Stein rule is everywhere below the James-Stein risk. The user then has the option of repeating the simulation exercise or exiting the program.
EX. 4.3.1: Conduct and Power of the GLR F-Test of $c\beta = r$ (C4GLR_F.gss)

Program Background:

This program demonstrates the conduct of the F-test of the linear restriction $c\beta = r$ for the normal linear regression model $y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i$ where $i = 1, \ldots, 40$. The noise component is generated pseudorandom $\varepsilon \sim N(0, I)$, and the explanatory variables $(x_2, x_3, \text{and } x_4)$ are pseudorandom $N(0,1)$ and held fixed in repeated samples. The first objective of the program is to conduct a test of $H_0: \beta_2 = r_1$ and $\beta_3 = r_2$, where $r_1$ and $r_2$ are selected by the user.

The GLR statistic for a test of $H_0: c\beta = r$ is stated in Equation (4.3.10)

$$
\lambda(Y) = \left[ 1 + \frac{(c\hat{\beta} - r)' [c(x'x)^{-1} c']^{-1} (c\hat{\beta} - r)}{\hat{\varepsilon}'\hat{\varepsilon}} \right]^{-n/2}
$$

The statistic may be rewritten in terms of the F test statistic (4.3.11)

$$
F = \frac{(c\hat{\beta} - r)' [c(x'x)^{-1} c']^{-1} (c\hat{\beta} - r)}{jS^2}
$$

Under $H_0$, the test statistic is distributed as central $F(j, n - k)$ where $j$ is the number of rows in $c$ and $r$. Under the alternate hypothesis the test statistic is distributed as $F(j, n - k, \lambda)$, and

$$
\lambda = (c\beta - r)' \left[ \sigma^2 c(x'x)^{-1} c' \right]^{-1} (c\beta - r) / 2.
$$

The power function for the
level $\alpha$ test is

$$\pi(\lambda) = \int_{f_{1-\alpha}}^{\infty} F(z; j, n-k, \lambda) dz$$

where $f_{1-\alpha}$ is the critical value for the test.

Next, GAUSS presents a sequence of graphics windows that illustrate the relationship between $\pi(\lambda)$ and the probabilities of Type I and II errors under $F(j, n-k, \lambda)$. Finally, GAUSS approximates $\pi(\lambda)$ by Monte Carlo simulation, and we consider $H_0: \beta = 0$ with alternatives represented by $\beta_i \neq 0$. Under orthogonalized $\mathbf{x}$ such that $\mathbf{x}'\mathbf{x} = \mathbf{I}_k$, the noncentrality parameter reduces to $\lambda = \beta_i^2 / 2$. Recall that GAUSS uses the alternate form of the noncentrality parameter, $\sqrt{2\lambda} = \beta_i$.

**User Input:**

The user is asked to select the two elements of the vector $\mathbf{r}$ for a test of the linear restriction $H_0: \beta_2 = 2$ and $\beta_3 = 5$. Note that $H_0$ is true, and the elements of $\mathbf{r}$ are restricted to $-10 \leq r_1, r_2 \leq 10$. For the power simulation, the user is asked to specify the number of Monte Carlo experiments (1, 2, or 4) plus the sample sizes ($5 < n < 250$) and the number of Monte Carlo trials ($10 < m < 1000$) for each experiment.

**Program Output:**

GAUSS first generates $n = 40$ observations and reports the ML estimates of $\beta$ and the associated standard errors. Given the value of $\mathbf{r}$ selected by the user, GAUSS uses the
GLR_F() procedure to compute the observed F test statistic and the p-value for the test of $H_0$. Next, GAUSS plots a sequence of graphics windows that relate $\pi(\lambda)$ to the probabilities of Type I and II errors under the null and alternative distributions. In each window, the densities of $F(j, n - k, \lambda)$ are plotted with $\lambda = 0$ (i.e., under $H_0$) and $\lambda \neq 0$, and the areas under the PDFs are associated with the level of $\pi(\lambda)$.

Finally, after completing the Monte Carlo simulation exercise, GAUSS plots the estimated and actual power functions for the F-test as a function of the GAUSS form of the noncentrality parameter, $\sqrt{2\lambda} = \beta_1$. The user should find that the estimated power function approaches $\pi(\lambda)$ as the number of Monte Carlo trials ($m$) increases. Also, the simulated power should roughly correspond to $\pi(0) = 0.10$ under $H_0$ and $\pi(\lambda) \to 1$ as $\sqrt{2\lambda} = \beta_1 \to \infty$. 
EX. 4.3.2: Conduct and Power of the GLR T-Test of \( c\beta = r \) (C4GLR_T.gss)

**Program Background:**

This GAUSS program demonstrates the conduct of the two-sided T-test of the null hypothesis \( H_0: \beta_2 + \beta_3 = r \) (i.e., \( j = 1 \)). The program is based on outcomes from the normal linear regression model \( y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \epsilon_i \) where \( i = 1, \ldots, 40 \). The noise component is generated pseudorandom \( \epsilon \sim N(0,I) \), and the non-constant explanatory variables \( (x_2, x_3, \text{and} x_4) \) are generated as pseudorandom \( N(0,1) \) and held fixed in repeated samples. The first objective of the program is to conduct a two-sided test of \( H_0: \beta_2 + \beta_3 = r \).

The T statistic is stated in Equation (4.3.17) and takes the form

\[
T = \frac{c\hat{\beta} - r}{\left[ S^2 c (x'x)^{-1} c' \right]^{1/2}}
\]

Under the null hypothesis that the restriction is correct, the test statistic is distributed as Student \( T(n-k) \). Under the alternate hypothesis, the test statistic is distributed as \( T(n-k, \delta) \) where the noncentrality parameter is \( \delta = (c\beta - r) \left[ \sigma^2 c (x'x)^{-1} c' \right]^{1/2} \). The power function for the two-sided level \( \alpha \) test is

\[
\pi(\delta) = \int_{-t_{1-\alpha/2}}^{-t_{1-\alpha/2}} T(z; n-k, \delta) dz + \int_{t_{\alpha/2}}^{\infty} T(z; n-k, \delta) dz
\]
where $t_{1-\alpha/2}$ is the critical value for the test.

For the Monte Carlo simulation of the power function, we consider the special case $H_0 : \beta_1 = 0$. Under orthogonalized $x$ such that $xx' = I_k$, the noncentrality parameter reduces to $\delta = \beta_1$ in this case. GAUSS represents cases under the null and alternative hypotheses with $\beta_i \in \{-5, -4.75, ..., 5\}$.

**User Input:**

The user is asked to select the scalar, $-10 < r < 10$, for the hypothesis $H_0 : c\beta = r$. For the power simulation exercise, the user is asked to specify the number of Monte Carlo experiments (1, 2, or 4), the sample sizes ($5 < n < 250$), and the number of Monte Carlo trials ($10 < m < 1000$) for each experiment.

**Program Output:**

GAUSS first generates $n = 40$ observations and reports the ML estimates of $\beta$ and the associated standard errors and t-ratios. Given the value of $r$ selected by the user, GAUSS uses the GLR_T() procedure to compute the observed test statistic and the p-value for the two-sided T-test of $H_0 : c\beta = r$ at $\alpha = 0.10$. After completing the Monte Carlo simulation exercise, GAUSS plots the estimated and actual power functions for the T-test. The user should find that the estimated power function approaches $\pi(\delta)$ as the number of Monte Carlo trials ($m$) increases. For the two-sided test, $\pi(\delta)$ is approximately U-shaped such that $\pi(0) = 0.1$ and $\pi(\delta) \to 1$ as $|\delta| \to \infty$. 
EX. 4.3.3: Conduct and Power of the GLR T-Test
of $c\beta \leq r$ (C4T_INEQ.gss)

**Program Background:**

This program demonstrates the conduct of the T-test of the linear restriction $c\beta \leq r$ for the normal linear regression model $y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \epsilon_i$ where $i = 1, \ldots, 40$. The noise component is $\epsilon \sim N(0, I)$, and the explanatory variables ($x_2$, $x_3$, and $x_4$) are pseudorandom $N(0,1)$. The first objective of the program is to conduct a test of the null hypothesis $H_0 : \beta_2 + \beta_3 \leq r$ where $r$ is selected by the user.

The T test statistic of the linear restriction takes the form

$$T = \frac{c\beta - r}{S^2 c (x'x)^{-1} c'}^{1/2}$$

Under the null hypothesis that the restriction is correct, the test statistic is distributed as Student $T(n - k)$. Under the alternate hypothesis, the test statistic is distributed as $T(n - k, \delta)$ where the noncentrality parameter is $\delta = (c\beta - r)\left[\sigma^2 c (x'x)^{-1} c'\right]^{1/2}$. The power function for the one-sided level $\alpha$ test is stated in Equation (4.3.23)

$$\pi(\delta) = \int_{t_{1-\alpha}}^{\infty} T(z; n - k, \delta) \, dz$$

where $t_{1-\alpha}$ is the critical value for the test.

The second objective of the program is to approximate $\pi(\delta)$ by Monte Carlo
simulation. In this case, we consider a one-sided T test of null hypothesis \( H_0 : \beta_i \leq 0 \).

GAUSS uses the singular value decomposition (SVD) to orthogonalize \( x \) such that \( x'x = I_k \), and the noncentrality parameter for this test reduces to \( \delta = \beta_1 \).

**User Input:**

The user is first asked to select the upper bound on \( \beta_2 + \beta_3 \), \(-10 < r < 20\). For the power function simulation exercise, the user is asked to specify the number of Monte Carlo experiments (1, 2, or 4), the sample sizes (5 < n < 250), and the number of Monte Carlo trials (10 < m < 1000) for each experiment.

**Program Output:**

GAUSS generates \( n = 40 \) observations and reports the ML estimates of \( \beta \), the estimated standard errors, and the associated t-ratios. Given the value of \( r \) selected by the user, GAUSS uses the GLR_T() procedure to compute the observed test statistic and the p-value for the one-sided T-test of \( H_0 : \beta_2 + \beta_3 \leq r \). After completing the Monte Carlo simulation exercise, GAUSS plots the estimated and actual power functions for the T-test. The user should find that the estimated power function approaches \( \pi(\delta) \) as the number of Monte Carlo trials (m) increases. Due to the one-sided character of the 10% level test, the user should note that the simulated power function approximately reflects the properties

\[
\pi(0) = 0.1 \quad (\text{i.e., under } H_0), \quad \pi(\delta) \to 0 \quad \text{as} \quad \delta \to -\infty, \quad \text{and} \quad \pi(\delta) \to 1 \quad \text{as} \quad \delta \to \infty.
\]
EX. 4.3.4: Conduct and Size of the Bonferroni Joint Test of $c^\beta \leq r$ (C4Bjoint.gss)

**Program Background:**

This GAUSS program demonstrates the conduct of the Bonferroni joint test of multiple linear inequality restrictions $c^\beta \leq r$. We consider outcomes from the normal linear regression model $y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i$ where $i=1,\ldots,40$. The noise component is generated as pseudorandom $\varepsilon \sim N(\mathbf{0},\mathbf{I})$, and the non-constant explanatory variables $(x_2, x_3, \text{and } x_4)$ are pseudorandom Uniform(0,1). The first objective of the program is to test the null hypothesis $H_0 : \beta_2 \leq r_1$ and $\beta_3 \leq r_2$.

The $T$ test statistic of the individual linear inequality restriction ($i=1,2$) is based on Equation (4.3.17) and takes the form

$$T = \frac{c^\beta - r}{S^2c_i (x'x)^{-1}c'_i}^{1/2}$$

For $H_0 : \beta_2 \leq r_1$ and $\beta_3 \leq r_2$, note that $c_1 = [0 \ 1 \ 0 \ 0]$ and $c_2 = [0 \ 0 \ 1 \ 0]$. Under $H_0$, the test statistic is distributed as Student $T(n-k)$ and the $i^{th}$ component of $H_0$ is rejected if the observed test statistic $t$ exceeds the critical value $t_{1-\alpha}$. The critical value is the 100(1-$\alpha$)% quantile of the $T(n-k)$ distribution.

For the joint hypothesis test, the composite null hypothesis is rejected if one or more of the $m$ component hypotheses are rejected. If the probability of committing a
Type I error in the $i^{th}$ component test is $\alpha_i$, then Bonferroni’s inequality implies that the upper bound on the Type I error rate for the joint test is $\alpha \leq \sum_{i=1}^{m} \alpha_i$. Thus, the overall size of the joint test may be much larger than the intended size of the individual T-tests ($\alpha_i$). To control the overall Type I error rate of the joint test at level $\alpha$, we can set $\alpha_i = \alpha / m$ by the Bonferroni inequality. The critical values of the component tests in the Bonferroni joint test are then the $100(1 - \alpha/m)\%$ quantiles of the Student $T(n-k)$ distribution.

To demonstrate the impact of the Bonferroni adjustment, GAUSS conducts a Monte Carlo simulation of the Type I error rate for tests of $H_0$. For each Monte Carlo trial, GAUSS determines the test decision based on the unadjusted T-tests ($\alpha_i = 0.10$) and the Bonferroni adjusted T-tests ($\alpha_i = 0.10 / 2 = 0.05$). The share of false rejections for each procedure is an estimate of the Type I error rates.

**User Input:**

The user is asked to select the vector $\mathbf{r}$ for $H_0: c\mathbf{\beta} \leq \mathbf{r}$ where $-10 \leq r_1, r_2 \leq 10$. GAUSS sets the number of Monte Carlo trials at $m = 10,000$, and the user may repeat the exercise to view another set of simulation results.

**Program Output:**

GAUSS generates $n = 40$ observations and reports the ML estimates of $\mathbf{\beta}$, the estimated standard errors, and the associated t-ratios. Given the value of $\mathbf{r}$ selected by the user, GAUSS plots the joint rejection regions for the 10% (unadjusted) T-tests and the
Bonferroni joint test with overall size of 10% (each conducted at the 5% level). For ease of visual interpretation, the regions are expressed in terms of values of $r$ and $\beta$ rather than the T-test statistics. As such, the rejection regions may be viewed as the inverted one-sided confidence regions associated with the linear inequality hypotheses. The user should find that the Bonferroni rejection region is smaller than the joint region for the unadjusted tests, implying that the Type I error rate is smaller for the Bonferroni joint test. GAUSS also reports the individual test statistics, critical values, and test decisions for the components of the null hypothesis under the two test procedures.

After completing the Monte Carlo simulation exercise, GAUSS reports the estimated and intended (10%) rejection probabilities for the unadjusted and Bonferroni joint test procedures. Although the results will vary due to sampling variation in the Monte Carlo draws, the user should find that the size of the unadjusted test procedure is nearly double the intended Type I error rate. The size of the Bonferroni joint test should be slightly less than 10% due to the inaccuracy of the Bonferroni inequality.
EX. 4.3.5: Conduct and Size of the Bonferroni Joint Test of $c\beta = r$ (C4BTest.gss)

Program Background:

This GAUSS program demonstrates the conduct of the Bonferroni joint test of multiple linear equality restrictions $c\beta = r$. We consider outcomes generated from the normal linear regression model $y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \varepsilon_i$ where $i = 1, ..., 40$. The noise component is generated as pseudorandom $\varepsilon \sim N(0, I)$, and the non-constant explanatory variables ($x_2, x_3,$ and $x_4$) are generated as pseudorandom $N(0,1)$. The first objective of this program is to conduct a test of the joint null hypothesis $H_0 : \beta_2 = r_1$ and $\beta_3 = r_2$ where $r_1$ and $r_2$ are selected by the user.

Under the normality assumption, we can use an F test of $H_0: c\beta = r$, and the test statistic is stated in Equation (4.3.11)

$$F = \frac{(c\hat{\beta} - r)\left[c(X'X)^{-1}c\right]^{-1}(c\hat{\beta} - r)}{jS^2}$$

Under $H_0$, the test statistic is distributed as central $F(j, n - k)$ where $j$ is the number of rows in $c$ and $r$. Under the alternate hypothesis the test statistic is distributed as $F(j, n - k, \lambda)$ where the noncentrality parameter (see Equation (4.3.12)) is

$$\lambda = (c\beta - r)\left[\sigma^2 c(X'X)^{-1}c\right]^{-1}(c\beta - r)/2.$$ The power function for the level $\alpha$ test is stated in Equation (4.3.13)
\[ \pi(\lambda) = \int_{f_{1-\alpha}}^{\infty} F(z; j, n-k, \lambda) \, dz \]

where \( f_{1-\alpha} \) is the critical value for the test.

Based on the F-test results, we cannot determine which of the component restrictions is incorrect if we reject \( H_0 \). To diagnose the potential source of a rejection, we can use the T test statistics for the \( i^{th} \) individual linear inequality restrictions

\[ T = \frac{c_i \hat{\beta} - r_i}{\left[ S^2 c_i (\mathbf{x}' \mathbf{x})^{-1} c_i' \right]^{1/2}} \]

Under \( H_0 \), the test statistic is distributed as Student \( T(n-k) \). To conduct the joint hypothesis test, we reject the composite null hypothesis if one or more of the \( m \) component hypotheses are rejected. Following our discussion in Example 4.3.4, note that the overall probability of committing a Type I error based on component tests of level \( \alpha_i \) can be controlled with Bonferroni’s inequality, \( \alpha \leq \sum_{i=1}^{m} \alpha_i \). A test with overall Type I error rate of \( \alpha \) is conducted with individual tests conducted at level \( \alpha_i = \alpha/m \) by the Bonferroni inequality. The critical values of the component two-sided tests in the Bonferroni joint test are the \( 100(1-\alpha/2m)\% \) quantiles of the Student \( T(n-k) \) distribution.

**User Input:**

The user is asked to select the elements of the vector \( \mathbf{r} \) for the linear equality restrictions on elements \( \beta_2 \) and \( \beta_3 \), \(-10 \leq r_1, r_2 \leq 10\). GAUSS sets the number of simulation trials in the Monte Carlo experiment at \( m = 500 \) to limit computation time. The user may repeat
the exercise to view another set of simulation results.

**Program Output:**

GAUSS generates \( n = 40 \) observations and reports the ML estimates of \( \beta \), the estimated standard errors, and \( t \)-ratios. Given the value of \( r \) selected by the user, GAUSS reports the observed test statistics, critical values, and decisions for the 10% level F-test as well as the outcomes of the individual (adjusted and unadjusted) \( T \)-tests. The user should find that the critical values of the Bonferroni joint test are larger than the unadjusted \( T \) critical values, and the adjusted tests are less likely to reject the composite null hypothesis. Given that \( \beta \) is known, the user is asked to determine if the test decisions lead to Type I or Type II errors.

After completing the Monte Carlo simulation exercise, GAUSS plots the power function for the F-test, \( \pi(\lambda) \), as a function of the GAUSS form of the noncentrality parameter, \( \sqrt{2\lambda} \). GAUSS also plots the simulated power function for the Bonferroni joint test. The power functions are provided for samples of size \( n = 40, 60, 80, \) and 100 under the DSP described above. The user should find that the simulated power of the Bonferroni joint test is similar to \( \pi(\lambda) \) for each sample size \( n \).
EX. 4.3.6: Conduct and Power of the GLR Test of

\( H_0: \sigma^2 \leq \sigma_0^2 \) \hspace{1cm} (C4Chisq.gss)

**Program Background:**

This GAUSS program demonstrates the conduct of the GLR test of the inequality hypothesis \( H_0: \sigma^2 \leq \sigma_0^2 \). We consider the noise variance parameter \( \sigma^2 \) for the normal linear regression model \( y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \varepsilon_i \) where \( i = 1, \ldots, 40 \). The noise component is generated as pseudorandom \( \varepsilon \sim N(0,1) \) such that \( \sigma^2 = 1 \), and the explanatory variables \( (x_2, x_3, \text{ and } x_4) \) are also pseudorandom \( N(0,1) \) outcomes and held fixed in repeated samples.

The GLR procedure is based on the likelihood ratio stated in Equation (4.3.28)

\[
\lambda(Y) = \frac{\max_{\sigma^2 \leq \sigma_0^2, \beta} L(\beta, \sigma^2 | Y, x)}{\max_{\sigma, \beta} L(\beta, \sigma^2 | Y, x)}
\]

As discussed in section 4.3.7 in the text, \( \lambda(Y) \) may be manipulated to show that the null hypothesis should be rejected if

\[
\frac{(n-k)S^2}{\sigma_0^2} \geq \tau
\]

for some critical value \( \tau \). Under the assumption \( \sigma^2 = \sigma_0^2 \), the test statistic is distributed as central Chisquare\((n-k)\), and the critical value for the level \( \alpha \) test is \( \tau_{1-\alpha} = \chi^2_{n-k,1-\alpha} \). The power function for the level \( \alpha \) test is stated in Equation (4.3.30)
\[
\pi(\sigma^2) = \Pr\left[ \frac{(n-k)s^2}{\sigma_0^2} \geq \chi_{1-\alpha}^2 \mid \sigma^2 \right]
\]

As noted in the text, the test is unbiased and consistent.

**User Input:**

The user is asked to select the upper bound on the variance parameter, \(\sigma_0^2\), for the hypothesis test. For the power simulation, the user is asked to specify the number of Monte Carlo experiments (1, 2, or 4) and the sample sizes (5 < \(n\) < 250) and the number of Monte Carlo trials (10 < \(m\) < 1000) for each experiment.

**Program Output:**

GAUSS generates a set of 40 observations for the stated regression model and reports the parameter estimates, estimated standard errors, and associated t-statistics for the ML estimator of \(\beta\). GAUSS also reports the ML estimate of \(\sigma^2\). Given the value of \(\sigma_0^2\) selected by the user, GAUSS computes the transformed GLR test statistic and the p-value under the Chisquare(n – k) null distribution. After completing the Monte Carlo simulation exercise, GAUSS plots the estimated and actual power functions for the GLR test as a function of variance ratio, \(\sigma^2/\sigma_0^2\). The user should find that the estimated power function approaches the actual power function as the number of Monte Carlo trials (\(m\)) increases.

Also, the test is consistent, and the user should find that \(\pi(\sigma^2) \to 0\) for \(\sigma^2/\sigma_0^2 < 1\) and \(\pi(\sigma^2) \to 1\) for \(\sigma^2/\sigma_0^2 > 1\) as \(n \to \infty\).
**EX. 4.4.1: Formation of Confidence Regions under the GLR (F) Criterion (C4Region.gss)**

*Program Background:*

This program demonstrates the formation of 100(1-\(\alpha\))% confidence regions for the linear combination \(c^\beta\) under the normal linear regression model \(y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \varepsilon_i\) where \(i = 1, \ldots, 40\). The noise component is generated as pseudorandom \(\varepsilon \sim N(0,I)\), and the non-constant explanatory variables \((x_2, x_3, \text{ and } x_4)\) are pseudorandom \(N(0,1)\). The program also compares the F-based confidence with the region formed as the intersection of the individual confidence intervals based on the T criterion.

Under the GLR (F) criterion, the 100(1-\(\alpha\))% confidence region stated in Equation (4.4.1) is

\[
\text{CR}^{\text{GLR}} = \bigg\{ \mathbf{r} : \left(\mathbf{c}^\hat{\beta} - \mathbf{r}\right) \left[S^2 \mathbf{e} (\mathbf{x}'\mathbf{x})^{-1} \mathbf{c}'\right]^{-1} \left(\mathbf{c}^\hat{\beta} - \mathbf{r}\right) < f_{j-\alpha} \bigg\}
\]

The bound for the hyper-elliptical confidence region is the 100(1-\(\alpha\))% quantile of the central \(F(j, n-k)\) distribution where \(j\) is the number of rows in \(\mathbf{c}\) and \(\mathbf{r}\). The GAUSS procedure FRegion() provided in the EF library may be used to compute the boundary of the ellipse in the special case \(j = 2\). In the case that \(j = 1\) and the restriction is scalar-valued, the confidence interval may be stated as in Equation (4.4.2)

\[
\text{CR}^{\text{GLR}} = \left(\mathbf{c}^\hat{\beta} - t_{1-\alpha/2} \left(S^2 \mathbf{c} (\mathbf{x}'\mathbf{x})^{-1} \mathbf{c}'\right)^{1/2}, \mathbf{c}^\hat{\beta} + t_{1-\alpha/2} \left(S^2 \mathbf{c} (\mathbf{x}'\mathbf{x})^{-1} \mathbf{c}'\right)^{1/2}\right)
\]
where $t_{1-\alpha/2}$ is the $100(1-\alpha/2)\%$ quantile of the Student $T(n-k)$ distribution. In the non-scalar case, we may use the Bonferroni inequality to form confidence regions for the $i^{th}$ element of $c\beta$ as

$$CR^{GLR} = \left( c\hat{\beta} - t_{1-\alpha/2j} \left( S^2 c_i (x'x)^{-1} c'_i \right)^{1/2} , c\hat{\beta} + t_{1-\alpha/2j} \left( S^2 c_i (x'x)^{-1} c'_i \right)^{1/2} \right)$$

Thus, each component of the joint confidence region will have a coverage probability equal to $\alpha_i = \alpha/j$. The coverage probability for the Bonferroni confidence region will not be exact due to the inaccuracy of the Bonferroni inequality, and the coverage probability is bounded from below at $1 - \sum_{i=1}^{j} \alpha_i$. Hence, the actual coverage probability should be slightly larger than $100(1-\alpha)\%$.

**User Input:**

After viewing the simulation results, the user may repeat the Monte Carlo exercise to generate another set of estimated coverage probabilities.

**Program Output:**

GAUSS generates $n = 40$ observations and reports the ML estimates of $\beta$ and the 90% confidence intervals for each parameter. Note that the 90% intervals may be formed using the third expression above with $j = 1$ and $c_i$ equal to the $(1 \times k)$ row vector of zeros with one in the $i^{th}$ column. GAUSS then plots the joint confidence region for $\beta_2$ and $\beta_3$ based on the GLR (F) and Bonferroni (T) criteria. The user should note that $CR^{GLR}$ is an ellipse
centered on the estimates of $\beta_2$ and $\beta_3$, and the Bonferroni region is rectangular. Further, the two regions will include subsets of the parameter space that are not included in the other region.

Next, GAUSS simulates the coverage probabilities of the two procedures. For each of $m = 10,000$ Monte Carlo trials, GAUSS computes the 90% confidence regions and determines if the regions include the true values of $\beta_2$ and $\beta_3$. The estimated coverage probability for a particular procedure is the share of the $m$ trials in which the region covers the true parameter values. After completing the Monte Carlo simulation exercise, GAUSS reports the estimated coverage probabilities, and the user should find that both procedures provide coverage near 90%. The GLR procedure should provide slightly more accurate coverage because it is based on an exact finite sample distribution.
EX. 4.4.2: Formation of Joint One-sided Confidence Bounds (C4Bounds.gss)

**Program Background:**
This program demonstrates the formation of joint one-sided confidence bounds for the linear combination $c\beta$ based on the Bonferroni inequality. For the normal linear regression model $y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \epsilon_i$ where $i = 1, \ldots, 40$, the noise component is generated as pseudorandom $\epsilon \sim \mathcal{N}(0,1)$. Two of the non-constant explanatory variables ($x_2$ and $x_4$) are pseudorandom $\mathcal{N}(0,1)$, and $x_3 = -0.6x_2$ plus $\mathcal{N}(0,1)$ pseudorandom noise to enhance the demonstration.

If $c\beta$ is scalar-valued, we can use the T-criterion to form a lower bounded interval with $100(1-\alpha)\%$ coverage as in Equation (4.4.3)

$$CR^T = \left( c\hat{\beta} - t_{1-\alpha} \left[ S^2 c(x'x)^{-1} c' \right]^{1/2}, \infty \right)$$

The associated upper bounded interval is stated in Equation (4.4.4)

$$CR^T = \left( -\infty, c\hat{\beta} + t_{1-\alpha} \left[ S^2 c(x'x)^{-1} c' \right]^{1/2} \right)$$

If the row dimension of $c$ is greater than one, we can form a confidence region as the intersection of the component intervals. However, the coverage probability of the resulting region will exceed the intended coverage probability. To control the overall coverage probability at approximately $100(1-\alpha)\%$, we can select the $100(1-\alpha/j)\%$
quantiles of the Student $T(n-k)$ distribution by the Bonferroni inequality. The $i^{th}$ lower bounded interval is

$$\text{CR}_i^T = \left( c_i \hat{\beta} - t_{1-\alpha/j} \left[ S^2 c_i (x'x)^{-1} c_i' \right]^{1/2}, \infty \right)$$

and the $i^{th}$ upper bounded interval is

$$\text{CR}_i^T = \left( -\infty, c_i \hat{\beta} + t_{1-\alpha/j} \left[ S^2 c_i (x'x)^{-1} c_i' \right]^{1/2} \right)$$

For each component interval, $\alpha_i = \alpha/j$ and the Bonferroni inequality implies that the overall coverage probability may be bounded from below at $1 - \sum_{i=1}^{j} \alpha_i$.

**User Input:**

The user may repeat the exercise to view another set of simulation results.

**Program Output:**

GAUSS $n = 40$ observations and reports the ML estimates of $\beta$ and the individual two-sided 95% confidence intervals. GAUSS then forms the upper and lower bounds for the 90% confidence regions based on the Bonferroni criterion. The upper and lower bounded regions and the ML estimates of $\beta_2$ and $\beta_3$ are plotted by GAUSS.

GAUSS then approximates the coverage probabilities of the one-sided regions by Monte Carlo simulation. For each of $m = 10,000$ trials, GAUSS computes the upper and lower bounds under the Bonferroni criterion and determines if the joint region covers the
true values of $\beta_2$ and $\beta_3$. The share of trials in which the region covers $\beta_2$ and $\beta_3$ is the estimate of the coverage probability. GAUSS reports the estimated coverage probabilities for the upper and lower bounded confidence regions. Although the results will vary due to sampling variation in the Monte Carlo draws, the user should find that the estimated coverage probabilities are often slightly larger than 90% due to the inaccuracy of the Bonferroni inequality.
EX. 4.4.3: Formation of Confidence Intervals for $S^2$

(C4S2CR.gss)

Program Background:

This GAUSS program demonstrates the formation of upper and lower bounded confidence intervals for the unbiased variance estimator, $S^2$. For the normal linear regression model $y_i = 10 + 2x_{i2} + 5x_{i3} - 3x_{i4} + \varepsilon_i$ where $i = 1, \ldots, 40$, the noise component is generated as pseudorandom $\varepsilon \sim N(0,1)$ and the explanatory variables ($x_2$, $x_3$, and $x_4$) are pseudorandom $N(0,1)$. As discussed in section 4.4.4 in the text, the known Chisquare distribution result for the variance estimator may be used to form the lower bounded confidence interval as in Equation (4.4.6)

$$CR = \left( \frac{(n-k)S^2}{\chi^2_{\alpha}}, \infty \right)$$

where $\chi^2_{\alpha}$ is the 100\% quantile of the central Chisquare(n-k) distribution. An upper bounded confidence interval may be formed as stated in Equation (4.4.7)

$$CR = \left[ 0, \frac{(n-k)S^2}{\chi^2_{1-\alpha}} \right]$$

Note that the upper bounded interval is the complement of the rejection region for the level-$\alpha$ test stated in Equation (4.3.29).
**User Input:**

The user is asked to specify the number of Monte Carlo experiments (1, 2, ..., 8), the sample sizes (5 < n < 250), and the number of trials (50 < m < 5000) for each experiment.

**Program Output:**

GAUSS generates n = 40 observations and reports the ML estimates of $\beta$, the estimated standard errors, and the associated t-ratios. GAUSS also reports the estimate of $\sigma^2$ based on the unbiased estimator $S^2$. Then, GAUSS computes the upper and lower bounded 90% confidence intervals for $S^2$ and reports the bounds to the user. After completing the Monte Carlo simulation exercise, GAUSS reports the estimated and intended coverage probability for the set of experiments requested by the user. The user should find that the estimated coverage probability approaches 90% as the number of Monte Carlo trials (m) increases. Further, the coverage accuracy should not depend on the sample size (n) because the upper and lower bounded intervals are based on exact distribution results rather than a large sample approximation.
Program Background:

The purpose of this GAUSS program is to replicate the quadratic risk plots provided in Figure 4.1 by Monte Carlo simulation. We consider the problem of imposing the linear restriction \( \beta = 0 \) on the normal linear regression model \( y = x\beta + \epsilon \) for \( n = 40 \). The noise component is \( \epsilon \sim N(0, I) \), and the non-constant explanatory variables \( (x_2, x_3, \text{and } x_4) \) are pseudorandom \( N(0, 1) \). To simplify the risk calculations, we also use the singular value decomposition (SVD) to orthogonalize \( x \) such that \( x'x = I \).

In this case, the unrestricted estimator may be written as \( \hat{\beta} = \beta + x'\epsilon \), and the quadratic risk of \( \hat{\beta} \) is simply

\[
\rho(\beta, \hat{\beta}) = E\left[ (\hat{\beta} - \beta)' (\hat{\beta} - \beta) \right] = k
\]

The restricted ML (RML) estimator takes the value \( r = 0 \) for all sample outcomes, and the hypothesis error for the restricted estimator is \( \delta = \beta \). Thus, the quadratic risk of the RML estimator is simply \( \beta'\beta \). When plotted as functions of \( \beta'\beta \), note that the risk function for \( \hat{\beta} \) is constant (see Figure 4.1) and the RML risk function is linear with unitary slope (i.e., the 45-degree line).

The preliminary test (pretest) estimator is based on an initial hypothesis test of \( H_0 : \beta = 0 \). If we reject \( H_0 \) under the F-test criterion (4.3.11), the estimate of \( \beta \) is based
on the outcome of \( \hat{\beta} \). Otherwise, the estimate of \( \beta \) is taken to be \( r = 0 \). Following Equation (4.5.4), the pretest estimator may be written as a linear combination of the ML and RML estimators with weights determined by the outcome of the F statistic. In this case, the pretest estimator may be stated as

\[
\hat{\beta} = I_{(\tau,\alpha)}(F)\hat{\beta}
\]

where \( \tau \) is the level \( \alpha \) critical value for the F-test of \( \beta = 0 \). As \( \alpha \) increases such that we are more likely to reject \( H_0 \), \( \tau \) becomes smaller and the pretest estimator behaves more like the ML estimator. The quadratic risk function for \( \hat{\beta} \) is stated in Equation (4.5.6)

\[
\rho\left(\beta, \hat{\beta}\right) = \sigma^2 k - E \left[I_{(0,1)}(F)\left(\hat{\beta} - \beta\right)\left(\hat{\beta} - \beta\right)^\prime\right] + \delta \delta^\prime P[f < \tau]
\]

The risk expression has been analytically evaluated by Judge and Bock (1978) but is very complicated. In this program, GAUSS uses Monte Carlo experiments to simulate the risk function.

**User Input:**

The user is asked to specify the number of Monte Carlo trials (50 < \( m \) < 5000) used in each experiment. The user may repeat the demonstration to review the results as many times as desired.

**Program Output:**

GAUSS plots the estimated quadratic risk functions for the ML, RML, and pretest
estimators. In this example, $\delta = \beta$ and the horizontal axis of the risk plots is enumerated in units of $\delta' \delta = \beta' \beta$ (following Figure 4.1). The user should find that the simulated $\hat{\rho}(\beta,\hat{\beta})$ is roughly constant (subject to sampling error) at $k = 4$ for all values of $\delta$ (recall that $\sigma^2 = 1$). Accordingly, the quadratic risk of the restricted ML estimator is linear as a function of $\beta' \beta$. In general, $\hat{\rho}(\beta,\hat{\beta})$ is similar to $\rho(\beta,\hat{\rho})$ for small $\beta' \beta$ because the F-test commits Type II errors with high probability. As $\beta' \beta$ increases, $\hat{\rho}(\beta,\hat{\beta}) \rightarrow \rho(\beta,\hat{\beta})$ because the Type II error rate declines and the F-test rejects $H_0$ with much higher probability. The user should find that $\hat{\rho}(\beta,\hat{\beta})$ is more similar to $\rho(\beta,\hat{\rho})$ for small values of $\alpha$ because the F-test is less likely to reject $H_0$. Accordingly, $\hat{\rho}(\beta,\hat{\beta})$ follows $\rho(\beta,\hat{\beta})$ more closely as $\alpha$ increases.
EX. 5.2.1: Building a Least Squares Procedure in GAUSS (C5Proc.gss)

Program Background:

This program outlines the steps required to build a GAUSS procedure that computes the least squares or maximum likelihood estimates \( \hat{\beta} = (x'x)^{-1} x'Y \) based on Equations (3.2.9) or (5.2.10). The easyols() procedure is a simplified version of the ols() procedure (native to GAUSS Light) and returns information that will be used in other estimation and inferential programs in the early chapters of the book. The procedure is included in the EF library.

User Input:

The interactive GAUSS program steps through the construction of the procedure but does not ask the user for other types of input. The input to the easyols() procedure includes the (n x 1) vector of observed dependent responses (y), the (n x k) matrix of explanatory variables (x), and a binary (0-1) variable (I) that indicates GAUSS should add an intercept term to x. The user should set the binary variable as

\[
I = \begin{cases} 
1 & \text{include an intercept} \\
0 & \text{exclude an intercept} 
\end{cases}
\]

The user may set I = 0 if x already includes a column vector of ones or if the model should not include a constant term. As a preliminary check on the data, the procedure does verify
that \( y \) and \( x \) are conformable and that \( I \) is binary. A call to the easyols() procedure takes the general form easyols(\( y, I, x \)), and

**Program Output:**

The interactive program does not include a demonstration problem because the procedure is used in other GAUSS programs provided for Chapter 5. Thus, the only output from this program is the sequence of steps involved in constructing easyols(). The output from the easyols() procedure includes the least squares parameter estimates and the estimated variance matrix for the estimator

\[
\hat{\beta} = (x'x)^{-1} x'y \quad \text{and} \quad \text{cov}(\hat{\beta}) = s^2 (x'x)^{-1}
\]

The estimate of \( \beta \) is stated in Equation (5.2.9), and the estimated covariance matrix is stated in Equation (5.4.6). Recall that \( s^2 \) is the outcome of the unbiased estimator of \( \sigma^2 \) from Equation (5.4.5).
EX. 5.3.2: Consistency of the Least Squares Estimator (C5Plim.gss)

Program Background:

This GAUSS program uses a bootstrap resampling exercise to simulate the bias and variance of \( \hat{\beta} \) for a simple linear regression model

\[
y_i = \beta_1 + \beta_2 x_i + \varepsilon_i
\]

where \( i = 1, \ldots, n \). The explanatory variable \( x \) is generated as pseudorandom \( N(10,1) \) and held fixed in repeated samples for the simulation exercise. The noise vector \( \varepsilon \) is generated from a probability model selected by the user (see below). For details on the residual bootstrap procedure, the user should refer to the documentation of C5Boot.gss and Appendix section A.3.

User Input:

The user is first asked to select the number of simulation experiments (1, 2, \ldots, 8), the samples sizes for each (10 < n < 1000) experiment, and the number of trials in each experiment (10 < m < 1000). Then, GAUSS asks the user to select values of the location parameters (-10 < \( \beta_1, \beta_2 < 10 \)). Finally, the user selects one of three probability models for the noise component, \( \varepsilon \sim N(0,1), \varepsilon_i \sim \text{Uniform}(\sqrt{3} , \sqrt{3}), \) or \( \varepsilon_i \sim \left( \chi^2(3) - 3 \right)/\sqrt{6} \) (i.e., centered and scaled ChiSquare(3) distribution). Note that each of the candidate probability
models has $E[\varepsilon_i] = 0$ and $\text{var}(\varepsilon_i) = 1$. After GAUSS reports the simulation results, the user can repeat the experiment as often as desired.

**Program Output:**

GAUSS reports the simulated bias and variance of the replicated least squares intercept ($\beta_1$) and slope ($\beta_2$) estimates for each sample size (n) requested by the user. For the intercept parameter $\beta_1$, the simulated moments reported by GAUSS are computed as

$$\hat{E}(\hat{\beta}_1) = m^{-1} \sum_{j=1}^{m} \hat{b}_{(j)}^* \quad \text{and} \quad \text{var}(\hat{\beta}_1) = (m-1)^{-1} \sum_{j=1}^{m} (\hat{b}_{(j)}^* - \hat{E}(\hat{\beta}_1))^2$$

The user should find that the bias and variance approach zero as the sample size (n) increases, which reflects the mean-square convergence property $\hat{\beta} \overset{m}{\rightarrow} \beta$. GAUSS then presents box and whisker plots for the set of bootstrap distributions for the intercept and slope estimators. The boxes represent the interquartile range of the bootstrap outcomes, and the whiskers indicate the 5% and 95% quantiles of the bootstrap outcomes. The user should find that the boxes and whiskers collapse on the true parameter values (selected by the user) as n increases due to the consistency of $\hat{\beta}$. 
EX. 5.3.3: Asymptotic Normality of the Least
Squares Estimator (C5Plim.gss)

**Program Background:**

This GAUSS program uses a bootstrap resampling exercise to simulate the sampling
distribution of $\hat{\beta}$ for a simple linear regression model

$$y_i = \beta_1 + \beta_2 x_i + \epsilon_i$$

where $i = 1, \ldots, n$. The explanatory variable $x$ is generated as pseudorandom $\mathcal{N}(10,1)$ and
held fixed in repeated samples for the simulation exercise. The noise vector $\epsilon$ is generated
from a probability model selected by the user (see below). For details on the residual
bootstrap procedure, the user should refer to the documentation of C5Boot.gss and
Appendix section A.3.

**User Input:**

The user is first asked to select the number of simulation experiments (1, 2, 3, or 4), the
samples sizes for each $(10 < n < 1000)$ experiment, and the number of bootstrap samples
generated in each experiment $(20 < m < 2000)$. Then, GAUSS asks the user to select
values of the location parameters $(-10 < \beta_1, \beta_2 < 10)$. Finally, the user selects one of three
probability models for the noise component, $\epsilon \sim \mathcal{N}(0,1), \epsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})$, or
$\epsilon_i \sim \left(\chi^2(3) - 3\right)/\sqrt{6}$ (i.e., centered and scaled Chisquare(3) distribution). Note that each
of the candidate probability models has \[ E[\varepsilon_i] = 0 \] and \[ \text{var}(\varepsilon_i) = 1. \] After GAUSS reports the simulation results, the user can repeat the experiment as often as desired.

**Program Output:**

GAUSS first plots histograms of the replicated bootstrap outcomes for each sample size. The histograms are estimates of the density functions of \( \hat{\beta}_1 \) and \( \hat{\beta}_2 \) (refer to Equation (21.1.2) and the associated discussion in section 21.1). The yellow lines in each histogram represent the LS estimates from the initial sample (\( \hat{b}_1 \) or \( \hat{b}_2 \)), which are viewed as the true values of \( \beta_1 \) and \( \beta_2 \) in the bootstrap simulation exercise. The user should find that the histograms roughly exhibit normal character and are symmetric about the yellow lines, especially as the number of bootstrap samples \( m \) and the sample size \( n \) increase.

Next, GAUSS forms the empirical distribution function (EDF) of the replicated estimates of \( \beta_1 \) and \( \beta_2 \). For each sample size \( n \), the EDF is plotted with the associated asymptotic normal distribution, \[ \mathcal{N}\left(\hat{\beta}_i, s^2 \left[ (X'X)^{-1} \right]_{ii} \right) \text{ for } i=1,2. \] Again, note that the estimates of \( b \) and \( s^2 \) from the initial sample are treated as the true values of the parameters in the bootstrap world. As in the histograms, the LS estimates of \( \beta_1 \) and \( \beta_2 \) are indicated with yellow lines, and the true intercept and slope coefficients (selected by the user) are marked with red lines. As \( m \) and \( n \) increase, the user should find that the EDF approaches the CDF due to the asymptotic normality of \( \hat{\beta} \).
EX. 5.4.1: Unbiasedness and Consistency of $S^2$

(C5S2.gss)

**Program Background:**

This GAUSS program uses a bootstrap resampling exercise to simulate the bias and variance of the unbiased variance estimator, $S^2$. For purposes of the simulation, we consider a simple linear regression model

$$y_i = \beta_1 + \beta_2 x_i + \epsilon_i$$

where $i = 1, \ldots, n$. The explanatory variable $x$ is generated as pseudorandom $N(10,1)$ and held fixed in repeated samples for the simulation exercise. The noise vector $\epsilon$ is pseudorandom $N\left(0, \sigma^2 I_n\right)$, and the user selects the value of the variance parameter $\sigma^2$ (see below). For details on the residual bootstrap procedure, the user should refer to the documentation of Ex. 5.3.1 (C5Boot.gss) and Appendix section A.3.

**User Input:**

At the beginning of the program, the user has the option to review the steps involved in constructing the GAUSS procedure s2() (provided in the EF library). The call to the procedure is s2(y, I, x) where y and x are the observed data. The binary (0-1) variable I indicates to GAUSS if a constant term should be added to x. The user should set $I = 1$ if the column of ones should be appended to x, and $I = 0$ otherwise (see the documentation
for C5Proc.gss for more details). GAUSS also verifies that \( y \) and \( x \) are conformable before computing and reporting the sample outcome of \( S^2 \) based on Equation (5.4.5).

Next, the user selects the location parameters \((-10 < \beta_1, \beta_2 < 10\) and the variance parameter \((0 < \sigma^2 < 5\) for the simple linear regression model. Finally, the user selected the number of bootstrap experiments \((1, 2, \ldots, 8)\), the sample size \((10 < n < 1000)\), and the number of bootstrap samples \((10 < m < 1000)\) drawn for each experiment. After GAUSS reports the results of the simulation exercise, the user can opt to repeat the experiment.

**Program Output:**

After completing the bootstrap resampling exercise, GAUSS reports the sample bias and variance of the replicated variance estimates \( S^2^* \) for each sample size \( n \) requested by the user. The simulated moments reported by GAUSS are computed as

\[
\hat{\text{bias}}(S^2) = m^{-1} \sum_{j=1}^{m} s_j^2 - s^2 \quad \text{and} \quad \hat{\text{var}}(S^2) = (m-1)^{-1} \sum_{j=1}^{m} (s_j^2 - \hat{E}(S^2))^2
\]

The user should find that the bias and variance of the estimator approach zero as \( n \) becomes large due to the consistency of \( S^2 \). Then, GAUSS presents a box and whiskers plot of the bootstrap distributions for each sample size selected by the user. The red line in the plot indicates the true value of \( \sigma^2 \) (selected by the user). The boxes represent the interquartile range of the replicated outcomes, and the whiskers indicate the 5% and 95% quantiles of the bootstrap distribution. The user should find that the boxes and whiskers collapse on \( \sigma^2 \) as \( n \) and \( m \) become large due to the consistency of \( S^2 \).
Ex. 6.2.1: Conduct and Power of the Wald Test of $c \beta = r$ (C6Wald.gss)

Program Background:
This program demonstrates the conduct of the Wald test of the linear restriction $c \beta = r$ for the semiparametric linear regression model $y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i$. The sample size is $n = 40$, the noise component is generated under a probability model selected by the user, and the explanatory variables ($x_2$, $x_3$, and $x_4$) are pseudorandom $N(0,1)$ and held fixed in repeated samples. The first objective of the program is to conduct a test of $H_0 : \beta_2 = r_1$ and $\beta_3 = r_2$, where $r_1$ and $r_2$ are selected by the user. The second objective of the program is to simulate the power function of the Wald test of the joint null hypothesis $H_0 : \beta = 0$. The bootstrap methods discussed in section A.3 are used for the simulation exercise, and the results are compared to the asymptotic power function based on the limiting Chi-square distribution.

The operational version of the Wald statistic for a test of the joint null hypothesis $H_0 : c \beta = r$ is stated in Equation (6.2.2)

$$W = (c \hat{\beta} - r) \left[ S^2 c (x'x)^{-1} c' \right]^{-1} (c \hat{\beta} - r)$$

Under $H_0$, the test statistic is asymptotically distributed as central Chi-square($j$) where $j$ is the number of rows in $c$ and $r$ ($\text{rank}(c) = j \leq k$). Under the alternate hypothesis, $W$ is
asymptotically Chi-square($j, \lambda$) with $\lambda = \delta \left[ \sigma^2 c \Xi^{-1} c^T \right]^{-1} \delta / 2$. Recall that we assume

$(n^{-1} x' x) \to \Xi$ and that the hypothesis error under Pitman drift is $n^{1/2} \delta$ (see Section 6.2.1.a). The power function for the level $\alpha$ test is stated in Equation (6.2.7)

$$\pi(\lambda) = \int_{\tau}^{\infty} \text{Chi-square}(z; j, \lambda) \, dz$$

where $\tau$ is the critical value for the $\alpha$-level Wald test from Equation (6.2.6)

$$\alpha = \int_{\tau}^{\infty} \text{Chi-square}(z; j, 0) \, dz$$

**User Input:**

First, the user selects one of three probability models to represent the noise components, $\varepsilon \sim \mathcal{N}(0, I), \varepsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})$, or $\varepsilon_i \sim \left( \chi^2(3) - 3 \right) / \sqrt{6}$ (i.e., centered and scaled Chi-square(3) outcomes). Note that each of the candidate probability models has $E[\varepsilon_i] = 0$ and $\text{var}(\varepsilon_i) = 1$. Then, the user is asked to select the two elements of the vector $r$ for a test of the linear restriction $02 3H : \beta_2 = 2$ and $\beta_3 = 5$, and the elements of $r$ are restricted to $-10 \leq r_i, r_2 \leq 10$. For the power simulation exercise, the user is asked to specify the number of bootstrap experiments (1, 2, 3, or 4) plus the associated sample sizes ($5 < n < 250$) and the number of bootstrap trials ($10 < m < 10,000$) for each simulation experiment.
**Program Output:**

GAUSS first generates $n = 40$ observations and reports the LS estimates of $\beta$ and the associated standard errors. Given the value of $r$ selected by the user, GAUSS uses the MLR_Wald() procedure from the EF library to compute the observed W test statistic and the asymptotic p-value for the test of $H_0$. GAUSS briefly explains the use of procedure, and the standard call to the procedure takes the form

$$\{w, pvalue\} = \text{MLR}_\text{Wald}(c, \text{bhat}, r, \text{cov})$$

where ‘bhat’ is the vector of LS estimates $b$ and ‘cov’ is the estimated covariance matrix, $\text{cov} (\hat{b}) = s^2 (X'X)^{-1}$.

After completing the bootstrap simulation exercise, GAUSS plots the estimated and asymptotic power functions for the Wald test as a function of the GAUSS form of the noncentrality parameter, $\sqrt{2\lambda}$. The user should find that the estimated power function approaches the asymptotic power function $\pi(\lambda)$ as the number of bootstrap trials ($m$) and the sample size ($n$) increase. Also, the simulated power should roughly correspond to the asymptotic power under $H_0$, $\pi(0) = 0.10$, and $\pi(\lambda) \to 1$ as $\sqrt{2\lambda} \to \infty$. 
Ex. 6.2.2: Conduct of the LM Test of H0 : $c\beta = r$

(C6LM.gss)

Program Background:

The first purpose of this program is to describe the construction of the GAUSS procedure MLR_LM(), which may be used to conduct a Lagrange Multiplier (LM) test of the joint null hypothesis $H_0 : c\beta = r$. The LM test for the semiparametric linear regression model is developed in section 6.2.2 in the text, and we focus on the operational version of the score form of the LM statistic stated in Equation (6.2.29)

$$LM = \left( c\hat{\beta} - r \right) \left[ S_r^2 c (x'x)^{-1} c' \right]^{-1} \left( c\hat{\beta} - r \right)$$

Recall that $\hat{\beta}$ is the unrestricted LS estimator of $\beta$, and $S_r^2$ is the restricted estimator of $\sigma^2$ from Equation (6.2.16)

$$S_r^2 = \frac{(Y - x\hat{\beta}_r)' (Y - x\hat{\beta}_r)}{n - k + j}$$

Under $H_0$, LM is asymptotically Chi-square($j$) where $j$ is the number of rows in $c$ and $r$ (i.e., $\text{rank}(c) = j \leq k$). Under $H_a$ with hypothesis error $n^{-1/2} \delta$ (i.e., Pitman drift form), LM is asymptotically Chi-square($j, \lambda$) with noncentrality parameter $\lambda = \delta' \left( c\Xi^{-1} c' \right)^{-1} \delta / 2$ and $(n^{-1} x'x) \rightarrow \Xi$ by assumption.

The second purpose of this program is to compare outcomes of the score form of
the LM test statistic with the associated Wald (W) test outcomes. As noted in section 6.2.4, the numerators of the test statistics, \( (c\hat{\beta} - r)^{\prime} \left[ c(x'x)^{-1} c' \right]^{-1} (c\hat{\beta} - r) \), are identical, but the numerators (\( S_r^2 \) and \( S^2 \)) differ. Although \( W \geq LM \) iff \( S_r^2 \geq S^2 \), the outcomes of the estimators of \( \sigma^2 \) cannot be unconditionally ordered because each employs a different denominator (\( n - k + j \) versus \( n - k \)) to achieve unbiasedness. If we replace the denominators of \( S_r^2 \) and \( S^2 \) with \( n \) so that the estimators are biased but consistent, then \( W \geq LM \) in all cases because the restricted LS estimator \( \hat{\beta}_r \) necessarily satisfies

\[
(Y - x\hat{\beta}_r)^{\prime} (Y - x\hat{\beta}_r) \geq (Y - x\hat{\beta})^{\prime} (Y - x\hat{\beta}).
\]

To demonstrate the comparison of \( W \) and LM outcomes based on the divide-by-n (biased) estimators of \( \sigma^2 \), we consider \( n = 40 \) pseudorandom outcomes generated for the linear regression model \( y = x\beta + \varepsilon \). The \( (k - 1) = 3 \) non-constant explanatory variables are generated pseudorandom \( N(0,1) \) and are treated as fixed in repeated samples. The noise outcomes are pseudorandom \( N(0, 0.25) \), and the location parameter vector is specified as \( \beta = (0 \ 0 \ 0 \ \beta_4)^{\prime} \). For a test of \( H_0 : \beta = \theta \), note that the noncentrality parameter \( \lambda \) is proportional to \( \beta_4 \). Thus, to compare the test outcomes for different values of \( \beta_4 \), GAUSS computes the observed outcomes of the \( W \) and LM statistics and plots the results along with the 10% critical value from the limiting null Chi-square(4) distribution.

**User Input:**

The user does not provide any input to this program but can repeat the demonstration as
many times as desired.

**Program Output:**

First, GAUSS outlines the steps involved in constructing the MLR_LM() procedure. The call to the procedure takes the form

```
{lm, pvalue} = MLR_LM(c, bhat, r, cov)
```

where ‘bhat’ is the unrestricted LS estimates of $\beta$ and ‘cov’ is the estimated covariance matrix based on the restricted LS estimator of $\sigma^2$, $s^2_r (x'x)^{-1}$. Next, GAUSS generates the observations as described above and plots the test results for $\beta_4 \in [0, 0.4]$. Given that we use the divide-by-n estimators of $\sigma^2$, the user should find that the outcomes of the Wald test statistic are weakly larger than the LM test outcomes and that the difference increases as $\beta_4$ becomes larger.
Ex. 6.3.1: Formation of Confidence Regions under the Wald Criterion (C6CRWald.gss)

Program Background:

This program demonstrates the formation of 100(1-\(\alpha\))% confidence regions for the linear combination \(c\beta\) under the semiparametric linear regression model

\[ y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i \]

The sample size is set at \(n = 40\), the noise component is generated from a probability model selected by the user, and the non-constant explanatory variables \((x_2, x_3, \text{ and } x_4)\) are pseudorandom \(N(0,1)\) and treated as fixed in repeated samples. For a given sample, the program compares the Wald-based confidence region with the region formed as the intersection of the individual confidence intervals based on the Z criterion.

Under the Wald criterion, the 100(1-\(\alpha\))% confidence region for \(c\beta\) is stated in Equation (6.3.2) as

\[
CR^W = \left\{ r : \left( c\hat{\beta} - r \right) ^{\prime} \left[ s^2 c (x'x)^{-1} c^{\prime} \right]^{-1} \left( c\hat{\beta} - r \right) < \tau \right\}
\]

The bound for the ellipsoidal confidence region (\(\tau\)) is the 100(1-\(\alpha\))% quantile of the central Chi-square(j) distribution where \(j\) is the number of rows in \(c\) and \(r\) (i.e., \(\text{rank}(c) = j \leq k\)). The GAUSS procedure CRegion() provided in the EF library (and described in the program C4CRegion) may be used to compute the boundary of the ellipse in the special case \(j = 2\). GAUSS conducts a bootstrap simulation exercise designed to estimate the
coverage probability of CR\(^W\) for different sample sizes (n) and numbers of trials (m)
selected by the user.

Alternatively, we may use the Bonferroni inequality to form a joint confidence
region based on the individual confidence intervals for the \(i^{th}\) element of \(\mathbf{c}\beta\) as

\[
CR\(^W\) = \left( \mathbf{c} \hat{\beta} - z_{1-\alpha/2j} \left( s^2 \mathbf{c}_i \mathbf{x}' \mathbf{x}^{-1} \mathbf{c}'_i \right)^{1/2} , \mathbf{c} \hat{\beta} + z_{1-\alpha/2j} \left( s^2 \mathbf{c}_i \mathbf{x}' \mathbf{x}^{-1} \mathbf{c}'_i \right)^{1/2} \right)
\]

The component \(z_{1-\alpha/2j}\) is the \((1 - \alpha/2j)\) quantile of the N(0,1) limiting distribution. Thus,
each component of the joint confidence region will have a coverage probability equal to
\(\alpha_i = \alpha/j\). The coverage probability for the Bonferroni confidence region is bounded from
below at \(1 - \sum_{i=1}^{j} \alpha_i\), and the actual coverage probability should be slightly larger than
100(1-\(\alpha\))% (even as \(n\) becomes large).

**User Input:**

First, the user selects one of three probability models to represent the noise components,
\(\varepsilon \sim N(0,1), \varepsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})\), or \(\varepsilon_i \sim \left( \chi^2(3) - 3 \right)/\sqrt{6}\) (i.e., centered and scaled
Chi-square(3) outcomes). Note that each of the candidate probability models has
\(E[\varepsilon_i] = 0\) and \(\text{var}(\varepsilon_i) = 1\). For the bootstrap simulation exercise, the user is asked to
specify the number of bootstrap experiments (1, 2, …, 8) plus the associated sample sizes
\((5 < n < 250)\) and the number of bootstrap trials \((10 < m < 10,000)\) for each simulation
experiment.
**Program Output:**

GAUSS generates \( n = 40 \) observations and reports the LS estimates of \( \beta \) and the individual 90% confidence intervals for each parameter. Note that the 90% intervals may be formed using the second form of \( CR^w \) above with \( j = 1 \) and \( c_i \) equal to the \((1 \times k)\) row vector of zeros with one in the \( i^{th} \) column. GAUSS then plots the joint confidence region for \( \beta_2 \) and \( \beta_3 \) based on the Wald and Bonferroni criteria. The user should note that \( CR^w \) is a green ellipse centered on the LS estimates of \( \beta_2 \) and \( \beta_3 \) (blue dot), and the Bonferroni region is the yellow rectangular. Further, the two regions will include subsets of the parameter space that are not included in the other region. The true values of \( \beta_2 \) and \( \beta_3 \) are indicated with a red dot.

Next, GAUSS estimates the coverage probability of the Wald-based region using a bootstrap simulation procedure. For each of \( m \) bootstrap trials, GAUSS computes the 90% confidence regions and determines if the regions include the true values of \( \beta_2 \) and \( \beta_3 \). The estimated coverage probability for each sample size \( (n) \) is the share of the \( m \) trials in which the region covers the true parameter values. After completing the simulation exercise, GAUSS reports the estimated coverage probabilities, and the user should find that the Wald region exhibits coverage near 90%. The estimated coverage probabilities should approach the intended value of 0.9 as \( n \) and \( m \) become large.
Ex. 6.3.2: Formation of Confidence Regions under the LM Criterion (C6CRLM.gss)

Program Background:

This program demonstrates the formation of 100(1-\(\alpha\))% LM confidence regions for the linear combination \(c\beta\) under the semiparametric linear regression model

\[ y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i \]

The sample size is set at \(n = 40\), and the noise component is generated from a probability model selected by the user. The non-constant explanatory variables \(x_2\) and \(x_4\) are pseudorandom \(N(0,1)\) and treated as fixed in repeated samples. The remaining explanatory variable is generated as \(x_3 = x_2 - 0.6z\) where \(z\) is pseudorandom \(N(0,1)\). For a given sample, the program compares the LM-based 90% confidence region with the associated Wald-based region and the set formed as the intersection of the individual confidence intervals based on the Z criterion.

Under the score form of the LM criterion, the asymptotic 100(1-\(\alpha\))% confidence region for \(c\beta\) is stated in Equation (6.3.4) as

\[
\text{CR}^{\text{LM}} = \left\{ r : (c\hat{b} - r)' \left[ s^2 c (x' x)^{-1} c' \right]^{-1} (c\hat{b} - r) < \tau \right\}
\]

The bound for the ellipsoidal (\(\tau\)) is the 100(1-\(\alpha\))% quantile of the central Chi-square(\(j\)) distribution where \(j\) is the number of rows in \(c\) and \(r\) (i.e., rank(\(c\)) = \(j \leq k\)). Note that the sole difference between \(\text{CR}^{\text{LM}}\) and \(\text{CR}^{\text{W}}\) in Equation (6.3.2) is the estimator of \(\sigma^2\), \(S_i^2\) in
the LM region and $S^2$ in the Wald region. As discussed in Example 6.3.1 (C6CRWald.gss), the GAUSS procedure CRegion() provided in the EF library (and described in the program C4Cregion.gss) may be used to compute the boundary of the ellipse in the special case $j = 2$. GAUSS conducts a bootstrap simulation exercise designed to estimate the coverage probability of $CR^W$ and $CR^{LM}$ for different sample sizes $(n)$ and numbers of trials $(m)$ selected by the user.

Alternatively, we may use the Bonferroni inequality to form a joint confidence region based on the individual confidence intervals for the $i$th element of $c \beta$ as

$$CR^{LM} = \left( c \hat{h} - z_{1-\alpha/2j} \left( s_i^2 c_i (x'x)^{-1} c'_i \right)^{1/2}, c \hat{h} + z_{1-\alpha/2j} \left( s_i^2 c_i (x'x)^{-1} c'_i \right)^{1/2} \right)$$

The component $z_{1-\alpha/2j}$ is the $(1 - \alpha/2j)$ quantile of the $N(0,1)$ limiting distribution. Thus, each component of the joint confidence region will have a coverage probability equal to $\alpha_i = \alpha/j$. The coverage probability for the Bonferroni confidence region is bounded from below at $1 - \sum_{i=1}^{j} \alpha_i$, and the actual coverage probability should be slightly larger than $100(1-\alpha)\%$ (even as $n$ becomes large).

**User Input:**

First, the user selects one of three probability models to represent the noise components, $\varepsilon \sim N(0, I)$, $\varepsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})$, or $\varepsilon_i \sim \left( \chi^2(3) - 3 \right) / \sqrt{6}$ (i.e., centered and scaled Chi-square(3) outcomes). Note that each of the candidate probability models has $E[\varepsilon_i] = 0$ and $\text{var}(\varepsilon_i) = 1$. For the bootstrap simulation exercise, the user is asked to
specify the number of bootstrap experiments (1, 2, ..., 8) plus the associated sample sizes (5 < n < 250) and the number of bootstrap trials (10 < m < 10,000) for each simulation experiment.

Program Output:

GAUSS generates n = 40 observations and reports the LS estimates of β and the individual asymptotic 90% confidence intervals for each parameter. Note that the 90% intervals are based on the restricted estimator of σ², and the restrictions β₂ = 2 and β₃ = 5 are imposed (and are the true parameter values) for this demonstration. Given sᵢ², we can use the second form of CR^{LM} above with j = 1 and cᵢ equal to the (1 x k) row vector of zeros with one in the iᵗʰ column. GAUSS then plots the asymptotic 90% joint confidence regions for β₂ and β₃ based on the LM (cyan ellipse), Wald (green ellipse) and Bonferroni (yellow box) criteria. The user should note that ellipses are centered on the LS estimates of β₂ and β₃ (blue dot), and the true parameter values are indicated with a red dot. If sᵢ² < s², then the user should find that LM region lies within the Wald region.

Next, GAUSS estimates the coverage probability of the LM and Wald regions using a bootstrap simulation procedure. For each of m bootstrap trials, GAUSS computes the 90% confidence regions from the initial sample and determines if the regions include the true values of β₂ and β₃. The estimated coverage probability for each sample size (n) is the share of the m trials in which the region covers the true parameter values. After completing the simulation exercise, GAUSS reports the estimated coverage probabilities, and the user should find that the LM and Wald regions exhibit coverage near 90%. The estimated coverage probabilities should approach the intended value of 0.9 as n and m
become large.
Ex. 6.4.1: Conduct and Power of the One-Sided Z Test of $c\beta \leq r$ (C6Z.gss)

Program Background:

This program demonstrates the conduct of the one-sided Z test of the linear inequality restrictions $c\beta \leq r$ for the semiparametric linear regression model

$$y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i$$

The sample size is $n = 40$, the noise component is generated under a probability model selected by the user, and the explanatory variables ($x_2$, $x_3$, and $x_4$) are pseudorandom N(0,1) and held fixed in repeated samples. The first objective of the program is to conduct a Z test of $H_0 : \beta_2 \leq r_1$ and $\beta_3 \leq r_2$, where $r_1$ and $r_2$ are selected by the user. The second objective of the program is to simulate the power function of the Z test of the null hypothesis $H_0 : \beta_i \leq 0$. The simulation exercise is constructed from bootstrap replication of an initial sample, and the estimated power function is compared to the asymptotic power function based on the limiting N(0,1) distribution of the Z statistic.

If $j = 1$, the Z statistic for a one-sided test of the null hypothesis $H_0 : c\beta \leq r$ is stated in Equation (6.4.8)

$$Z = \frac{c\hat{\beta} - r}{\left[ S^2c(x'x)^{-1}c' \right]^{1/2}}$$

Under $H_0$, $Z$ is asymptotically N(0,1). If the alternate hypothesis $H_a : c\beta = r + n^{-1/2}\delta$ is
correct (i.e., Pitman drift), then the asymptotic distribution of $Z$ is $N\left(\delta/\left[\sigma^2 \Xi^{-1}\epsilon'\right]^{1/2},1\right)$

where $\left(n^{-1}x'x\right) \to \Xi$ by assumption. The power function for the asymptotic level $\alpha$ test is

simply $\pi(\delta) = Pr\left[Z \geq z_{1-\alpha}\right]$, as stated in Equation (6.4.4), and the critical value $z_{1-\alpha}$ is the $100(1-\alpha)\%$ quantile of the $N(0,1)$ distribution. To test $j > 1$ joint null hypotheses of the form $H_0: \epsilon \beta \leq r_i$ with overall level $\alpha$, we can conduct each one-sided test at level

$\alpha_i = \alpha/j$ under the Bonferroni criterion.

**User Input:**

First, the user selects one of three probability models to represent the noise components,

$\epsilon \sim N(0,1)$, $\epsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})$, or $\epsilon_i \sim \left(\chi^2(3)-3\right)/\sqrt{6}$ (i.e., centered and scaled Chi-square(3) outcomes). Note that each of the candidate probability models has $E[\epsilon_i] = 0$ and $\text{var}(\epsilon_i) = 1$. Then, the user is asked to select the two elements of the vector $r$ for a test of the linear restriction $H_0: \beta_2 \leq r_1$ and $\beta_3 \leq r_2$, and the elements of $r$ are restricted to $-10 \leq r_1, r_2 \leq 10$. For the power simulation exercise, the user is asked to specify the number of bootstrap experiments (1, 2, or 4) plus the associated sample sizes ($5 < n < 250$) and the number of bootstrap trials ($10 < m < 10,000$) for each simulation experiment.

**Program Output:**

GAUSS first generates $n = 40$ observations and reports the LS estimates of $\beta$ and the
associated standard error estimates. Given the value of $r$ selected by the user, GAUSS uses the MLR_Z() procedure from the EF library to compute the observed Z test statistics and the asymptotic p-values for the tests of the $j = 2$ hypotheses. GAUSS briefly explains the use of procedure, and the standard call to the procedure takes the form

$$\{z, pvalues\} = \text{MLR}_Z(c, \text{bhat}, r, \text{cov})$$

where ‘bhat’ is the vector of LS estimates $b$ and ‘cov’ is the estimated covariance matrix, $\text{cov}(\hat{b}) = s^2 (XX')^{-1}$. Note that the returned values ‘z’ and ‘pvalues’ are ($j \times 1$) vectors, and the user should compare the reported p-values with $\alpha_i = \alpha / j$ to conduct a Bonferroni joint test with overall size $\alpha$.

After completing the bootstrap simulation exercise, GAUSS plots the estimated and asymptotic power functions for the Z test of $H_0 : \beta_i \leq 0$. Under Pitman drift, the domain of the power functions is represented as $\beta_i \sqrt{n}$. The user should find that the estimated power function approaches the asymptotic power function $\pi(\delta)$ as the number of bootstrap trials ($m$) and the sample size ($n$) increase. Also, the simulated power should roughly correspond to the asymptotic power under $H_0$, $\pi(0) = 0.10$, and $\pi(\delta) \to 1$ as $\beta_i \sqrt{n} \to \infty$. 
Ex. 6.4.2: Formation and Coverage of Joint Upper-bounded Confidence Intervals under the Bonferroni and Z Criteria (C6Joint.gss)

**Program Background:**

This program demonstrates the formation of joint upper-bounded confidence intervals for parameters of the semiparametric linear regression model

\[ y_i = 40 + 2x_{i2} + 5x_{i3} - 8x_{i4} + \varepsilon_i \]

The sample size is \( n = 40 \), the noise component is generated under a probability model selected by the user, and the explanatory variables \( (x_2, x_3, \text{ and } x_4) \) are pseudorandom \( N(0,1) \) and held fixed in repeated samples. The first objective of the program is to compute the asymptotic 90% upper-bounded confidence intervals for \( \beta_2 \) and \( \beta_3 \) under the Z criterion. The second objective of the program is to simulate the coverage probability of the joint upper-bounded region for \( \beta_1 \) and \( \beta_2 \) formed from the separate Z-regions under the Bonferroni inequality. The simulation exercise is conducted by bootstrap resampling of an initial sample, and the results are compared to the intended coverage probability (0.9) for the joint region.

Under the Z criterion, the 100(1 – \( \alpha \))% lower bounded confidence interval for \( c\beta \) is stated in Equation (6.4.9)

\[
CR_Z = \left\{ r : r > \hat{c}\beta - z_{1-\alpha}\left[ s^2c(x'x)^{-1}c' \right]^{1/2} \right\}
\]
Accordingly, the $100(1 - a)\%$ upper bounded confidence interval for $\mathbf{c} \mathbf{\beta}$ is

$$
\text{CR}^Z = \left\{ r : r < c \hat{\mathbf{b}} + z_{1-\alpha} \left[ s^2 \mathbf{c} (\mathbf{x}'\mathbf{x})^{-1} \mathbf{c}' \right]^{1/2} \right\}
$$

as stated in Equation (6.4.10). To construct a joint confidence region (upper or lower bounded) with overall coverage probability $100(1 - \alpha)\%$ from $m$ components, we can use the Bonferroni inequality to adjust the coverage probability of the component intervals, $\alpha_i = \alpha/m$. For example, the lower-bounded region for an individual interval is

$$
\text{CR}^Z = \left\{ r : r > c \hat{\mathbf{b}} - z_{1-\alpha/m} \left[ s^2 \mathbf{c}_i (\mathbf{x}'\mathbf{x})^{-1} \mathbf{c}_i' \right]^{1/2} \right\}
$$

From Equation (6.4.11), the coverage probability for the joint region has a lower bound equal to $1 - \sum_{i=1}^{m} \alpha_i = 1 - \alpha$.

**User Input:**

First, the user selects one of three probability models to represent the noise components, $\epsilon \sim N(0, \mathbf{I})$, $\epsilon_i \sim \text{Uniform}(-\sqrt{3}, \sqrt{3})$, or $\epsilon_i \sim \left( \chi^2(3) - 3 \right) / \sqrt{6}$ (i.e., centered and scaled Chi-square(3) outcomes). Note that each of the candidate probability models has $E[\epsilon_i] = 0$ and $\text{var}(\epsilon_i) = 1$. For the coverage simulation exercise, the user is asked to specify the number of bootstrap experiments ($1, 2, \ldots, 8$) plus the associated sample sizes ($5 < n < 250$) and the number of bootstrap trials ($10 < m < 10,000$) for each simulation experiment.
Program Output:

GAUSS first generates n = 40 observations and reports the LS estimates of $\beta$ and the associated standard errors. GAUSS briefly explains the use of CR_Joint() procedure from the EF library as a means to compute upper or lower confidence bounds for $c\beta$. The call to the GAUSS procedure takes the form

$$\text{CR_Joint}(c, \text{bhat}, \text{cov}, \text{alpha}, \text{up})$$

where ‘bhat’ is the vector of LS estimates $b$ and ‘cov’ is the estimated covariance matrix, $\text{cov}(\hat{\beta}) = s^2 (X'X)^{-1}$, ‘alpha’ is the complement of the coverage probability, and ‘up’ is an indicator (binary) variable that is set to one for upper-bounded intervals. Note that $e$ may include more than one row so that the procedure returns multiple bounds.

After completing the bootstrap simulation exercise, GAUSS reports the estimated coverage probability for each sample size (n). The user should find that the simulated coverage probabilities may be slightly larger than the intended rate (0.90) because the Bonferroni inequality provides a lower bound for the actual coverage probability.
Ex. 6.6.1: Housing Demand Demonstration by Dr. Thomas L. Marsh (C6Marsh.gss)

Program Background:

The purpose of this GAUSS program is to demonstrate the properties of the LS estimator in the context of a real data set. Dr. Marsh constructs a semiparametric linear regression model of the demand for newly manufactured homes

\[ \ln(Q_i) = \beta_1 + \sum_{k=2}^{6} \beta_k \ln(x_{ik}) + \varepsilon_i \]

The dependent variable is the natural logarithm of the number of newly manufactured homes placed for residential use in the United States \( Q_i \), and the non-constant explanatory variables are the natural logarithms of the average real sales price \( x_{i2} \), the average term length of installment loans \( x_{i3} \), the average interest rate \( x_{i4} \), the share of manufactured housing credit in total installment credit \( x_{i5} \), and the number of new homes shipped to retailers \( x_{i6} \). The data are annual U.S. observations for 1975-94, and the model is based on the research reported in the article Glennon, D., “Estimating the Income, Price, and Interest Elasticities of Housing Demand,” *Journal of Urban Economics*, 25(1989):219-29.

The analysis conducted in the program is divided in three parts. First, the least squares estimates \( \hat{b} \), the estimated standard errors based on \( s^2(\hat{X}'\hat{X})^{-1} \), and the Z statistics for individual tests of parameter significance \( H_0 : \beta_k = 0 \) (e.g., see Equation (6.4.8)) are
Second, GAUSS conducts a bootstrap simulation exercise designed to demonstrate the sampling properties of the LS estimator and associated test statistics and confidence regions. Following the discussion in Section A.3, the bootstrap procedure is based on sampling with replacement from the LS residuals, \( \hat{\epsilon} \), to form a replicate of the noise component, \( \varepsilon_j^* \), for each trial, \( j = 1, \ldots, m \). The regressors \( \mathbf{x} \) are treated as fixed in repeated samples, the LS estimates of \( \beta \) are treated as the true parameters values, and the replicated outcomes of the dependent variable are generated as \( y_j^* = \mathbf{x} \hat{\beta} + \varepsilon_j^* \) for each \( j \). GAUSS computes the estimates \( \hat{\beta}_j^* \) and \( s_j^2 \) for each trial and stores the results. The sample moments of the bootstrap outcomes are used to estimate the mean and standard error of \( \hat{\beta} \) and \( S^2 \) (see the section “Program Output” below for details).

The sampling properties of Wald and LM test statistics are also demonstrated with the bootstrap outcomes. GAUSS uses the Wald statistic from Equation (6.2.2)

\[
W = (c\hat{\beta} - r)' \left[ S^2 c (x'x)^{-1} c' \right]^{-1} (c\hat{\beta} - r)
\]

and the operational version of the LM statistic in score form

\[
LM = (c\hat{\beta} - r)' \left[ S^2 c (x'x)^{-1} c' \right]^{-1} (c\hat{\beta} - r)
\]

from Equation (6.2.29) to test \( H_0 : \beta_2 = \hat{\beta}_2 \) and \( \beta_4 = \hat{\beta}_4 \), \( H_0 : \beta_2 = \hat{\beta}_2 \), and \( H_0 : \beta_4 = \hat{\beta}_4 \) for each bootstrap trial. Given that \( \hat{\beta}_2 \) and \( \hat{\beta}_4 \) are treated as the true parameter values under the bootstrap procedure, the bootstrap test results may be used to simulate the Type I error rates for the tests. GAUSS also simulates the Type I error probabilities for Z tests
of the one-sided null hypotheses $H_0: \beta_4 \leq \hat{b}_4$ and $H_0: \beta_4 \geq \hat{b}_4$.

The bootstrap outcomes are also used to simulate the coverage probabilities for Wald- and LM-based confidence regions. The region based on the Wald criterion is stated in Equation (6.3.2)

$$\text{CR}^W = \left\{ r : \frac{(\hat{c} \hat{b} - r)' \left[ c (x' x)^{-1} c' \right] (\hat{c} \hat{b} - r)}{s^2} < \tau \right\}$$

and the region based on the score form of the LM statistic is

$$\text{CR}^{LM} = \left\{ r : \frac{(\hat{c} \hat{b} - r)' \left[ c (x' x)^{-1} c' \right] (\hat{c} \hat{b} - r)}{s^2} < \tau \right\}$$

from Equation (6.3.4). The critical value of the $\alpha$-level asymptotic tests ($\tau$) is the bound for the ellipsoidal regions. Accordingly, GAUSS computes the Wald and LM 90% confidence regions for $\beta_2$ and $\beta_4$ and compares the simulated coverage probabilities with the intended coverage rate (0.9).

The third part of the program is designed to demonstrate the impact of increasing sample size on the properties of the estimators, and we consider sample sizes that are integer multiples (2, 4, 8, and 32) of $n$. To replicate samples generated with size larger than $n = 20$, we must form an “initial” set of observations of $x$ of the desired size. GAUSS forms a matrix of explanatory variables with sample size $nJ$ as $x_j = (1_J \otimes x)$, where $\otimes$ is the Kronecker product and $1_J$ is a column vector of $J$ ones. Then, the dependent variables may be formed by drawing $nJ$ outcomes with replacement from $\hat{e}$ and combining to create $y_j^* = x_j \hat{b} + \epsilon_j^*$ as above.
User Input:

The user does not provide any direct input but can repeat the demonstration as desired.

Program Output:

Given the sample of \( n = 20 \) observations, GAUSS computes the LS estimates of \( \beta \) and the noise variance parameter, \( \sigma^2 \). The LS estimates \( \mathbf{b} \) plus the estimated standard errors and the \( Z \) statistics for individual tests of \( H_0 : \beta_k = 0 \) are reported to the user. The user should note that the estimated parameters represent elasticities due to the log-log form of the demand function. For example, the estimated price and interest elasticities of housing demand are –0.44 and 0.15 (Glennon reports estimated price and interest elasticities of –0.58 and –0.37).

In the second part of the program, GAUSS generates \( m = 1000 \) bootstrap samples by residual resampling and reports the sample average and standard deviation of the replicated bootstrap outcomes

\[
\bar{b}_k = m^{-1} \sum_{j=1}^{m} b_{jk}^* \quad \text{and} \quad \sqrt{\frac{(m-1)^{-1}}{m}} \sum_{j=1}^{m} (b_{jk}^* - \bar{b}_k)
\]

The sample averages are bootstrap estimators of the expected value of \( \hat{\beta}_k \), and the sample standard deviations are the bootstrap estimators of the square roots of \( \text{var} \left[ \hat{\beta}_k \right] \). Given that \( \mathbf{b} \) is the true parameter vector in the bootstrap simulation, the user should find that the sample averages are close to the LS estimates due to the unbiasedness of \( \hat{\beta} \).
For the bootstrap replications of the Wald, LM, and Z tests, GAUSS reports the share of the m bootstrap replicates of the test statistics that exceeded the 5% critical value (τ) for each null hypothesis stated above. Given that the null hypotheses are correct, the reported shares are bootstrap estimates of the Type I error rates, and the user should find that the estimates are reasonably close to 5% (the intended size of the tests). GAUSS then reports the average upper and lower bound for the asymptotic Wald and LM 90% confidence intervals for \( \beta_2 \) and \( \beta_4 \) computed from the m bootstrap replicates. The share of the m intervals that actually included the true parameter values is also reported, and the user should find that estimated coverage probabilities are similar to the intended coverage rate (0.9). Finally, GAUSS plots the empirical distribution functions (EDFs) for the replicated outcomes of \( \hat{\beta} \) and \( S^2 \). The bootstrap averages \( \overline{b}_k \) are denoted with a solid yellow line, and the true parameter values \( b \) are denoted with a red line.

In the third part of the program, GAUSS first reports and plots the average of the bootstrap replicates for each extended sample size (2n, 4n, 8n, and 32n). The red line indicates the true values of \( \beta \) and \( \sigma^2 \) in the bootstrap simulation exercise. Next, the consistency property \( \hat{\beta} \xrightarrow{p} \beta \) is demonstrated with boxplots of the bootstrap replicates. For each element of \( \beta \) and \( \sigma^2 \), GAUSS forms the box and whisker plot based on the m bootstrap replicates. The boxes represent the interquartile range (middle 50%) of the outcomes, and the red line indicates the true parameter value. The user should find that the boxes and whiskers collapse on the red line as the extended sample size (nJ) increases.

Finally, GAUSS computes the EDF and associated limiting normal CDF for each parameter and extended sample size. The mean and standard deviation of the absolute
discrepancy between the distributions is also computed and reported by GAUSS. The user should note that these statistics should approach zero as \( nJ \) becomes large due to the asymptotic normality of \( \hat{\beta} \). GAUSS also plots the EDF and CDF for each case, and the user should also be able to visually observe the EDF approach the CDF as \( nJ \) increases.