# Applications of interval analysis to selected topics in statistical computing

by

Kevin Wright

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Statistics

Major Professor: William Kennedy

Iowa State University

Ames, Iowa

1999

# Contents

# Chapter 1

# GENERAL INTRODUCTION

## 1.1 Background

Since the beginning of mathematics, error has been a part of calculations, and reducing error has been an important component of research in many fields. Even with digital computers, error is still present. There are two common types of error in statistical calculations: error from rounding and algorithmic error.

Error from rounding occurs because the computer has only a finite set of numbers to approximate the whole real line. Even simple fractions are often impossible to represent exactly as a decimal on a computer. An additional complication arises from the requirement that calculations be performed in base 2 instead of base 10, further increasing the potential for error during base conversion of numbers.

Algorithmic error typically occurs when an infinite algorithm is terminated after a finite amount of time. For example, infinite series are frequently used to calculate functions such as exp, sin, cos, etc. Truncating the infinite series to a finite number of terms will result in some error, regardless of how many digits of precision are used for the calculations.

Given that error exists, logical questions that follow are ones like "How large is the error" and "Is the error likely to be a problem?" Consider the system of linear equations $Ax = b$ where

$$A = \begin{pmatrix} 64919121 & -159018721 \\ 41869520.5 & -102558961 \end{pmatrix} \text{ and } b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{1.1}$$

The true solution to this set of equations is

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 205117922 \\ 83739041 \end{pmatrix}. \tag{1.2}$$

As reported by Bohlender (1990), even using IEEE double-precision arithmetic yields the following answer:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 102558961 \\ 41869520.5 \end{pmatrix}, \tag{1.3}$$

which is completely wrong.

As a second example, Hammer et al. (1993a) consider an example with two real vectors,

$$x = (10^{20}, 1223, 10^{18}, 10^{15}, 3, -10^{12})$$
$$y = (10^{20}, 2, -10^{22}, 10^{13}, 2111, 10^{16}).$$

The dot product of these two vectors is $x \cdot y = 10^{40} + 2446 - 10^{40} + 10^{28} + 6333 - 10^{28} = 8779$. On *all* standard computers, the result of this dot product is zero. In contrast, the use of interval arithmetic (introduced later) guarantees that the scalar product is somewhere in the interval $[0, 1.93429e + 25]$, possibly an acceptable answer, but likely to spur further investigation into the reasons behind the wide interval. It would be possible in this example to use a computer with more mantissa digits and obtain an accurate answer, but that assumes that one knows of the need to use more precision, and in any case, no matter how many digits of accuracy are maintained, it is always possible for actual computations to exceed this accuracy. Clearly, there is a need for additional understanding of error and ways to control the error. One of the tools which takes a step in the direction of these goals is interval analysis.

## 1.2 Introduction to Interval Analysis

A suitable introduction to interval analysis can be found in Moore (1979). The basic idea of interval analysis is to work with an interval not only as a set of numbers, but also as a number by itself. A real interval $\mathbf{x}$ is defined $\mathbf{x} = [\underline{x}, \overline{x}]$ where $\underline{x}$ and $\overline{x}$ are real numbers with $\underline{x} \leq \overline{x}$. Let $\mathbf{x} = [\underline{x}, \overline{x}]$ and $\mathbf{y} = [\underline{y}, \overline{y}]$. Arithmetic operations for intervals are defined:

$$\mathbf{x} * \mathbf{y} = \{x * y : x \in \mathbf{x}, y \in \mathbf{y}\} \text{ for } * \in \{+, -, \cdot, \div\}.$$

Closed–form expressions of the basic interval arithmetic operators also exist and are the means by which rational expressions are actually computed. The closed–form expressions are:

$$\mathbf{x} + \mathbf{y} = [\underline{x} + \underline{y}, \overline{x} + \overline{y}]$$
$$\mathbf{x} - \mathbf{y} = [\underline{x} - \overline{y}, \overline{x} - \underline{y}]$$
$$\mathbf{x} \cdot \mathbf{y} = [\min(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}), \max(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y})]$$
$$1/\mathbf{y} = [1/\overline{y}, 1/\underline{y}], \quad 0 \notin \mathbf{y}$$
$$\mathbf{x}/\mathbf{y} = \mathbf{x} \cdot (1/\mathbf{y}), \qquad 0 \notin \mathbf{y}$$

Examples of interval arithmetic:

$$[1, 1] + [-2, 5] = [-1, 6]$$

$$[-2, 3] \cdot [1, 4] = [-8, 12].$$

From these basic definitions, the following properties are observed to hold for intervals:

$$\mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z}$$

$$\mathbf{x} \cdot (\mathbf{y} \cdot \mathbf{z}) = (\mathbf{x} \cdot \mathbf{y}) \cdot \mathbf{z}$$

$$\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$$

$$\mathbf{x} \cdot \mathbf{y} = \mathbf{y} \cdot \mathbf{x}$$

for any intervals $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$. Interval addition and multiplication is therefore both associative and commutative for both addition and multiplication. One of the ways in which interval arithmetic differs from real arithmetic is that interval arithmetic is not distributive, but subdistributive only. As a consequence, interval subtraction and division are not the inverse of interval addition and multiplication. For example:

$$[0, 1] - [0, 1] = [-1, 1]$$

$$[1, 2]/[1, 2] = [1/2, 2].$$

For interval arithmetic, the subdistributive law states that for intervals $\mathbf{x}, \mathbf{y}$, and $\mathbf{z}$,

$$\mathbf{x} \cdot (\mathbf{y} + \mathbf{z}) \subseteq \mathbf{x} \cdot \mathbf{y} + \mathbf{x} \cdot \mathbf{z}.$$

Further basic definitions and properties of intervals can be found in Ratscheck and Rokne (1984).

## 1.3 Interval Inclusion of Functions

Extending the ideas of real functions to interval functions is not straightforward. Several approaches are possible. An *interval function* is defined as an interval–valued function of one or more interval arguments.

A function $\mathbf{f}(\mathbf{x}_1, \ldots \mathbf{x}_n)$ is said to be an *interval extension* of $f(x_1, \ldots, x_n)$ if

$$\mathbf{f}([x_1, x_1], \ldots, [x_n, x_n]) = f(x_1, \ldots, x_n)$$

for all $x_i, i = 1, \ldots, n$.

Let $f(x_1, \ldots, x_n)$ be a real–valued function of $n$ real variables. For $x_1, \ldots, x_n$ defined over the intervals $\mathbf{x}_1, \ldots, \mathbf{x}_n$ respectively, the *united extension* of $f$ over $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is given by

$$\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \{f(x_1, \ldots, x_n) : x_1 \in \mathbf{x}_1, \ldots, x_n \in \mathbf{x}_n\}.$$

The united extension of a function is unique, however, it need not be an interval.

One kind of interval extension that is fairly logical is the so–called *natural interval extension* which is defined this way: let $f(x_1, \ldots, x_n)$ be a rational function of $n$ variables. Consider any sequence of arithmetic steps which serve to evaluate $f$ with given arguments $x_1, \ldots, x_n$. Suppose the arguments $x_i$ are replaced by corresponding intervals $\mathbf{x}_i, (i = 1, \ldots, n)$ and the arithmetic steps in the sequence used to evaluate $f$ are replaced by the corresponding interval arithmetic steps. The result will be an interval $\mathbf{f}(\mathbf{x}_1, \ldots, \mathbf{x}_n)$. This interval contains the value of $f(x_1, \ldots, x_n)$ for all $x_i \in \mathbf{x}_i, (i = 1, \ldots, n)$.

An interval–valued function $\mathbf{f}$ is said to be *inclusion monotonic* if $\mathbf{f}(\mathbf{x}) \subseteq \mathbf{f}(\mathbf{y})$ whenever $\mathbf{x} \subseteq \mathbf{y}$. A fundamental theorem from interval analysis states that rational interval functions are inclusion monotonic.

It should be noted that an interval extension need not be unique, but can depend on the form of the real function. For example, while the following three expressions all represent the same real function,

$$f_1(x) = x^2 - x + 1$$

$$f_2(x) = (x - \frac{1}{2})^2 + \frac{3}{4}$$

$$f_3(x) = x \cdot (x - 1) + 1,$$

the corresponding natural interval extensions

$$\mathbf{f}_1(\mathbf{x}) = \mathbf{x}^2 - \mathbf{x} + 1$$

$$\mathbf{f}_2(\mathbf{x}) = (\mathbf{x} - \frac{1}{2})^2 + \frac{3}{4}$$

$$\mathbf{f}_3(\mathbf{x}) = \mathbf{x} \cdot (\mathbf{x} - 1) + 1$$

do *not* represent the same function. For illustration of how these functions differ, consider the following results:

$$\mathbf{f}_1([0, 2]) = [-1, 5]$$

$$\mathbf{f}_2([0, 2]) = \left[\frac{3}{4}, 3\right]$$

$$\mathbf{f}_3([0, 2]) = [-1, 3].$$

Thus, attention and care must given to the choice of an interval extension in order to obtain the narrowest possible interval result. As shown in figure 1.1, the true range of $f$ over $[0, 2]$ is $[\frac{3}{4}, 3]$, which is precisely that computed by the interval function $\mathbf{f}_2$. As shown by Hansen (1997), this is because $\mathbf{x}$ appears in the expression of $\mathbf{f}_2$ only once. In general, when a given interval argument appears only once in a function, the evaluation of the interval function produces a *sharp* interval that exactly matches the true range of the function over the given interval. When a given interval argument appears more than once in a function, the evaluation of the function may produce an interval wider than the sharp enclosure. This characteristic of interval analysis is referred to as the *dependency problem*. In some cases the dependency problem can be eliminated by simply changing a definition. For example, suppose $\mathbf{x} = [-1, 2]$. Then thinking of the square operation as simply multiplying two intervals together, $\mathbf{x}^2 = \mathbf{x} \cdot \mathbf{x} = [-1, 2] \cdot [-1, 2] = [-2, 4]$. This is not a sharp interval and contains negative real numbers–obviously not a desirable feature. It is possible to fix this problem with an appropriate definition of $\mathbf{x}^2$ as $\mathbf{x}^2 = \{x^2 : x \in \mathbf{x}\}$. Then, for example, $[-1, 2]^2 = [0, 4]$.

With the basic aspects of interval analysis thus defined, how should interval analysis be applied to particular problems? Walster (1988) presents a set of principles to guide the thinking of interval analysis:

1. Interval algorithms should bound all sources of error

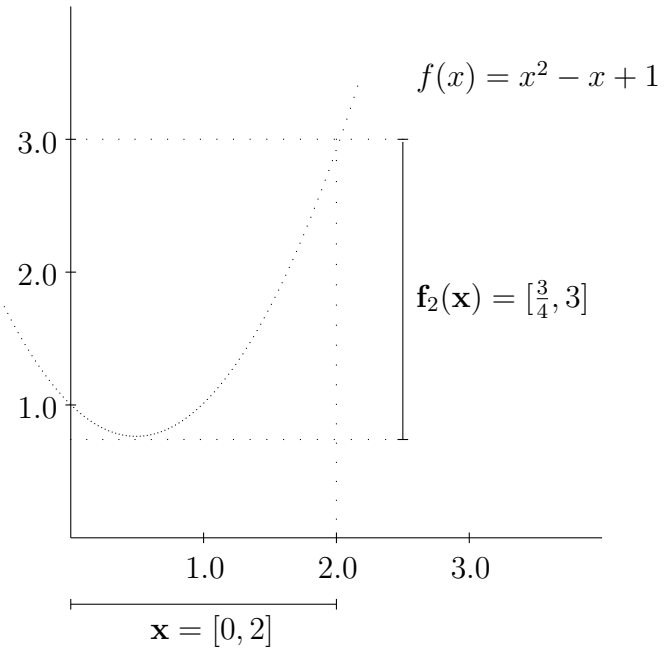


Figure 1.1: Example of an interval extension.

2. Interval input/output conventions should be consistent with people's normal interpretation of numerical accuracy

3. The application of interval algorithms should be universal

4. Where interval algorithms currently do not exist, we should get to work developing them rather than abandoning the principle of universal applicability.

## 1.4 Interval Analysis and Digital Computing

Efforts to produce accurate numerical results have been around nearly as long as number systems. The current widelyused approach for representation of numbers on computer systems dates back to the late 1970s with the establishment of a task force by the IEEE. From that work, and from the work of a second task force, emerged two stan dards ANSI/IEEE (1985, 1987). A brief overview and history of the hardware aspects of the standards can be found in Cody (1988).The widelyused Intel 80x87 series of chips conform to this standard as do the numerical processors in many workstations. The standards focus mainly on the representation of floating point numbers, but also include rules for handling NaNs, overflow, underflow, rounding, etc. As regards to rounding floating point numbers, the first standard ANSI/IEEE (1985) specifies that the floating point processor support rounding to zero, nearest, $+\infty$, and $-\infty$. Software support for allowing user access to the various rounding modes, while available, is generally not as easily available

as would be desired. As one simple example, Fortran 77 has no direct support for control of the rounding mode. Programmers wishing to utilize the special rounding modes need specialized software packages and/or detailed knowledge of a com puter's inner workings. Examples of such packages include programming languages like PascalXSC Hammer et al. (1993b), ACRITHXSC Walter (1993), FortranSC Walter (1988); Metzger (1988), and CXSC Lawo (1993). A comparison of these environments (and others) can be found in Kearfott (1996a). These scientific computing languages typ ically have some combination of support for interval data types, dynamic vectors and arrays, dot product expressions, rounding control, a large set of standard mathematical functions, operator overloading and userdefined operators.

For example, consider a software package with functions RoundUp and RoundDown that are used to control the rounding mode. Imagine using these functions on a computer which has three digits in floatingpoint numbers. The interval enclosure of 1/3 would be computed as

$$[\text{RoundDown}(1/3); \text{RoundUp}(1/3)] = [0.333; 0.334] = [0.33^{4}_{3}].$$

Use of the directed rounding rigorously guarantees that the resulting interval contains the true value. By using the monotonicity property of rational interval functions and by using other properties, more complicated functions can be constructed. The exponential function, for example, can be computed via a series expression,

$$\exp(x) = \sum_{n=0}^{\infty} \frac{x^n}{n!}$$

which converges for all real $x$. For $x \leq 0$, the inequality

$$\left| \exp(x) - \sum_{n=0}^{N} \frac{x^n}{n!} \right| \leq \frac{x^{N+1}}{(N+1)!}$$

can be used to calculate and enclosure of $\exp(x)$ by rearranging the expression as

$$\sum_{n=0}^{N} \frac{x^n}{n!} - \frac{x^{N+1}}{(N+1)!} \leq \exp(x) \leq \sum_{n=0}^{N} \frac{x^n}{n!} + \frac{x^{N+1}}{(N+1)!}$$

and using interval arithmetic to calculate the rational functions which bound exp(x) above and below, then taking the hull of the these intervals.

An example from Krämer (1993) illustrates a completely different approach to inter val enclosures of functions, in this case the natural logarithm. Beginning with $x_0 > 0$ and $y_0 > 0$, the sequences

$$x_{n+1} := \sqrt{x_n \frac{x_n + y_n}{2}}$$

$$y_{n+1} := \sqrt{x_n \frac{y_n + y_n}{2}}$$

both converge to $\log(\frac{x_0}{y_0})$. At each step,

$$\lim_{n\to\infty} x_n = \lim_{n\to\infty} y_n \in [\min(x_n, y_n), \max(x_n, y_n)].$$

To compute $\log(x)$, set $x_0 = x$ and $y_0 = 1$. The sequences $x_n$ and $y_n$ can either be computed with scalars (and careful use of rounding modes) or with intervals, in which case $log(x) \in Hull(\mathbf{x}_n; \mathbf{y}_n)$.

An extensive set of algorithms for computing standard functions and inverse standard functions for interval arguments can be found in Brane (1988) and Krämer (1988).

## 1.5 Automatic Differentiation

Automatic differentiation is a method for computing derivatives of functions without resorting to symbolic manipulation or numerical approximations, but instead using standard differentiation rules and propagation of numerical values. No explicit expression for the derivatives is required.

Automatic differentiation frequently arises in the context of interval analysis algorithms. Algorithms in interval analysis which use derivatives of functions need an accurate and efficient means by which to compute the derivative of the function. Numerical derivatives are generally insufficiently accurate for interval calculations. Symbolic derivatives can be difficult to calculate efficiently and may be difficult to include in an algorithm, either by programming symbolic manipulations from scratch, or by linking to an external program (e.g. Maple or Mathematica) that actually calculates the symbolic derivatives. Corliss (1988) makes the surprising claim that with automatic differentiation, "on most computers, it is less expensive to evaluate the first few derivatives of $f$ than it is to evaluate $f$ itself."

For simplicity of expression, the techniques of automatic differentiation typically use Taylor coefficients instead of derivatives directly. For the Taylor coefficients of a real function $f$ , the following notation is used:

$$(f)_k := \frac{1}{k!} f^{(k)}(x_0) := \frac{1}{k!} \frac{d^k f}{dx^k}|_{x=x_0}, k = 0, 1, \ldots$$

Using this notation, $f(x_0) = (f)_0$, $f'(x_0) = (f)_1$, and $f^{(k)}(x_0) = k!(f)_k$ . The rules from calculus for differentiation of basic arithmetic functions lead to the following Taylor coefficients:

$$(f + g)_k = (f)_k + (g)_k$$

$$(f - g)_k = (f)_k - (g)_k$$

$$(f \cdot g)_k = \sum_{j=0}^{k} (f)_j \cdot (g)_{k-j}$$

$$(f/g)_k = \frac{1}{(g)_0} \left( (f)_k - \sum_{j=0}^{k-1} (f/g)_j \cdot (g)_{k-j} \right)$$

For the exponential function,

$$(w)_0 = \exp((f)_0)$$

$$(w)_k = \frac{1}{k}\sum_{j=0}^{k-1}(k-j)\cdot(w)_j\cdot(f)_{k-j}, k \geq 1$$

Similar rules exist for other functions (e.g. sin, cos).

## 1.6 Previous Applications of Interval Analysis to Statistics

Applications of interval analysis to statistical problems have previously appeared in the areas listed below.

### 1.6.1 Score Model

One early application of interval arithmetic to statistics is due to Walster (1988), where consideration is given to a score model $x_i = t + \epsilon_i$ where $\epsilon_i \sim (0, \sigma_\epsilon)$ with the observations being independent. The usual estimate of $t$ is $\overline{x}$, which has the desirable properties of being an unbiased estimate of $t$ and of having an increasing precision as the number of observations, $n$, increases, i.e. $Var(\overline{x}) = \sigma_\epsilon^2/n$. If it is possible to bound the errors $\epsilon_i$, then for some finite $\delta$, $-\delta < \epsilon_i < \delta$ for each $i$ and it is possible to construct interval data as $x_i = [x_i - \delta, x_i + \delta]$. One interval estimate of $t$ is to compute the mean of the interval data, $\overline{x} = [\overline{x} - \delta, \overline{x} + \delta]$. Unfortunately, this estimator has a fixed width, regardless of the amount of data collected. As an alternative, since the errors are bounded, it is the case that $t \in [x_i - \delta, x_i + \delta]$ for every $i$ and thus

$$\mathbf{t} = \bigcap_{i=1}^{n}[x_i - \delta, x_i + \delta] = [\max(x_i) - \delta, \min(x_i) + \delta]$$

can also be used to estimate t. This estimator has at least the potential of decreasing in width as more data is collected.

### 1.6.2 Probability Distributions

From random number simulation to critical points of hypothesis tests, statistical distributions are an important part of statistical computing. Wang and Kennedy (1990) present a selfvalidated method for the calculation of bivariate Normal probabilities over rectangular regions using interval analysis. Their method served as the basis for comparing several different methods for computing prob abilities of a bivariate Normal distribution when issues of speed and accuracy are of concern.

In dimensions higher than two, Wang and Kennedy (1992) use a Taylor series ex pansion of the multivariate Normal integral and automatic differentiation to calculate

interval enclosures of probabilities over rectangular regions. The method is used to conclude that the Taylor series approximation gives more accurate results than an algorithm by Schervish.

More recently, Wang and Kennedy (1994a) developed various intervalbased ways for obtaining selfvalidated probabilities and percentiles of several univariate distributions, including Normal, Incomplete Gamma, Incomplete Beta, and Noncentral Chi-Square. The work was expanded to include the Central and Noncentral F distributions in Wang and Kennedy (1995).

Using a slightly different approach, Wang (1994) used a MasPar (Massively Parallel) machine for selfvalidated probabilities from multivariate Normal and multivariate t distributions.

### 1.6.3 Least Squares

In the traditional Least Squares methodology, dependent variables $y_1, y_2, \ldots, y_n$ are assumed to be related to independent variables $X_1, X_2, \ldots, X_n$ by the relation

$$y = X\beta + e$$

where the errors are independent and distributed as N(0,$\sigma^2$). When $X$ has full rank, there is a unique $b$ that minimizes the sum of the squares of the errors, $\|y - X\beta\|_2$. This unique solution,

$$b = (X'X)^{-1}X'y,$$

has many desirable properties, such as unbiasedness and being the maximum likelihood estimator of $\beta$.

If $\mathbf{X} = [\underline{X}, \overline{X}]$ and $y = [\underline{y}, \overline{y}]$, then an interval least squares solution provides an enclosure of

$$\mathbf{B} = \mathbf{X'y} = \{X'y : X \in \mathbf{X}, y \in \mathbf{y}\}.$$

Interval least squares was researched by Gay (1988) with the conclusion (p. 203): "Interval leastsquares estimates can furnish rigorous and reasonably tight bounds on the effect of errors in the independent variables on forecasts and, as special cases, on parameter estimates."

### 1.6.4 Optimization

Optimization techniques are an essential tool in statistical computing. Some of the areas which require optimization methods include:

1. Maximum likelihood
2. Optimal experimental design
3. Projection pursuit
4. Nonlinear least squares
5. Minimum volume ellipsoids

Traditional optimization methods used for these problems include Steepest Descent, NewtonRaphson, DUD (Doesn't Use Derivatives), and Simulated Annealing. The more widely used methods are discussed in the books by Thisted (1988) and Kennedy and Gentle (1980). Simulated Annealing has been applied to optimal design problems in a paper by Bohachevsky et al. (1986), while DUD is discussed in a nonlinear leastsquares context by Ralston and Jennrich (1978). Simulated Annealing has perhaps the best chance of the methods listed here for avoiding being trapped in a local optimum and has proven effective for solving some problems, but requires parameters that can be difficult to tune so that a global optimum is located with any degree of certainty.

Interval analysis can also be used for global optimization. Hansen (1988) and Kearfott (1996a) are two monographs on this topic. A short description of the basic idea of interval global optimization is now given in terms of maximization. (Minimization is very similar). Interval analysis can be used to enclose the range of a function over a given domain. A noninterval optimization method can be used to quickly locate a local optimum $y^*$, which has interval enclosure $\mathbf{f}(y^*)$. The maximum value of the function $f$ then must be at least as large as the lower bound of the interval enclosure $\underline{\mathbf{f}}(y^*)$. Starting with an initial interval box $\mathbf{y}_0$, the region is bisected repeatedly into ever smaller boxes. If a box $\mathbf{y}_i$ satisfies $\mathbf{f}(\mathbf{y}_i) < \underline{\mathbf{f}}(y^*)$, then it is known that $\mathbf{y}_i$ does not contain a global maximum and can be discarded.

Wang (1994) uses interval global optimization algorithms on a MasPar machine to solve several problems from nonlinear regression, optimal design, and maximum likelihood for a moving average model.

## 1.7 Example Applications of Interval Analysis to Statistics

This section describes several ways (that have not previously been published) in which interval analysis can be utilized in statistics. These are simple examples (and so are not presented in great detail) that serve as demonstrations.

### 1.7.1 Experimental Design

Table 1.1 from Montgomery (1991) (page 151) illustrates an experimental design with missing data in the response variable. In this case there is one missing value. A traditional approach to analysis of this data is to minimize the error sum of squares with respect to $x_m$. As stated by Montgomery (1991),

$$SS_E = \sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij}^2 - \frac{1}{b}\sum_{i=1}^{a}\left(\sum_{j=1}^{b} y_{ij}\right)^2 - \frac{1}{1}\sum_{j=1}^{b}\left(\sum_{i=1}^{1} y_{ij}\right)^2 + \frac{1}{ab}\left(\sum_{i=1}^{a}\sum_{j=1}^{b} y_{ij}\right)^2$$

or in the case of one missing value,

$$SS_E = x_m^2 - \frac{1}{b}(y'_{i.} + x_m)^2 - \frac{1}{a}(y'_{.j} + x_m)^2 + \frac{1}{ab}(y'_{..} + x_m)^2 + R$$

Table 1.1: Randomized complete block design with one missing value

| | Block | | | |
|---|---|---|---|---|
| Type of Tip | 1 | 2 | 3 | 4 |
| 1 | -2 | -1 | 1 | 5 |
| 2 | -1 | 2 | $x_m$ | 4 |
| 3 | -3 | 1 | 0 | 2 |
| 4 | 2 | 1 | 5 | 7 |

where R includes terms not involving $x_m$ . Solving this yields

$$x_m = \frac{ay'_{i.} + by'_{.j} - y'..}{(a-1)(b-1)}$$

for the missing data. For the data presented here, the estimate of the missing data is $\hat{x}_m = 1.22$. The analysis of variance is performed with the estimated value and a reduction in the error degrees of freedom by one.

This is a useful approach for analysis of variance, but offers little help in the way of sensitivity analysis for the estimates of the parameters in the design. Designating the observed data by $Y$ and the appropriate design matrix by $X$, the model can be written $Y = X\beta$. Using the value of 1.22 as an estimate of the missing data, $\hat{\beta} = (X'X)^{-1}X'Y$ gives

$$\hat{\beta} = \begin{bmatrix} 7.11125 \\ -5.5 \\ -5.25 \\ -2.695 \\ -3 \\ -3.195 \\ -4.25 \end{bmatrix}$$

as the best linear unbiased estimator.

Suppose the missing value $x_m$ is replaced by an interval value $\mathbf{x}_m$. The choice of an estimate for $\mathbf{x}_m$ can be made in several ways. The experimenter may posses knowledge about the response (dependent) variable which provides obvious bounds. If, for example, the response variable is a proportion, then $[0, 1]$ must be the widest possible estimate of $\mathbf{x}_m$. Depending on which response value is missing, it may be possible to use another approach. For illustration purposes, $\mathbf{x}_m$ is here estimated by $[-3, 7]$, which encompasses the range of values observed in the response variable. Let $Y_I$ be interval data equal to the original data $Y$ with the exception that $Y_I$ uses $\mathbf{x}_m$ in place of the missing data $x_m$ .

Setting $\hat{\beta}_I = (X'X)^{-1}X'Y_I$ yields

$$\hat{\beta}_I = \begin{bmatrix} [6.75, 7.375] \\ [-5.5, -5.5] \\ [-5.25, -5.25] \\ [-3.75, -1.25] \\ [-3, -3] \\ [-4.25, -1.75] \\ [-4.25, -4.25] \end{bmatrix}$$

as an interval estimate of $\beta$.

It is now possible to see immediately and at a glance which values and to what extent the values of $\hat{\beta}_I$ are influenced by the interval estimate of $x_m$ .

### 1.7.2 Kolmogorov Smirnov Distribution

Tables for estimating the goodness of fit of empirical distributions were published by Smirnov (1948). The distribution used is

$$F(x) = 1 - 2\sum_{i=1}^{\infty}(-1)^{i-1}e^{-i^2x^2}. \tag{1.4}$$

Let $\mathbf{F}_k$ denote the interval extension of (1.4) truncated after $k$ terms. Since the series in (1.4) is alternating, $F(x) \in Hull(\mathbf{F}_k, \mathbf{F}_{k+1})$ for each $k$. The series is calculated until no change in $Hull(\mathbf{F}_k, \mathbf{F}_{k+1})$ is observed from one iteration of $k$ to the next. Examples of the selfvalidated values for the distribution function appear in tables 1.2 and 1.3. The tables validate the results published in Smirnov (1948) with the exceptions of some values in the earlier published work that appear to be typographical and/or rounding errors.

### 1.7.3 A Bivariate F Distribution

Let $X \sim \chi^2(2n), Y \sim \chi^2(2n)$, and $X_0 \sim \chi^2(2m)$ be independent random variables. The random variable $V = \frac{min(X,Y)/n}{X_0/m}$ is known as the smaller of bivariate correlated F variables, or studentized minimum Chi–square variable. Some references to this distribution are given by Hamdy et al. (1988).

The density of V is given by

$$h(v) = 2\sum_{i=0}^{n-1} \frac{\binom{n+i-1}{i}(n/m)^{i+n}v^{n+i-1}}{\beta(n+i,m)(1+2vn/m)^{n+m+i}} I(v > 0), \tag{1.5}$$

Hamdy et al. (1988) give an algorithm for finding $c$ in $1 - \alpha = \int_c^{\infty} h(v)dv$. The heart of the algorithm involves computation of

$$q(h) = \sum_{i=0}^{n-1}\sum_{j=0}^{m-1} \binom{n+i-1}{i}\binom{n+m+i-1}{j}\left(\frac{1}{2}\right)^{n+i-1} h^j(1-h)^{n+m+i-1-j} \tag{1.6}$$

Table 1.2: Guaranteed bounds for distribution points of Smirnov's distribution

| $x$ | $F(x)$ | $x$ | $F(x)$ |
|---|---|---|---|
| 0.4 | $0.00280767322270^{29}_{15}$ | 1.5 | $0.977782037383474^{9}_{8}$ |
| 0.5 | $0.03605475633512^{56}_{44}$ | 1.6 | $0.988047956760803^{5}_{4}$ |
| 0.6 | $0.135717220949396^{62}_{52}$ | 1.7 | $0.993822569365555^{9}_{8}$ |
| 0.7 | $0.28876480497031^{12}_{02}$ | 1.8 | $0.996932378652420^{3}_{2}$ |
| 0.8 | $0.45585758842580^{24}_{16}$ | 1.9 | $0.998536395162812^{7}_{6}$ |
| 0.9 | $0.607269292059345^{9}_{4}$ | 2.0 | $0.999329074744220^{4}_{3}$ |
| 1.0 | $0.730000328322645^{7}_{3}$ | 2.1 | $0.99970450327953^{7}_{069}$ |
| 1.1 | $0.822281807393590^{90}_{88}$ | 2.2 | $0.999874956992450^{4}_{3}$ |
| 1.2 | $0.887750333329275^{1}_{0}$ | 2.3 | $0.999949161306967^{6}_{5}$ |
| 1.3 | $0.931907778155233^{7}_{5}$ | 2.4 | $0.999980140991388^{3}_{2}$ |
| 1.4 | $0.960318120461885^{7}_{6}$ | 2.5 | $0.999992546693655^{9}_{8}$ |

Table 1.3: Guaranteed bounds for critical points of Smirnov's distribution

| $x$ | $F(x)$ |
|---|---|
| $1.2238478702170825^{5}_{1}$ | 0.90 |
| $1.3580986393225^{505}_{500}$ | 0.95 |
| $1.627623611518^{517}_{484}$ | 0.99 |

where $h = (1 + 2nc/m)^{-1}$ $0 \leq h \leq 1$. Since $q(h)$ is a rational function, its interval extension $\mathbf{q}(\mathbf{h})$ is immediately obtainable. For a given $\alpha$, the determination of $c$ proceeds via some suitable root-finding method. For the present work, derivative-free bracket-secant and bisection methods were used. Some percentile points of this distribution are tabulated below in table 1.4. This tabulation is a subset of the tables in Gupta and Sobel (1962). Table 1.4 verifies the results in Gupta and Sobel (1962) with the exception of a few values in the earlier work that appear to be rounding errors.

## 1.8 Internet Resources for Interval Analysis

The world wide web serves as an extremely valuable resource for researchers inter ested in interval computations. A pair of starting points for searching the WWW are given here.

Development of some interval software packages (including BIAS/PROFIL) is pro-

Table 1.4: Percentile points of $V$ with degrees of freedom $m = n$

| $m$ | $1 = \alpha = 0.75$ | $1 - \alpha = 0.90$ |
|---|---|---|
| 1 | $0.166666666666666^{9}_{5}$ | $0.0555555555555555^{69}_{47}$ |
| 2 | $0.31674887432700^{76}_{69}$ | $0.16295234436871^{81}_{76}$ |
| 3 | $0.40404581771562^{19}_{10}$ | $0.241694741845574^{8}_{3}$ |
| 4 | $0.46279003617582^{35}_{24}$ | $0.300165442768985^{3}_{44}$ |
| 5 | $0.50596585743400^{66}_{49}$ | $0.34570359411078^{26}_{16}$ |
| 6 | $0.53953488035766^{87}_{69}$ | $0.38253359804302^{29}_{18}$ |
| 7 | $0.56666603181305^{36}_{14}$ | $0.41318079377463^{40}_{25}$ |
| 8 | $0.58922356764353^{60}_{56}$ | $0.43924707520903^{33}_{18}$ |
| 9 | $0.60838773909400^{75}_{37}$ | $0.46180265697818^{07}_{784}$ |
| 10 | $0.62494835251274^{32}_{398}$ | $0.48159369356954^{73}_{51}$ |
| 11 | $0.63945734273308^{47}_{12}$ | $0.49915892547182^{61}_{41}$ |
| 12 | $0.6523140778575^{116}_{078}$ | $0.51489877713092^{55}_{30}$ |
| 13 | $0.66381591094938^{65}_{23}$ | $0.52911819748746^{76}_{41}$ |
| 14 | $0.67418959284749^{65}_{16}$ | $0.54205427500907^{35}_{04}$ |
| 15 | $0.68361157764423^{81}_{32}$ | $0.5538946419155^{329}_{296}$ |
| 16 | $0.69222159319099^{76}_{27}$ | $0.56479009787345^{44}_{06}$ |
| 17 | $0.7001319730934^{227}_{165}$ | $0.57486348227275^{75}_{33}$ |
| 18 | $0.70743423464199^{89}_{31}$ | $0.58421608660013^{74}_{29}$ |
| 19 | $0.7142038168150^{350}_{285}$ | $0.5929322731883^{256}_{11}$ |
| 20 | $0.720503558738^{701}_{694}$ | $0.6010830234646^{128}_{077}$ |
| 21 | $0.7263862971195^{708}_{639}$ | $0.60872855022104^{79}_{28}$ |
| 22 | $0.73189683545944^{79}_{11}$ | $0.61592033239380^{82}_{24}$ |
| 23 | $0.7370734575331^{603}_{525}$ | $0.6227026926850^{731}_{673}$ |
| 24 | $0.7419491050757^{183}_{04}$ | $0.6291140384091^{090}_{38}$ |
| 25 | $0.7465523045448^{913}_{829}$ | $0.6351878476747^{249}_{185}$ |

ceeding at Technische Universitat HamburgHarburg. See http://www.ti3.tuharburg.de/indexEnglisch.html for details.

The URL for the *Interval Computations* journal homepage is http://cs.utep.edu/intervalcomp/main.html. This site contains numerous links to bibliographies, software, homepages of interval computations centers and individuals, etc.

## 1.9 Dissertation Organization

The remainder of this dissertation is divided into several chapters, each of which applies methods of interval analysis to a separate problem in statistics. The following three chapters are written as manuscripts for submission to scientific journals. This arrangement means there may be some duplicity and some differences in notation between chapters, but also means that each chapter can be read independently of the others.

The chapter which follows this one addresses calculating critical points and tail probabilities for several bivariate Chi–square distributions. Since locating the critical point for a given tail probability of a distribution typically involves rootfinding, there is also development of an interval secant algorithm (with Illinois modification). Some tables of verified percentile points of the distributions are presented.

The third chapter takes a similar approach, but considers a class of bivariate F distributions. Series expansions are used in the calculations together with selfvalidated numerical quadrature rules.

The fourth chapter considers a different type of problem altogether. Several ways of using interval analysis together with the EM algorithm are considered. One of the ways in which these two have successfully been combined is to consider an enclosure of the gradient of the loglikelihood and eliminate portions of the parameter space where the gradient is not zero. The methodology for this technique is presented along with an algorithm for the procedure. The methodology is applied to several examples.

Conclusions are presented at the end of the dissertation.

# Chapter 2

# SELF-VALIDATED COMPUTATIONS FOR THE PROBABILITIES OF THE CENTRAL BIVARIATE CHI-SQUARE DISTRIBUTION


## Abstract

Self-validated computations using interval arithmetic produce results with a guaranteed error bound. This article presents methods for self-validated computation of probabilities and percentile points of the bivariate chi-square distribution. For the computation of critical points $(c_1, c_2)$ in $P(Y_1 \leq c_1, Y_2 \leq c_2) = 1 - \alpha$, the case $c_1 = c_2$ is considered. A combination of interval bracket-secant and bisection algorithms is developed for finding enclosures of the percentile points of the distribution.


## 2.1 Introduction

Several applications in statistical inference rely on the existence of a bivariate chi-square distribution. As mentioned in Jensen and Howe (1968), such areas include simultaneous inferences for variances, simultaneous tests in analysis of variance, simultaneous tests for goodness of fit, and the distribution of the larger of correlated Chi-square -variates. Gunst and Webster (1973) show how the bivariate chi-square distribution can be applied to the density function of a linear combination of independent chi-square random variables. Jensen and Jones (1969) further discuss simultaneous confidence intervals for variances while Tuprah and Woodall (1986) present a related application to bivariate dispersion quality control charts. For example, consider a manufacturing process which is characterized by two random variables, $X$ and $Y$. Suppose $\sigma_1$ and $\sigma_2$ are the respective process standard deviations of the quality characteristics. It is desirable to detect shifts

in the process standard deviations away from specified target values $\sigma_{1_0}$ and $\sigma_{2_0}$. If the two random variables are not independent, then the bivariate chi-square distribution can be used to construct bounds for determining when $(\sigma_1, \sigma_2)$ is significantly off target.

Computations on digital computers should not be undertaken without giving some thought to error analysis. Examples of erroneous results obtained through naive computations appear often enough in scientific literature to cause concern. The techniques of interval analysis pioneered by Moore (1966, 1979) can be used to provide guaranteed error bounds for the results of mathematical computations. Guaranteed error bounds, provided they are sufficiently tight, can be used, to assess the accuracy of tabled values or to evaluate the quality of results produced by scalar algorithms. Interval analysis has been successfully employed in statistical areas, e.g. Wang and Kennedy (1994b), but has not yet had wide exposure to the statistical community.

The goals of this paper are:
1. To present truncation error bounds for infinite series related to some bivariate chi-square distributions;
2. To apply these bounds using interval analysis;
3. To develop and apply intervalized secant-bracket methods for root-finding to the location of critical points; and
4. To compare results obtained with previously published results and promote the utility of interval analysis.

## 2.2 Interval Analysis

An interval $\mathbf{x}$ is defined to be a closed, bounded set of real numbers, $\mathbf{x} = [\underline{x}, \overline{x}]$. Throughout this paper, boldface is used to indicate intervals. To assist the reader in immediately grasping the accuracy of numerical values, a shortened form of interval notation which will sometimes be used to represent intervals is $2.3^{67}_{33} = [2.333, 2.367]$.

Let $\mathbf{x} = [\underline{x}, \overline{x}]$ and $\mathbf{y} = [\underline{y}, \overline{y}]$ be two intervals. The interval arithmetic operations are defined as

$$\mathbf{x} \circ \mathbf{y} = \{x \circ y : x \in \mathbf{x}, y \in \mathbf{y}\},$$

where $\circ \in \{+, -, \cdot, /\}$ and division is undefined for $0 \in \mathbf{y}$. Interval arithmetic operations can be expressed in closed form using real arithmetic operations. For example,

$$\mathbf{x} \cdot \mathbf{y} = [\min(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}), \max(\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y})].$$

Similar expressions exist for $+, -$, and $/$.

An *interval function* is an interval-valued function of one or more interval arguments. A function $\mathbf{f}(\mathbf{x}_1, \ldots \mathbf{x}_n)$ is said to be an *interval extension* of $f(x_1, \ldots, x_n)$ if

$$\mathbf{f}([x_1, x_1], \ldots, [x_n, x_n]) = f(x_1, \ldots, x_n)$$

for all $x_i, i = 1, \ldots, n$. An interval-valued function $\mathbf{f}$ is said to be *inclusion monotonic* if $\mathbf{f}(\mathbf{x}) \subset \mathbf{f}(\mathbf{y})$ whenever $\mathbf{x} \subset \mathbf{y}$. A fundamental theorem from interval analysis states that rational interval functions are inclusion monotonic.

In this paper, the *natural interval extension* of a real function will be used. This is an interval-valued function in which intervals and interval operations are substituted for scalars and scalar operations. The value of the interval extension of a function is dependent on the form of the real function. For example, let $f_1(x) = xx - 2x$ and $f_2(x) = x(x-2)$. Let $\mathbf{f}_1$ and $\mathbf{f}_2$ be the corresponding natural interval extensions and let $\mathbf{x} = [-1, 2]$. Then $\mathbf{f}_1(\mathbf{x}) = [-6, 6]$ and $\mathbf{f}_2(\mathbf{x}) = [-6, 3]$ which both contain $[-1, 3]$, the true range of $f$ over $\mathbf{x}$. This feature of interval computations to sometimes overestimate the range of a function is referred to as *interval dependency*. Attention must be given to the exact expression of a function to reduce the effect of interval dependency.

When implementing interval arithmetic calculations on computers, care must be taken to ensure that rounding errors do not invalidate the inclusion monotonicity of interval results. One way to achieve this is through the use of directed rounding modes in the floating-point calculations. When calculating the lower endpoint of an interval result, the floating-point processor is set to round all results *down*. For calculation of the upper endpoint of an interval result, all calculations are rounded *up*. Using the symbols $\bigtriangledown$ and $\bigtriangleup$ to denote downward and upward rounding respectively, the actual computer implementation of interval addition is $\mathbf{x} + \mathbf{y} = [\bigtriangledown(\underline{x} + \underline{y}), \bigtriangleup(\overline{x} + \overline{y})]$. Correct use of the rounding modes guarantees that the computed result contains the true answer.

## 2.3 Bivariate Chi-Square Distributions

Consider two random variables with variances $\sigma_1^2$ and $\sigma_2^2$. Let $s_1^2$ and $s_2^2$ be estimates of $\sigma_1^2$ and $\sigma_2^2$ such that $\nu_i s_i^2/\sigma_i^2$ follows a chi-square distribution with $\nu_i$ degrees of freedom. The joint distribution of $\nu_1 s_1^2/\sigma_1^2$ and $\nu_2 s_2^2/\sigma_2^2$ is referred to as a bivariate chi-square distribution. For example, suppose $\sigma_1$ and $\sigma_2$ are the process standard deviations for two characteristics that have a bivariate normal distribution. The sample variances $s_1^2$ and $s_2^2$ can be used along with a region determined by a bivariate chi-square distribution to simultaneously detect shifts in the process standard deviations away from specified target values.

Consideration will be given here to three cases of a bivariate chi-square distribution which are distinguished by degrees of freedom and the number of non-zero canonical correlations.

Case I. In the first case, let $\{(Z_{1i}, Z_{2i}), i = 1, \ldots, m\}$ be independent random variables, $Z_{ij} \sim N(0, 1)$ with (canonical) correlation between $Z_{1i}$ and $Z_{2i}$ of $\rho$. Then $Y_i = \sum_{j=1}^{m} Z_{ij}^2, i = 1, 2$ are chi-square random variables, each with $m$ degrees of freedom, and with $m$ non-zero canonical correlations $\rho$. The joint density of $Y_1$ and $Y_2$ is given by

$$f(y_1, y_2) = (1-\rho^2)^{m/2} \sum_{j=0}^{\infty} \frac{\Gamma(\frac{m}{2}+j)\rho^{2j}}{j!\Gamma(\frac{m}{2})} \times \frac{(y_1 y_2)^{(m/2)+j-1} \exp[-(y_1+y_2)/2(1-\rho^2)]}{[2^{(m/2)+j}\Gamma(\frac{m}{2}+j)(1-\rho^2)^{(m/2+j)/2}]^2} \tag{2.1}$$

and the distribution is given by

$$P[Y_1 \le d_1, Y_2 \le d_2] = (1-\rho^2)^{m/2} \times \sum_{j=0}^{\infty} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \rho^{2j} \gamma\left(\frac{m}{2}+i, d_1^*\right) \gamma\left(\frac{m}{2}+i, d_2^*\right), \tag{2.2}$$

where $\gamma(\alpha, d)$ is the incomplete gamma function,

$$\gamma(\alpha, d) = \int_0^d \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x} dx$$

and $d_j^* = d_j/(1-\rho_{12}^2)$. When the infinite series in (2.2) is truncated after $t+1$ terms, a bound on the truncation error $R_t$ given by Krishnaiah (1980a) is

$$R_t \le 1 - (1-\rho^2)^{m/2} \sum_{j=0}^{t} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \rho^{2j}. \tag{2.3}$$

For completeness, the derivation of this bound is given here. Let

$$k_j = (1-\rho^2)^{m/2} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \rho^{2j}.$$

Since $\gamma(\cdot,\cdot) \le 1$ and the $k_j$ are the density of a Negative Binomial distribution, the truncation error $R_t$ satisfies

$$R_t = \sum_{j=t+1}^{\infty} k_j \gamma\left(\frac{m}{2}+j, d_1^*\right) \gamma\left(\frac{m}{2}+j, d_2^*\right) \le \sum_{j=t+1}^{\infty} k_j = 1 - \sum_{j=0}^{t} k_j.$$

Let $P_t$ represent the result when the first $t+1$ terms of (2.2) are used, and let $\mathbf{p}_t$ and $\mathbf{r}_t$ represent the natural interval extensions of $P_t$ and $R_t$ respectively. Then

$$P[Y_1 \le d_1, Y_2 \le d_2] = P_t + R_t \in [\underline{p}_t, \overline{p}_t + \overline{r}_t]$$

for all $t$. The stopping value of $t$ used depends on machine and software precision. In practice, $\mathbf{r}_t$ is computed successively and iteration stops when $\mathbf{r}_{t-1} = \mathbf{r}_t$ or when the width of $\mathbf{r}_t$ is less than a specified tolerance.

Case II. Now consider the case when $Y_1$ and $Y_2$ follow chi-square distributions with $m$ and $n$ degrees of freedom respectively and have $m$ non-zero canonical correlations $\rho$. The joint distribution of $Y_1$ and $Y_2$ given by Gunst and Webster (1973) is

$$P[Y_1 \le d_1, Y_2 \le d_2,] = (1-\rho^2)^{(m+n)/2} \times \sum_{j=0}^{\infty}\sum_{k=0}^{\infty} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \frac{\Gamma(\frac{n}{2}+k)}{k!\Gamma(\frac{n}{2})} \rho^{2(j+k)} \gamma\left(\frac{m}{2}+j, d_1^*\right) \gamma\left(\frac{n}{2}+k, d_2^*\right). \tag{2.4}$$

Since a bound for the truncation error has not previously been published, one is given here. Because the method of derivation is completely analogous to the previous case, only the result is stated:

$$R_{t_1,t_2} \le 1 - (1-\rho^2)^{m/2+n/2} \sum_{j=0}^{t_1}\sum_{k=0}^{t_2} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \frac{\Gamma(\frac{n}{2}+k)}{k!\Gamma(\frac{n}{2})} \rho^{2(j+k)}. \tag{2.5}$$

Let $\mathbf{p}_{t_1,t_2}$ be the result obtained when the first $(t_1+1, t_2+1)$ terms of the natural interval extension of (2.4) are used, and let $\mathbf{r}_{t_1,t_2}$ be the result obtained from the natural interval extension of (2.5). Then

$$P[Y_1 \leq d_1, Y_2 \leq d_2] \in [\underline{p}_{t_1,t_2}, \overline{p}_{t_1,t_2} + \overline{r}_{t_1,t_2}]$$

for all pairs $(t_1, t_2)$.

Case III. In the final case, $Y_1$ and $Y_2$ have chi-square distributions with $m+n$ and $m+p$ degrees of freedom, respectively, and there are $m$ non-zero canonical correlations. The joint distribution given by Gunst and Webster (1973) is:

$$P[Y_1 \leq d_1, Y_2 \leq d_2] = (1-\rho^2)^{(m+n+p)/2} \sum_{j=0}^{\infty}\sum_{k=0}^{\infty}\sum_{l=0}^{\infty} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})} \times$$
$$\frac{\Gamma(\frac{n}{2}+k)}{k!\Gamma(\frac{n}{2})}\frac{\Gamma(\frac{p}{2}+l)}{l!\Gamma(\frac{p}{2})}\rho^{2(j+k+l)}\gamma\left(\frac{m}{2}+\frac{n}{2}+k+j, d_1^*\right)\gamma\left(\frac{m}{2}+\frac{p}{2}+j+l, d_2^*\right). \quad (2.6)$$

The bound on the truncation error is again derived in a manner completely analogous to the first case:

$$R_t \leq 1-(1-\rho^2)^{m/2+n/2+p/2} \times$$
$$\sum_{j=0}^{t_1}\sum_{k=0}^{t_2}\sum_{l=0}^{t_3} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})}\frac{\Gamma(\frac{n}{2}+k)}{k!\Gamma(\frac{n}{2})}\frac{\Gamma(\frac{p}{2}+l)}{l!\Gamma(\frac{p}{2})}\rho^{2(j+k+l)}. \quad (2.7)$$

Similar to before,

$$P[Y_1 \leq d_1, Y_2 \leq d_2] \in [\underline{p}_{t_1,t_2,t_3}, \overline{p}_{t_1,t_2,t_3} + \overline{r}_{t_1,t_2,t_3}]$$

for all triples $(t_1, t_2, t_3)$ and the choice of $(t_1, t_2, t_3)$ is determined by machine/software limitations or a tolerance level.

## 2.4 Interval Bracket-Secant and Bisection Root-Finding

For a distribution function $F(x)$, the $100p^{th}$ percentile $x_p$ is the solution of the equation $F(x_p) - p = 0$. For the bivariate chi-square distributions considered above, finding percentiles will involve solving

$$P(Y_1 \leq x_p, Y_2 \leq x_p) - p = 0,$$

which motivates a short discussion of interval root-finding techniques. Intervalized Newton methods for finding zeros of functions exist and could be used. However, interval Newton methods require an interval extension of both the function of interest and its derivative, which in this case are the distribution and density functions. Using a derivative-free search algorithm eliminates the need to obtain an enclosure of the density function. The technique of automatic differentiation presented in Moore (1979) could be used to obtain an enclosure of the derivative of a function, but only at the cost of

(sometimes considerable) loss of precision due to dependency and additional computing time.

The algorithm ZERO used here begins with an intervalized secant-bracket method using the Illinois modification. See Thisted (1988) for a complete explanation of this algorithm in the scalar case. Consider the general real equation $F(x) = 0$ as depicted by the curved line in figure 2.1. Let $\mathbf{F}(x) \equiv \mathbf{F}([x, x])$ (i.e. a scalar argument $x$ is interpreted by the function as the interval $[x, x]$) be the interval extension of $F$. The algorithm begins with the user-specified interval $[x_{i-1}, x_i]$, which bounds the zero of the function. Using an intervalized secant method, the algorithm finds the point $x_{i+1}$ and decides whether $[x_{i-1}, x_{i+1}]$ or $[x_{i+1}, x_i]$ now contains the zero of $F$. At some iteration $i + 1$ of the secant portion of the algorithm, $0 \in \mathbf{F}(x_{i+1})$ and it is not then known whether the zero of the function is to the left or the right of $x_{i+1}$. Figure 2.1 shows this condition.

The secant algorithm is successful in narrowing the enclosure of the zero of the function for the initial iterations, but may stop while the interval is wider than desired.

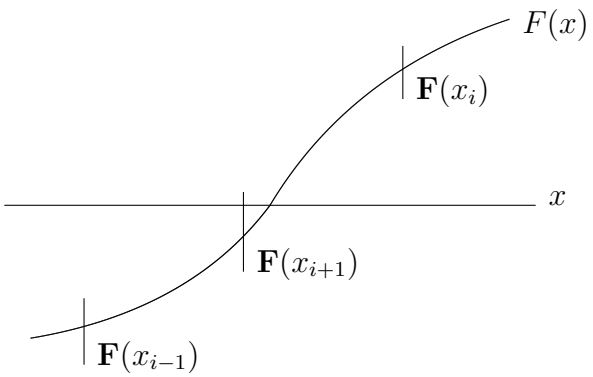


Figure 2.1: Termination of the bracket-secant portion of the algorithm. The curved line represents the real function and vertical line segments denote interval enclosures of the function.

After the bracket-secant portion of the algorithm terminates, let $x_L$ and $x_U$ represent the last two iterates $x_{i-1}$ and $x_i$. A bisection algorithm is called twice, once each on the lower $[x_L, x_{i+1}]$ and upper $[x_{i+1}, x_U]$ intervals, to tighten the enclosure of the zero as much as possible, i.e. until (but not including) $0 \in \mathbf{F}(x_L), 0 \in \mathbf{F}(x_U)$. Figure 2.2 illustrates conditions at the termination of algorithm ZERO.

ALGORITHM Zero(MACHINE_REAL $x_0$, $x_1$; INTERVAL_FUNCTION $\mathbf{F}$)
REM Bound the root of an increasing function $F$. Return $[l, u]$ to user.
REM Check that $F(x_0) < 0 < F(x_1)$ or $F(x_1) < 0 < F(x_0)$
REM Bold letters denote intervals
$l := x_0$
$u := x_1$
$\mathbf{FL} := \mathbf{F}([l, l])$

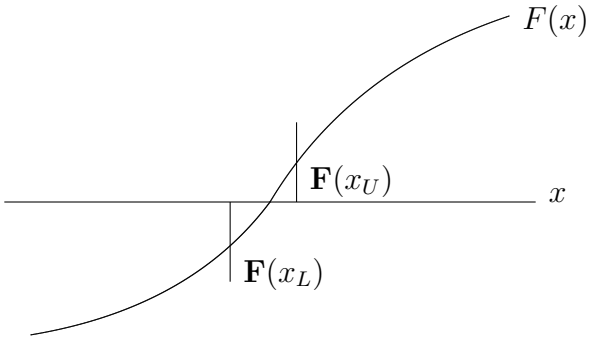


Figure 2.2: Termination of the bisection portion of the algorithm. The interval $[x_L, x_R]$ bounds the root of the function $F(x) = 0$.

$\mathbf{FU} := \mathbf{F}([u, u])$
REPEAT
    $\mathbf{X}_c := u - (u - l)/(1 - \mathbf{FL}/\mathbf{FU})$  REM Find the secant intercept
    $x_{i+1} := (\underline{\mathbf{X}}_c + \overline{\mathbf{X}}_c)/2$  REM Use the midpoint for the next iterate
    $\mathbf{F}_{i+1} := \mathbf{F}([x_{i+1}, x_{i+1}])$
    IF $\underline{\mathbf{F}}_{i+1} > 0$ THEN
        REM The zero is between $l$ and $x_{i+1}$
        $u := x_{i+1}$
        $\mathbf{FU} := \mathbf{F}_{i+1}$
        IF $\underline{\mathbf{F}}_i > 0$ THEN $\mathbf{FL} := \mathbf{FL}/2$  REM Illinois modification
    ELSE IF $\overline{\mathbf{F}}_{i+1} < 0$ THEN
        REM The zero is between $x_{i+1}$ and $u$
        $l := x_{i+1}$
        $\mathbf{FL} := \mathbf{F}_{i+1}$
        IF $\underline{\mathbf{F}}_i < 0$ THEN $\mathbf{FU} := \mathbf{FU}/2$  REM Illinois modification
    ELSE
        REM Cannot determine if the zero is to the left or right of $x_{i+1}$
        $Done :=$ TRUE
    IF $0 \in (\mathbf{FU} - \mathbf{FL})$ THEN
        REM In the next iteration, the denominator would contain zero
        $Done :=$ TRUE
    $\mathbf{F}_i := \mathbf{F}_{i+1}$
UNTIL ($Done =$ TRUE)
REM Use bisection to tighten the upper endpoint
$q := l$
REPEAT
    $prevq := q$
    $prevu := u$

$x := (q + u)/2$
$\mathbf{FX} := \mathbf{F}([x, x])$
IF $\underline{\mathbf{FX}} > 0$ THEN $u := x$
ELSE IF $\overline{\mathbf{FX}} < 0$ THEN $q := l := x$
ELSE $q := x$
UNTIL ( $(q = prevq)$ AND $(u = prevu)$ )
REM Use bisection to tighten the lower endpoint
$q := u$
REPEAT
$prevq := q$
$prevl := l$
$x := (l + q)/2$
$\mathbf{FX} := \mathbf{F}([x, x])$
IF $\underline{\mathbf{FX}} > 0$ THEN $u := x$
ELSE IF $\overline{\mathbf{FX}} < 0$ THEN $l := x$
ELSE $q := x$
UNTIL ( $(q = prevq)$ AND $(l = prevl)$ )
RETURN $[l, u]$

## 2.5 Numerical Results

Computer languages supporting overloaded operators are ideally suited to the implementation of interval arithmetic routines. For this reason, calculations in this paper made use of the BIAS/PROFIL (Basic Interval Arithmetic Subroutines/ Programmer's Optimized Fast Interval Library) C++ package developed by Knüppel (1993a,b) and were programmed on DEC 5000 and DEC Alpha workstations. Computing times for a single critical point varied from a few seconds in Case I to a few minutes in Case III. Routines to compute an enclosure of the Incomplete Gamma function were drawn from work by Wang and Kennedy (1994b) and from source code by Gessner (1992).

Since the expressions (2.2), (2.4), and (2.6) depend on $\rho$ only through $\rho^2$, tables need only include nonnegative values of $\rho$. For the first case, tables 2.1 and 2.2 illustrate critical points $c$ for $P(Y_1 \leq c, Y_2 \leq c) = 1 - \alpha$ where $Y_1 \sim \chi^2(m), Y_2 \sim \chi^2(m)$.

For the second case, $Y_1 \sim \chi^2(m), Y_2 \sim \chi^2(m + n), n > 0$ and there are $m$ nonzero canonical correlations. Examples of critical values $c$ for $P(Y_1 \leq c, Y_2 \leq c) = 1 - \alpha$ are given in table 2.3.

In the third case, $Y_1 \sim \chi^2(m + n), Y_2 \sim \chi^2(m + p), n > 0, p > 0$ and there are $m$ nonzero canonical correlations. Table 2.4 illustrates critical values for this case.

The illustrative tables presented here are limited to $c_1 = c_2$ when computing the values of $(c_1, c_2)$ in $P(Y_1 \leq c_1, Y_2 \leq c_2) = 1 - \alpha$. Other schemes are possible, fixing $c_1$ for example and calculating $c_2$, or by adding a constraint. Distinct values of $c_1$ and $c_2$ can be entered when $1 - \alpha$ is the quantity to be computed.

Tables for the approximate critical points of the bivariate chi-square distribution have previously appeared in Gunst (1973), Gunst and Webster (1973), and Krishnaiah (1980a).

Table 2.1: Upper 0.05 percentile points of the bivariate chi-square distribution: Case I

| $\rho$ | m = 2 | m = 12 | m = 40 |
|---|---|---|---|
| 0.1 | $7.348735242636^{94}_{62}$ | $23.291675614644^{9}_{3}$ | $59.27375898559^{93}_{83}$ |
| 0.2 | $7.337736654468^{73}_{52}$ | $23.279893907495^{5}_{0}$ | $59.25865809054^{33}_{18}$ |
| 0.3 | $7.318116097295^{33}_{00}$ | $23.257752618706^{8}_{3}$ | $59.22980920643^{15}_{99}$ |
| 0.4 | $7.28777721964^{231}_{194}$ | $23.22124105410^{20}_{13}$ | $59.18114430513^{16}_{90}$ |
| 0.5 | $7.243389878426^{35}_{03}$ | $23.16405309924^{80}_{72}$ | $59.10283493307^{35}_{11}$ |
| 0.6 | $7.179739084402^{47}_{00}$ | $23.07634122520^{19}_{06}$ | $58.97915719228^{26}_{82}$ |
| 0.7 | $7.088168635581^{74}_{21}$ | $22.94173918779^{26}_{09}$ | $58.78357869885^{65}_{04}$ |
| 0.8 | $6.95217862545^{563}_{462}$ | $22.72905146810^{30}_{00}$ | $58.46518690308^{30}_{10}$ |
| 0.9 | $6.73002568707^{699}_{492}$ | $22.35957467992^{63}_{95}$ | $57.89540621112^{59}_{42}$ |

Jensen and Howe (1968) determine the probability content over certain square and rectangular regions for which the marginal probabilities are specified. Dutt and Soms (1976) describe an alternative method for calculating multivariate chi-square probabilities using integral representations. The very narrow interval enclosures obtained using interval analysis can be useful for checking the accuracy of previously tabulated values. Indeed, Table I of Gunst (1973), *Upper 100α% Critical Points*, is discovered to be widely accurate to only two decimal places (three are given), and the values in Table II, *Lower 5% Critical Points*, appear to be wholly incorrect. As noted in the text which accompanies the two tables of Gunst (1973), the critical points for $\rho = 0.10$ are nearly identical to the univariate critical points for $\alpha^* = 1 - (1-\alpha)^{(1/2)}$. That this observation holds for values obtained in the current research and does not hold for the previously published values supports the correctness of the current research. The values in Table 6 of Krishnaiah (1980a), *Percentage Points of the bivariate chi-square distribution*, should be multiplied by two to obtain the correct values, and are then accurate to only two decimal places. If, for example, the incorrect values had been used to construct a confidence ellipsoid for the distribution of the variances of two random variables, the ellipsoid would be far too small to achieve the desired confidence.

The tables which appear in this paper are included to demonstrate the very high precision and the guarantee of accuracy which are obtained via the use of interval arithmetic. Except in certain cases, use of more digits in the table than significant digits in available data should be discouraged. The great value of such high-quality numbers is more likely in knowing that the second digit of a percentile point is guaranteed to be accurate than in knowing what the eighth digit is.

Table 2.2: Upper 0.01 percentile points of the bivariate chi-square distribution: Case I

| $\rho$ | m = 2 | m = 12 | m = 40 |
|---|---|---|---|
| 0.1 | $10.5901634351^{801}_{788}$ | $28.290923051363^{39}_{16}$ | $66.7537579396^{68}_{31}$ |
| 0.2 | $10.585327860^{63}_{42}$ | $28.2868608929^{24}_{00}$ | $66.749113744^{81}_{21}$ |
| 0.3 | $10.57564269^{497}_{481}$ | $28.2782188566^{28}_{04}$ | $66.739131842^{3039}_{2970}$ |
| 0.4 | $10.55852476^{2011}_{1993}$ | $28.261729392^{55}_{21}$ | $66.7196589^{1106}_{0966}$ |
| 0.5 | $10.5299369013^{86}_{70}$ | $28.231769051^{526}_{490}$ | $66.683093898^{385}_{282}$ |
| 0.6 | $10.4835863790^{91}_{68}$ | $28.17882820433^{81}_{21}$ | $66.615860015^{257}_{067}$ |
| 0.7 | $10.4089856476^{64}_{38}$ | $28.086135579^{724}_{655}$ | $66.492996288^{877}_{619}$ |
| 0.8 | $10.2860268072^{80}_{30}$ | $27.920471007^{549}_{418}$ | $66.26371335^{4451}_{3945}$ |
| 0.9 | $10.0635917^{2077}_{1974}$ | $27.596098526^{674}_{378}$ | $65.795086198^{9439}_{8534}$ |

## 2.6 Conclusions

Even in algorithms where theoretical error analysis bounds error terms, computer arithmetic rounding errors and cancellation can have catastrophic effects. In this paper, interval analysis techniques have been successfully applied to bivariate chi-square distributions to produce tables of critical values with guaranteed error bounds. The results obtained revealed inaccuracies and limitations of earlier published tables. Improved tables are useful in and of themselves, but it is expected that this research will be of more value in the long run by promoting the use of automatically verified computations and by providing new methodologies to achieve that end. For example, the techniques developed here have been modified slightly by the authors to produce tables of self-validated critical points of a bivariate $F$ distribution. Interval analysis does have limitations in applications, but further research in applying interval techniques to statistical sciences should produce fruitful results.

Persons interested in obtaining a copy of the software described here may contact either of the authors.

Table 2.3: Upper 0.05 percentile points of the bivariate chi-square distribution: Case II

| $m$ | $m+n$ | $\rho = 0.2$ | $\rho = 0.4$ |
|---|---|---|---|
| 8 | 10 | $19.25562949840^{83}_{57}$ | $19.2145575272^{118}_{051}$ |
| 8 | 12 | $21.43987191019^{31}_{01}$ | $21.4161617406^{314}_{243}$ |
| 8 | 14 | $23.8527709314^{125}_{093}$ | $23.8412410690^{155}_{085}$ |
| 8 | 16 | $26.3613220302^{700}_{665}$ | $26.356264614^{8029}_{7941}$ |
| 8 | 18 | $28.893827805 50^{94}_{57}$ | $28.8917353286^{769}_{655}$ |
| $m$ | $m+n$ | $\rho = 0.6$ | $\rho = 0.8$ |
| 8 | 10 | $19.1173225955^{465}_{268}$ | $18.90189596^{30343}_{29502}$ |
| 8 | 12 | $21.3604107620^{626}_{459}$ | $21.243609643^{9166}_{8386}$ |
| 8 | 14 | $23.8140331030^{887}_{653}$ | $23.759702671^{6462}_{5233}$ |
| 8 | 16 | $26.3442149104^{737}_{417}$ | $26.32112960 0^{3403}_{2334}$ |
| 8 | 18 | $28.886692666 1^{733}_{478}$ | $28.877401302^{6245}_{4641}$ |

Table 2.4: Upper 0.05 percentile points of the bivariate chi-square distribution: Case III

| $m$ | $n$ | $p$ | $\rho = 0.4$ | $\rho = 0.6$ |
|---|---|---|---|---|
| 7 | 1 | 11 | $28.892120984^{7236}_{6486}$ | $28.887920316^{7461}_{3450}$ |
| 6 | 2 | 12 | $28.8924903043^{929}_{189}$ | $28.88907530^{42914}_{39005}$ |
| 5 | 3 | 13 | $28.892843220^{4235}_{3505}$ | $28.890155051^{4641}_{1105}$ |
| 4 | 4 | 14 | $28.893179676^{8563}_{7850}$ | $28.891157355^{6196}_{1279}$ |
| 3 | 5 | 15 | $28.89349628^{9162}_{8497}$ | $28.89208036^{93269}_{88632}$ |
| 2 | 6 | 16 | $28.893803042^{8769}_{7959}$ | $28.892922583^{3886}_{0023}$ |
| 1 | 7 | 17 | $28.894089895^{9047}_{8054}$ | $28.89368281^{14038}_{09926}$ |

# Chapter 3

# SELF-VALIDATED CRITICAL POINTS OF A BIVARIATE F DISTRIBUTION


## Abstract

Self-validated computations using interval arithmetic produce results with a guaranteed error bound. This article presents methods for self-validated computation of probabilities and percentile points of a bivariate $F$ distribution. For the computation of critical points $(c_1, c_2)$ in $P(Y_1 \leq c_1, Y_2 \leq c_2) = 1 - \alpha$, the case $c_1 = c_2$ is considered. A combination of interval bracket-secant and bisection algorithms is utilized for finding enclosures of the percentile points of the distribution.


## 3.1 Introduction

A multivariate $F$ distribution, though not common, has useful applications in statistics. Schuurmann et al. (1975) point out its use in hypothesis testing under fixed-effects models, in certain two-way classification models, and in simultaneous testing of no treatment and block effects in symmetrical balanced incomplete block designs. Tables for percentage points of a multivariate $F$ distribution have appeared most recently in Krishnaiah (1980b).

Frequently implicit in the calculation of tables for critical points of statistical distributions is the assumption that numerical error does not invalidate the results. By using the techniques of interval analysis, it is possible to produce tables which are guaranteed to be free of rounding and certain other kinds of numerical error. This paper seeks to develop the techniques of interval analysis to a statistical application and promote the further utility of interval analysis to the statistical community.

## 3.2 Interval Analysis

Interval analysis first saw fruitful development in the 1960s, beginning with work published by Moore (1966). Since that time, interval analysis has seen extensive research a variety of applications, but has not seen wide exposure in statistical areas. This paper continues the development of interval analysis applications to statistical distributions as in Wang and Kennedy (1994b) and Wright and Kennedy (1999).

An interval $\mathbf{x}$ is a closed, real interval $[\underline{x}, \overline{x}]$, where the underscore and overscore are used to denote the lower and upper endpoints of the interval. In this paper, boldface is used to distinguish intervals from real numbers. Fundamental arithmetic operators can be defined for intervals. Let $\mathbf{x}$ and $\mathbf{y}$ be intervals. For $* \in \{+, -, \times, /\}$, define $\mathbf{x} * \mathbf{y} = \{x * y : x \in \mathbf{x}, y \in \mathbf{x}\}$, with division defined only when $0 \notin \mathbf{y}$. Closed-form expressions for the results of these operators exist, providing for easy computation of results, e.g.

$$\mathbf{x} \times \mathbf{y} = [\min\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\}, \max\{\underline{x}\underline{y}, \underline{x}\overline{y}, \overline{x}\underline{y}, \overline{x}\overline{y}\}].$$

An *interval function* is understood to be a function with interval arguments and an interval result. An interval function $\mathbf{f}$ is said to be *inclusion monotonic* if $\mathbf{x} \subset \mathbf{y}$ implies $\mathbf{f}(\mathbf{x}) \subset \mathbf{f}(\mathbf{y})$. A fundamental theorem from interval analysis states that rational interval functions are inclusion monotonic. When a rational interval function is evaluated on a computer, care must be taken to preserve the interval monotonicity of the function. This can be achieved by controlling the rounding mode of the CPU's floating-point processor. Processors which are compliant with the IEEE floating-point specifications (ANSI/IEEE (1987)) can be set to round down or round up, among other modes. To maintain inclusion monotonicity, the results of a lower endpoint computation are always rounded down and the results of an upper endpoint computation are always rounded up. For example, on a hypothetical three-digit computer and using directed triangles to indicate the appropriate rounding, the real fraction 1/6 is computed as $[1,1]/[6,6] = [\bigtriangledown(1/6), \bigtriangleup(1/6)] = [.166, .167]$. When intervals with many decimal digits are displayed, an easily-understood representation of intervals is, for example, $[.16^{7}_{6}]$. Most rounding-mode control can be made transparent to the programmer with the aid of appropriate software packages, such as the C++ libraries BIAS and PROFIL developed by Knüppel (1993a,b). With the definition of interval data types and overloaded operators for interval types, programming interval computations can be as simple as z := x + y.

## 3.3 Bivariate F Distribution

The multivariate $F$ distribution considered by Schuurmann et al. (1975) and Krishnaiah (1980b) is reconsidered here. Let $S = (s_{ij})$ be a Wishart random matrix with $m$ degrees of freedom and $E(S) = m\Sigma = m(\sigma_{ij})$. The joint distribution of $s_{11}, \ldots, s_{pp}$, the diagonal elements of $S$, is a multivariate $\chi^2$ distribution with $m$ degrees of freedom. The matrix $\Sigma$ is the covariance matrix of the underlying multivariate normal random variable. Let $F_i = \frac{s_{ii}\sigma^2/m}{s^2\sigma_{ii}/n}, i = 1, \ldots, p$ where $s^2/\sigma^2$ is independently distributed as a $\chi^2$ random variable

with $n$ degrees of freedom. Then the joint distribution of $F_1, \ldots, F_p$ is a multivariate $F$ distribution with $(m, n)$ degrees of freedom. When $p = 2$, $\rho = \sigma_{12}/(\sigma_{11}\sigma_{22})^{1/2}$ is the correlation between standard normal random variables that underlie the bivariate $\chi^2$ distribution. The bivariate distribution of $F_1$ and $F_2$, first introduced by Krishnaiah (1965), is given by

$$f(x_1, x_2) = \frac{n^{n/2}(1-\rho^2)^{(m+n)/2}}{\Gamma(\frac{m}{2})\Gamma(\frac{n}{2})} \sum_{i=0}^{\infty} \frac{\rho^{2i}\Gamma(m+\frac{n}{2}+2i)m^{m+2i}(x_1x_2)^{m/2+i-1}}{i!\Gamma(\frac{m}{2}+i)[n(1-\rho^2)+m(x_1+x_2)]^{m+n/2+2i}} \tag{3.1}$$

The distribution function can be expressed as

$$P(F_1 \le d_1, F_2 \le d_2) = (1-\rho^2)^{m/2} \sum_{j=0}^{\infty} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})}\rho^{2j}B_j \tag{3.2}$$

where

$$B_j = \int_0^{\infty} \frac{e^{-z/2}z^{n/2-1}}{2^{n/2}\Gamma(n/2)} I_{1j}I_{2j}dz$$

and

$$I_{kj} = \frac{1}{2^{m/2+j}\Gamma(m/2+j)} \int_0^{\frac{d_k m z}{2n(1-\rho^2)}} e^{-u/2}u^{m/2+j-1}du = \Gamma\left(\frac{m}{2}+j, \frac{d_k m z}{2n(1-\rho^2)}\right) \tag{3.3}$$

where $\Gamma(\alpha, d)$ is the incomplete gamma function,

$$\Gamma(\alpha, d) = \int_0^d \frac{1}{\Gamma(\alpha)} x^{\alpha-1}e^{-x}dx.$$

## 3.4 Computation of Tables

Krishnaiah (1980b) actually gives expressions for probability integrals over arbitrary rectangular regions, $P(c_1 \le F_1 \le d_1, c_2 \le F_2 \le d_2)$, but in all published tables of critical points, $c_1 = c_2 = 0$. The challenging aspect of computing an interval enclosure of a critical point for the bivariate $F$ distribution is to find an appropriate rational interval function which gives reasonably tight bounds for the enclosure.

When the infinite series in (3.2) is truncated after $t+1$ terms, a bound on the truncation error $R_t$ given by Schuurmann et al. (1975) is:

$$R_t \le 1 - (1-\rho^2)^{m/2} \sum_{j=0}^{t} \frac{\Gamma(\frac{m}{2}+j)}{j!\Gamma(\frac{m}{2})}\rho^{2j}. \tag{3.4}$$

Let $P_t$ denote the series in (3.2) truncated after $t+1$ terms. Also, let $\mathbf{p}_t$ and $\mathbf{r}_t$ denote interval enclosures of $P_t$ and $R_t$ respectively. Then

$$P[Y_1 \le d_1, Y_2 \le d_2] = P_t + R_t \in [\underline{p}_t, \overline{p}_t + \overline{r}_t]$$

for all $t$. The stopping value of $t$ which is used will depend on machine and/or software precision. In practice, $\mathbf{r}_t$ is computed successively and iteration stops when $\mathbf{r}_{t-1} = \mathbf{r}_t$.

Termination could also be specified to occur when the width of $\mathbf{r}_t$ is less than a specified tolerance.

A guaranteed enclosure of the incomplete gamma function is available by using methods found in Wang and Kennedy (1994b). To compute an enclosure for $B_j$ via a rational interval function, ideas similar to those of Amos and Bulgren (1972) are used. The integral is split into three pieces

$$\int_0^\infty \cdot\, dz = \int_0^{\varepsilon_1} \cdot\, dz + \int_{\varepsilon_1}^{\varepsilon_2} \cdot\, dz + \int_{\varepsilon_2}^\infty \cdot\, dz \tag{3.5}$$

The left and right tails of $B_j$ can be bound in the following ways:

$$0 \le \int_0^{\varepsilon_1} \cdot\, dz \le \varepsilon_1$$

$$0 \le \int_{\varepsilon_2}^\infty \cdot\, dz \le 1 - \Gamma\left(\frac{n}{2}, \frac{\varepsilon_2}{2}\right)$$

The middle integrand in the right-hand side of (3.5) covers a finite domain, over which the second derivative exists, and is computed via the the use of first-order Newton-Cotes quadrature. Some references to numerical quadrature with automatic result verification appear in Kelch (1993). The general form of Newton-Cotes quadrature is

$$\begin{aligned}\int_a^b f(x)dx = h\Big(&\frac{1}{2}f(a) + f(a+h) + f(a+2h) + \\ &\cdots + f(a+(m-1)h) + \frac{1}{2}f(a+hm)\Big) + E\end{aligned} \tag{3.6}$$

where $h = (b-a)/m$ and the error has the form

$$E = -\frac{(b-a)^3}{12m^2} f''(\xi)$$

for some $\xi \in (a, b)$. The interval extension of $E$ involves the computation of $\mathbf{f}''([a, b])$. If $f''$ is a rational function, as in this problem, then the interval extension $\mathbf{f}''$ is inclusion monotonic and the width of $\mathbf{f}''([a, b])$ is likely to be greater than $\mathbf{f}''([a', b'])$ for $[a', b'] \subset [a, b]$. Minimizing the width of the error term is one of the steps in achieving highly accurate final results. For this reason, the middle integral in the right-hand side of (3.5) is actually computed as the sum of a series of subintegrals, each of which is evaluated by numerical quadrature.

$$\int_{\epsilon_1}^{\epsilon_2} \cdot\, dz = \int_{\epsilon_1}^{\epsilon_1+\nu} \cdot\, dz + \int_{\epsilon_1+\nu}^{\epsilon_1+2\nu} \cdot\, dz + \cdots + \int_{\epsilon_1+(k-1)\nu}^{\epsilon_1+k\nu} \cdot\, dz + \int_{\epsilon_1+k\nu}^{\epsilon_2} \cdot\, dz \tag{3.7}$$

Tuning the parameters $(\epsilon_1, \epsilon_2, k, \nu, h, m)$ of the method used here is not an immediately straightforward matter. Generally speaking, increasing the number of quadrature points will increase the accuracy (i.e. narrowness) of the final answer. A limit is reached, however, when increasing the number of quadrature points becomes counter-productive. This happens because each interval function evaluation at a quadrature point results in

a slight amount of overestimation and underestimation of the true value. Increasing $m$ results in a narrower enclosure of the error $E$ for each integrand, but eventually this gain is nullified by the sum of the overestimated and underestimated function values. A similar phenomenon occurs in deciding how many subintegrals to use in evaluating $\int_{\epsilon_1}^{\epsilon_2} \cdot \, dz$.

It is common in interval analysis to use so-called automatic derivatives for evaluation of the derivatives of a function. See Moore (1979) for an overview. In this research, computed enclosures of $f''$ were found to be much narrower when a hand-derived expression for $f''$ was coded into the software. The resulting expression involves evaluation of $x^{m/2-2}$ over an interval with a lower endpoint of 0. This limits the degrees of freedom to $m \geq 5$.

Tables of critical values $d$ were computed for

$$P(F_1 \leq d, F_2 \leq d) = \alpha. \tag{3.8}$$

Critical values for non-rectangular regions could easily be computed uniquely by specifying an additional constraint. Solving the integral equation (3.8) for $d$ requires the use of an iterative algorithm for finding roots, e.g. Newton-Raphson. This paper uses an intervalized bracket-secant algorithm, switching to an intervalized bisection algorithm to further narrow the enclosure of the critical point after the bracket-secant algorithm terminates. Complete details of the method appear in Wright and Kennedy (1999).

## 3.5 Conclusions

The computation of self-validated critical values for this bivariate $F$ distribution is very time-consuming. The probability content of a rectangular region can be computed in a few minutes, but this is likely to be prohibitively costly for implementation of real-time computation of critical values. Since determination of critical points involves finding the roots of an equation, each entry in the table required several hours to compute on a DEC 5000 workstation. (Less time would be required for wider enclosures.) For this distribution, the real utility of interval analysis is the verification and guarantee of accuracy of previously published tables. The tables published in Schuurmann et al. (1975) and Krishnaiah (1980b) are generally quite accurate, but do have slight errors in the last (hundredths) digit which are likely due to rounding, exactly the kind of error which interval analysis can eliminate. Tables 3.1 and 3.2 are examples of the accuracy achieved by the software developed for this research.

Table 3.1: Upper 0.05 percentile points of the bivariate F distribution

| | $n = 10$ | | | |
|---|---|---|---|---|
| $m$ | $\rho = 0.1$ | $\rho = 0.3$ | $\rho = 0.5$ | $\rho = 0.7$ |
| 2 | $5.31734^{58}_{16}$ | $5.2768^{13}_{08}$ | $5.18758^{96}_{74}$ | $5.02529^{78}_{34}$ |
| 4 | $4.3110^{81}_{77}$ | $4.28342^{36}_{02}$ | $4.22239^{63}_{28}$ | $4.11107^{52}_{12}$ |
| 6 | $3.89658^{81}_{49}$ | $3.87377^{47}_{14}$ | $3.82350^{55}_{20}$ | $3.73205^{79}_{38}$ |
| 8 | $3.66318^{49}_{34}$ | $3.64307^{81}_{48}$ | $3.59884^{95}_{59}$ | $3.51864^{45}_{01}$ |
| 10 | $3.51093^{94}_{60}$ | $3.49262^{99}_{65}$ | $3.4524^{23}_{18}$ | $3.37971^{84}_{35}$ |

Table 3.2: Upper 0.01 percentile points of the bivariate F distribution

| | $n = 10$ | | | |
|---|---|---|---|---|
| $m$ | $\rho = 0.1$ | $\rho = 0.3$ | $\rho = 0.5$ | $\rho = 0.7$ |
| 2 | $9.3014^{99}_{81}$ | $9.2532^{24}_{09}$ | $9.1430^{61}_{46}$ | $8.9312^{17}_{01}$ |
| 4 | $7.1869^{98}_{87}$ | $7.1526^{70}_{58}$ | $7.0749^{36}_{24}$ | $6.9271^{94}_{83}$ |
| 6 | $6.3616^{99}_{77}$ | $6.3325^{93}_{83}$ | $6.26714^{96}_{03}$ | $6.1440^{67}_{58}$ |
| 8 | $5.9094^{83}_{61}$ | $5.8833^{28}_{19}$ | $5.8248^{23}_{14}$ | $5.7156^{75}_{66}$ |
| 10 | $5.6197^{60}_{38}$ | $5.5955^{96}_{87}$ | $5.5417^{56}_{47}$ | $5.4419^{38}_{29}$ |

# Chapter 4

# AN INTERVAL ANALYSIS APPROACH TO THE EM ALGORITHM


## Abstract

The EM algorithm is widely used in incomplete-data problems (and some complete-data problems) for parameter estimation. One limitation of the EM algorithm is that upon termination, it is not always near a global optimum. As reported by Wu (1982), when several stationary points exist, convergence to a particular stationary point depends on the choice of starting point. Furthermore, convergence to a saddle point or local minimum is also possible. In the EM algorithm, although the loglikelihood is unknown, an interval containing the gradient of the EM $q$ function can be computed at individual points using interval analysis methods. By using interval analysis to enclose the gradient of the EM $q$ function (and, consequently, the loglikelihood), an algorithm is developed which is able to locate all stationary points of the loglikelihood within any designated region of the parameter space. The algorithm is applied to several examples. In one example involving the $t$ distribution, the algorithm successfully locates (all) seven stationary points of the loglikelihood.



**Key Words:** Interval arithmetic, Optimization, Interval EM


## 4.1 Introduction

This article explores a variation of the EM algorithm which uses techniques of interval analysis to locate multiple stationary points of a loglikelihood.

Interval analysis can be used to compute an interval which encloses the range of a function over a given domain. By using interval analysis to compute an enclosure of the gradient of the loglikelihood over specific regions, those regions where the enclosure of the gradient does not contain zero can be ruled out from containing any stationary points.

The algorithm locates stationary points by repeatedly dividing into smaller regions precisely those regions which have not been ruled out.

The structure of this paper proceeds as follows. Section 4.2 presents an introduction to interval analysis sufficient to understand this paper. Some of the differences between calculations with real numbers and interval numbers are noted, along with some comments about performing interval arithmetic on digital computers. Section 4.3 briefly states the traditional EM algorithm, then Section 4.4 presents a new approach to the EM algorithm using interval analysis. Section 4.5 presents several examples of the algorithm applied to different problems. These examples demonstrate both the accuracy which interval arithmetic can provide and the ability of the algorithm to locate multiple stationary points. Section 4.6 provides some conclusions.

## 4.2 Interval Analysis

A good introduction to Interval Analysis can be found in monographs by Hansen (1992) and Moore (1979). Some of the fundamental concepts of interval analysis are now presented.

In this paper, intervals will be indicated by superscript $I$ and vectors will be denoted by boldface. An interval $x^I = [\underline{x}, \overline{x}]$ is a closed and bounded set of real numbers. For two intervals $x^I$ and $y^I$, interval arithmetic operators are defined in the following manner:

$$x^I \circ y^I = \{x \circ y : x \in x^I, y \in y^I\}$$

where $\circ \in \{+, -, *, /\}$ and division is undefined for $0 \in y^I$. For these four interval arithmetic operators, closed-form expressions can be obtained for direct calculation of results of the operations. For example, if $x^I = [\underline{x}, \overline{x}]$ and $y^I = [\underline{y}, \overline{y}]$, then $x^I + y^I = [\underline{x} + \underline{y}, \overline{x} + \overline{y}]$. The *Hull* of a set of intervals $x_1^I, \ldots, x_n^I$ is the smallest interval containing $x_1^I, \ldots, x_n^I$, i.e. $Hull(x_1^I, \ldots, x_n^I) = [\inf\{x : x \in x_i^I; i = 1, \ldots, n\}, \sup\{x : x \in x_i^I; i = 1, \ldots, n\}]$. An *interval vector* or *box* is simply a vector of intervals. An *interval function* is an interval-valued function of one or more interval arguments. In this paper, capital letters are used to denote interval functions. An interval function $F(x_1^I, \ldots x_n^I)$ is said to be an *interval extension* or *interval enclosure* of $f(x_1, \ldots, x_n)$ if $F([x_1, x_1], \ldots, [x_n, x_n]) = f(x_1, \ldots, x_n)$ for all $x_i; i = 1, \ldots, n$. An interval function $F$ is said to be *inclusion monotonic* if $F(x^I) \subset F(y^I)$ whenever $x^I \subset y^I$. A fundamental property of interval analysis is that rational interval functions are inclusion monotonic.

In this paper, the *natural interval extension* of a real function is used. This is an interval extension in which intervals and interval operations are substituted for scalars and scalar operations. The value of any interval extension of a function is dependent on the form of the real function. For example, let $f_1(x) = (x-1)(x+1)$ and $f_2(x) = xx - 1$. Let $F_1$ and $F_2$ be the corresponding natural interval extensions and let $x^I = [-2, 1]$. Then $F_1(x^I) = [-6, 3]$ and $F_2(x^I) = [-3, 3]$ which both contain $[-1, 3]$, the true range of $f_1$ and $f_2$ over $x^I$. This feature of interval computations to sometimes overestimate the range of a function is referred to as *interval dependency*. Attention must be given to the exact expression of an interval function to reduce the effect of interval dependency. Hansen (1992, 1997) presents some results regarding this topic.

When implementing interval arithmetic calculations on computers, care must be taken to ensure that rounding errors do not invalidate the inclusion monotonicity of interval results. One way to achieve this is through the use of directed rounding modes in the floating-point calculations. When calculating the lower endpoint of an interval result, the floating-point processor is set to round all results *down*. For calculation of the upper endpoint of an interval result, all calculations are rounded *up*. Using the symbols $\bigtriangledown$ and $\bigtriangleup$ to denote downward and upward rounding respectively, the actual computer implementation of interval addition is $x^I + y^I = [\bigtriangledown(\underline{x} + \underline{y}), \bigtriangleup(\overline{x} + \overline{y})]$. Correct use of the rounding modes guarantees that the computed result contains the true interval answer. Some programming languages and software packages are able to work with interval data types and interval operators. INTLIB_90 and C-XSC are among the more widely referenced tools. Some mention of other software environments is found in Kearfott (1996b). For the research in this paper, the computations were done using the BIAS/PROFIL package in C++ developed by Knüppel (1993c).

## 4.3 The EM Algorithm

The present-day incarnation of the EM algorithm first appeared in a landmark paper by Dempster et al. (1977). The EM algorithm is a general iterative algorithm for maximum likelihood estimation in incomplete-data problems. The EM algorithm has not only been successfully applied in obvious incomplete-data problems, but also in many situations where the data appears to be complete, but can be viewed as incomplete by introducing latent variables. The intuitive idea behind the EM algorithm is to iterate the following two steps:

Expectation step: Replace missing values (sufficient statistics) by estimated values.

Maximization step: Estimate parameters as if no data were missing.

Formally, starting with a parameter estimate $\boldsymbol{\phi}_p$, the E-step calculates the conditional expectation of the complete-data log likelihood, $\log L_c(\boldsymbol{\phi})$, as $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p) = E_{\boldsymbol{\phi}_p}\{\log L_c(\boldsymbol{\phi})\}$ and then the M-step chooses $\boldsymbol{\phi}_{p+1}$ to be any value of $\boldsymbol{\phi} \in \boldsymbol{\Omega}$ that maximizes $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$, i.e. $q(\boldsymbol{\phi}_{p+1}|\boldsymbol{\phi}_p) \geq q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$ for any $\boldsymbol{\phi} \in \Omega$.

## 4.4 Interval Arithmetic and EM

A method is now presented which uses certain properties of the EM algorithm and of interval arithmetic to locate all stationary points of the likelihood inside of a given region of the parameter space. Briefly, from the EM algorithm it is known that the $q$ function has a gradient which is equal to the gradient of the loglikelihood at stationary points of the loglikelihood. Using interval arithmetic, it is possible to derive interval vectors which enclose values of the gradient of the $q$ function even over regions which do not contain a stationary point.

The complete method is presented below, followed by a summary outline and additional comments. Some numerical results are presented in Section 4.5.

### 4.4.1 Enclosing the Gradient of the Log Likelihood

The fundamental task for the method being proposed will be to eliminate regions of the parameter space where it can be determined that a stationary point of the likelihood does not exist. This can be accomplished by finding a box which encloses the range of the gradient of the loglikelihood over a region. If, for example, the interval enclosure of the set of all values of the gradient of the loglikelihood $\ell(\boldsymbol{\phi})$ over the box $\boldsymbol{\phi}^I$,

$$\left\{ \left. \frac{\partial \ell(\boldsymbol{\phi})}{\partial \boldsymbol{\phi}} \right|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} : \boldsymbol{\phi}_p \in \boldsymbol{\phi}^I \right\},$$

does not contain zero in one or more of its coordinates, then the gradient of the log likelihood is nonzero over $\boldsymbol{\phi}^I$ and $\ell(\boldsymbol{\phi})$ does not contain a stationary point inside the box $\boldsymbol{\phi}^I$. A more thorough explanation of how this is accomplished is now presented by deriving an interval enclosure for the gradient of the log likelihood. The first part of this derivation is similar to the development in Dempster et al. (1977).

Denote the complete data (which includes missing values) by $\mathbf{x}$ and the observed (incomplete) data by $\mathbf{y}$, where $\mathbf{y} = \mathbf{y}(\mathbf{x})$. Let the density function of $\mathbf{x}$ be $f(\mathbf{x}|\boldsymbol{\phi})$, where $\boldsymbol{\phi} \in \boldsymbol{\Omega}$. From this, the density function for $\mathbf{y}$ is

$$g(\mathbf{y}|\boldsymbol{\phi}) = \int_{\mathbf{x}(\mathbf{y})} f(\mathbf{x}|\boldsymbol{\phi}) d\mathbf{x}.$$

For simplicity and tractability, the maximization step would ideally be accomplished over $\boldsymbol{\phi}$ in $\log f(\mathbf{x}|\boldsymbol{\phi})$. However, since $\mathbf{x}$ is unobservable, replace $\log f(\mathbf{x}|\boldsymbol{\phi})$ by its conditional expectation. To that end, let $k(\mathbf{x}|\mathbf{y}, \boldsymbol{\phi}) = f(\mathbf{x}|\boldsymbol{\phi})/g(\mathbf{y}|\boldsymbol{\phi})$ be the conditional density of $\mathbf{x}$ given $\mathbf{y}$ and $\boldsymbol{\phi}$. Using this, the log-likelihood can be written

$$\ell(\boldsymbol{\phi}) = \log g(\mathbf{y}|\boldsymbol{\phi}) = \log f(\mathbf{x}|\boldsymbol{\phi}) - \log k(\mathbf{x}|\mathbf{y}, \boldsymbol{\phi}).$$

Taking the conditional expectation (using $\boldsymbol{\phi}_p$ as an estimate for $\boldsymbol{\phi}$),

$$\ell(\boldsymbol{\phi}) = E_{\boldsymbol{\phi}_p}\left[\log f(\mathbf{x}|\boldsymbol{\phi})|\mathbf{y}\right] - E_{\boldsymbol{\phi}_p}\left[\log k(\mathbf{x}|\mathbf{y}, \boldsymbol{\phi})|\mathbf{y}\right].$$

For simplicity, this is often written $\ell(\boldsymbol{\phi}) = q(\boldsymbol{\phi}|\boldsymbol{\phi}_p) - h(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$. To find values of $\boldsymbol{\phi} \in \boldsymbol{\Omega}$ which maximize $\ell(\boldsymbol{\phi})$, solutions to

$$\frac{\partial \ell(\boldsymbol{\phi})}{\partial \boldsymbol{\phi}} = \frac{\partial q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}} - \frac{\partial h(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}} = 0$$

are needed.

Now, it is easy to show that $h(\boldsymbol{\phi}|\boldsymbol{\phi}_p) \leq h(\boldsymbol{\phi}_p|\boldsymbol{\phi}_p)$ for any $\boldsymbol{\phi} \in \boldsymbol{\Omega}$, i.e. $\boldsymbol{\phi}_p$ maximizes $h(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$ with respect to $\boldsymbol{\phi}$, and so $\left. \frac{\partial h(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}} \right|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} = 0$. It is therefore sufficient

when searching for maxima of $\ell(\boldsymbol{\phi})$ to limit consideration to $\frac{\partial q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}}$, specifically, to an enclosure of the gradient of the $q$ function over the box $\boldsymbol{\phi}^I$,

$$\left\{ \frac{\partial q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} : \boldsymbol{\phi}_p \in \boldsymbol{\phi}^I \right\}$$

for arbitrary $\boldsymbol{\phi}^I \in \boldsymbol{\Omega}$. Let $Q'(\boldsymbol{\phi}|\boldsymbol{\phi}^I) = [\underline{Q}'(\boldsymbol{\phi}|\boldsymbol{\phi}^I), \overline{Q}'(\boldsymbol{\phi}|\boldsymbol{\phi}^I)]$ be an interval extension of $\frac{\partial q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}}$ for interval $\boldsymbol{\phi}^I$ and $\boldsymbol{\phi}_p \in \boldsymbol{\phi}^I$. Note that $Q'(\boldsymbol{\phi}|\boldsymbol{\phi}^I)$ is *not* $\frac{\partial Q(\boldsymbol{\phi}|\boldsymbol{\phi}^I)}{\partial \boldsymbol{\phi}}$ where $Q(\boldsymbol{\phi}|\boldsymbol{\phi}^I)$ is the interval extension of $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$. Also, let $Q'_2(\boldsymbol{\phi}^I|\boldsymbol{\phi}^I)$ be an interval extension of $Q'(\boldsymbol{\phi}|\boldsymbol{\phi}^I)$.

At each $\boldsymbol{\phi}_p \in \boldsymbol{\phi}^I$, the enclosure of the gradient of the log likelihood can be obtained by

$$\frac{\partial \ell(\boldsymbol{\phi})}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} = \frac{\partial q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} \in [\underline{Q}'(\boldsymbol{\phi}_p|\boldsymbol{\phi}^I), \overline{Q}'(\boldsymbol{\phi}_p|\boldsymbol{\phi}^I)]$$

and

$$\left\{ \frac{\partial \ell(\boldsymbol{\phi})}{\partial \boldsymbol{\phi}}\Big|_{\boldsymbol{\phi}=\boldsymbol{\phi}_p} : \boldsymbol{\phi}_p \in \boldsymbol{\phi}^I \right\} \subset Q'_2(\boldsymbol{\phi}^I|\boldsymbol{\phi}^I).$$

If zero is not contained in $Q'_2(\boldsymbol{\phi}^I|\boldsymbol{\phi}^I)$, then the box $\boldsymbol{\phi}^I$ cannot contain a local maximizer of $\ell(\boldsymbol{\phi})$ and may therefore be excluded from further consideration. Thus $Q'_2(\boldsymbol{\phi}^I|\boldsymbol{\phi}^I)$ is an interval-valued function which encloses the union of the ranges of a class of interval functions $q(\boldsymbol{\phi}|\boldsymbol{\phi}^I)$ indexed by $\boldsymbol{\phi} \in \boldsymbol{\phi}^I$.

After a user of this method specifies an initial box $\boldsymbol{\phi}^I \in \boldsymbol{\Omega}$, locating optima of the loglikelihood proceeds by conducting a bisection search by dividing $\boldsymbol{\phi}^I$ into successively smaller boxes and evaluating the enclosure of the gradient of the log likelihood over each box. Boxes which do not contain a stationary point are discarded. The initial box $\boldsymbol{\phi}^I$ will frequently be quite large so as to (hopefully) enclose all stationary points of $\ell(\boldsymbol{\phi})$. At a certain point in this process, typically when the box size becomes smaller than a specified size, the subdividing stops and a list $\mathcal{G}$ of boxes from the grid search is output along with the enclosure of the gradient and the enclosure of the range of $q$ functions over each box. These boxes contain all the stationary points of $\ell(\boldsymbol{\phi})$ that exist within the initial interval box $\boldsymbol{\phi}^I$.

### 4.4.2 Definitions for Interval EM

In this section an interval EM algorithm is defined. A few necessary definitions are stated and then utilized in the interval EM method being presented.

Definition. An *interval EM algorithm* on an interval vector $\boldsymbol{\Phi}$ in a parameter space $\boldsymbol{\Omega}$ is an iterative method which employs a sequences of intervals $\boldsymbol{\phi}_0^I \to \boldsymbol{\phi}_1^I \to \cdots \boldsymbol{\phi}_p^I \to$ with respect to interval enclosures $Q(\boldsymbol{\phi}|\boldsymbol{\phi}_0^I)$, $Q(\boldsymbol{\phi}|\boldsymbol{\phi}_1^I), \cdots, Q(\boldsymbol{\phi}|\boldsymbol{\phi}_p^I)$ of sets of functions $q(\boldsymbol{\phi}|\boldsymbol{\phi}_0)$, $q(\boldsymbol{\phi}|\boldsymbol{\phi}_1), \cdots, q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$ so that $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p) \in Q(\boldsymbol{\phi}|\boldsymbol{\phi}_p^I)$ where $\boldsymbol{\phi}_p \in \boldsymbol{\phi}_p^I \subset \boldsymbol{\Phi}$ for each $p$. The interval $\boldsymbol{\phi}_{p+1}^I$ contains at least one value of $\boldsymbol{\phi}_{p+1}$ which maximizes a $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p)$ for at least one $\boldsymbol{\phi}_p \in \boldsymbol{\phi}_p^I \subset \boldsymbol{\Phi}$. Moving from $\boldsymbol{\phi}_p^I$ to $\boldsymbol{\phi}_{p+1}^I$ is referred to as an *interval EM step.*

Definition. An *interval GEM algorithm* is an interval EM algorithm except instead of maximizing $q(\boldsymbol{\phi}|\boldsymbol{\phi}_p^I)$ with respect to $\boldsymbol{\phi}$, the interval $\boldsymbol{\phi}_{p+1}^I$ contains as least one value $\boldsymbol{\phi}_{p+1}$ such that $q(\boldsymbol{\phi}_{p+1}|\boldsymbol{\phi}_p) \geq q(\boldsymbol{\phi}_p|\boldsymbol{\phi}_p)$, where $\boldsymbol{\phi}_{p+1} \in \boldsymbol{\phi}_{p+1}^I$, $\boldsymbol{\phi}_p \in \boldsymbol{\phi}_p^I$. Moving from $\boldsymbol{\phi}_p^I$ to $\boldsymbol{\phi}_{p+1}^I$ is referred to as an *interval GEM step*.

The current implementation of the method may be more easily understood by referring to Figure 4.1, which graphically illustrates an interval EM step in a hypothetical one-dimensional case. In Figure 4.1, the dotted lines $q(\boldsymbol{\phi}|\boldsymbol{\phi}_i)$ and $q(\boldsymbol{\phi}|\boldsymbol{\phi}_j)$ are two separate scalar $q$ functions that might be enountered in different iterations of a scalar EM algorithm. The solid lines $\underline{Q}(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$ and $\overline{Q}(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$ denote the extent of an interval-valued function $Q(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$ which encloses all the scalar $q$ functions $q(\boldsymbol{\phi}|\boldsymbol{\phi}_i)$ and $q(\boldsymbol{\phi}|\boldsymbol{\phi}_j)$ indexed by $\boldsymbol{\phi}_i \in \boldsymbol{\phi}_k^I$, $\boldsymbol{\phi}_j \in \boldsymbol{\phi}_k^I$. Finally, the vertical line segments $Q(\boldsymbol{\phi}_{k+1,1}^I|\boldsymbol{\phi}_{k+1,1}^I)$ and $Q(\boldsymbol{\phi}_{k+1,2}^I|\boldsymbol{\phi}_{k+1,2}^I)$ denote enclosures of $Q(\boldsymbol{\phi}|\boldsymbol{\phi}^I)$ evaluated for $\boldsymbol{\phi} \in \boldsymbol{\phi}_{k+1,1}^I$ and $\boldsymbol{\phi} \in \boldsymbol{\phi}_{k+1,2}^I$, respectively.

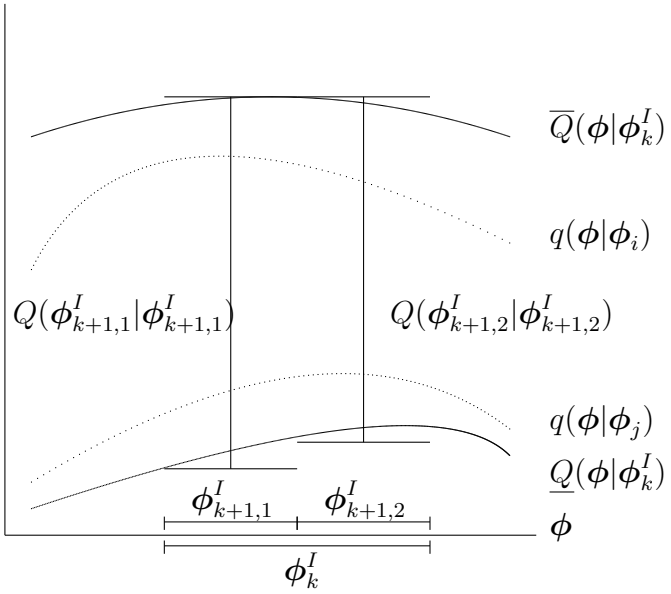


Figure 4.1: One interval EM step. $\overline{Q}(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$ and $\underline{Q}(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$ bound the extent of the interval-valued function $Q(\boldsymbol{\phi}|\boldsymbol{\phi}_k^I)$, while $q(\boldsymbol{\phi}|\boldsymbol{\phi}_i)$ and $q(\boldsymbol{\phi}|\boldsymbol{\phi}_j)$ are examples of two of the scalar functions contained within the interval function.

### 4.4.3 Full Bisection Search

The bisection algorithm starts with an initial box $\boldsymbol{\phi}_0^I$ in a list of boxes $\mathcal{G}$. The method proceeds by simply bisecting boxes from $\mathcal{G}$ until no boxes are left or until all boxes have reached a certain size. Let $m$ be the dimension of $\boldsymbol{\phi}$ and initialize $i := 1$. Proceed by removing and bisecting each box of $\mathcal{G}$ in the $i^{th}$ coordinate. Discard any boxes which do not contain zero in at least one direction of the enclosure of the gradient. Return all remaining boxes to $\mathcal{G}$ and increase $i$ by 1, resetting $i := 1$ when $i > m$. Repeat as

necessary until the diameter of every box is small. If at any point $\mathcal{G}$ becomes empty, print a message stating that no stationary points were contained in the initial region $\boldsymbol{\phi}_0^I$.

The bisection algorithm differs from traditional EM in that there are no expectation and maximization steps. The only use of the EM theory was to obtain an enclosure for the gradient of the loglikelihood of $\boldsymbol{\phi}$. Still, the bisection search can in some way be viewed as many simultaneous interval GEM algorithms. In making an interval GEM step from $\boldsymbol{\phi}_k^I$ to $\boldsymbol{\phi}_{k+1,1}^I$ and from $\boldsymbol{\phi}_k^I$ to $\boldsymbol{\phi}_{k+1,2}^I$, there will be a nondecreasing change in the lower bound of the enclosure of the $q$ functions, i.e. $\underline{Q}(\boldsymbol{\phi}_k^I|\boldsymbol{\phi}_k^I) \leq \underline{Q}(\boldsymbol{\phi}_{k+1,i}^I|\boldsymbol{\phi}_{k+1,i}^I)$ for $i = 1, 2$.

- ALGORITHM: Bisection Interval EM Search

  Input an initial interval box $\boldsymbol{\phi}_0^I$ and place it as the only element of the list $\mathcal{G}$.

  i := 0

  REPEAT

    i := (i + 1) mod m

    FOR j = 1 TO Length($\mathcal{G}$)

      Remove the first box from $\mathcal{G}$. Call it $\boldsymbol{\phi}^I$

      Bisect $\boldsymbol{\phi}^I$ along the $i^{th}$ direction, creating $\boldsymbol{\phi}_1^I$ and $\boldsymbol{\phi}_2^I$

      If $0 \in Q_2'(\boldsymbol{\phi}_k^I|\boldsymbol{\phi}_k^I)$, append $\boldsymbol{\phi}_k^I$ to $\mathcal{G}$, $k = 1, 2$

    NEXT

  UNTIL $\mathcal{G}$ is empty or maximum diameter of boxes $\leq \epsilon$

The method described above will not, of course, find any global optima which lie outside of the initial box $\boldsymbol{\phi}_0^I \subset \Omega$. In practice this is often not of concern, primarily because the observed data places practical limitations on the portion of the parameter space of interest. Also, in a manner similar to that observed by Hansen (1992), it is often possible to make the parameter space exceedingly large without significantly increasing the computing time to search for global optima.

Because the algorithm uses intervals instead of real numbers, measurement error in data and floating-point approximations can immediately be incorporated. For example, one might use $\pi^I = [3.14, 3.15]$ to indicate uncertainty in known constants. Even more useful is the ability to represent data as intervals, e.g. $x_i^I = [x_i - \delta, x_i + \delta]$, where $x_i$ is the observed value and $\delta$ is a bound on the measurement error.

### 4.4.4 Quick Search

For reasons of speed, memory, or accuracy considerations, the method described above may not always be optimally suitable. Let $m$ be the dimension of $\boldsymbol{\phi}$. Bisection of just one box from $\mathcal{G}$ has the potential to create $2^m$ additional boxes that will be added to $\mathcal{G}$. This might happen in situations where a region contains many stationary points or where the loglikelihood is relatively flat and the gradient is near zero. Since interval arithmetic

sometimes calculates an interval wider than optimal, it may be the case that the gradient is nonzero in every direction, but the enclosure of the gradient contains zero in at least one direction. If some combination of high dimensionality and/or fairly flat likelihood occurs, the length of $\mathcal{G}$ can grow exponentially.

A variation on the algorithm given above is now presented as a faster, smaller alternative. The variation comes about simply as a matter of which order the boxes of $\mathcal{G}$ are added to the list.

As before, boxes are removed from the start of the list $\mathcal{G}$. After removal, the box $\boldsymbol{\phi}$ is bisected in the coordinate of the maximum width of $\boldsymbol{\phi}$. Boxes for which the enclosure of the gradient does not contain zero are discarded. If only one half of $\boldsymbol{\phi}$ remains, prepend it to the list $\mathcal{G}$. If both halves remain, then evaluate and enclosure of the $q$ function, $Q(\boldsymbol{\phi}_i^I|\boldsymbol{\phi}_i^I)$, for each half. Prepend both halves to $\mathcal{G}$, with the half that has the greater lower bound of $Q(\boldsymbol{\phi}_i^I|\boldsymbol{\phi}_i^I)$ added last. (In Figure 4.1, $\boldsymbol{\phi}_2$ is prepended after $\boldsymbol{\phi}_1$.) When the first box of $\mathcal{G}$ has reached a user-specified tolerance size, the algorithm stops and prints out only the first box of $\mathcal{G}$.

This approach will ensure that at least one box $\boldsymbol{\phi}$ at the start of the list $\mathcal{G}$ is made as small as possible in as short of time as possible. Of course, the gain in speed comes at the price of losing the guarantee that a stationary point is contained in the output. It could happen that if a smaller tolerance was used, at the next step the box would be bisected and both halves discarded. All that can be said is that in the final box output, the enclosure of the gradient over the box contains zero in each direction. Nonetheless, given the speed advantages, this algorithm is potentially useful. It may provide a quick answer which can suggest a location for a fairly small region to feed into the full bisection search given above.

- ALGORITHM: Quick Interval EM Search

  Input an initial box $\boldsymbol{\phi}_0^I$ and place it as the only element of the list $\mathcal{G}$.

  Input $\epsilon$

  REPEAT

  Remove the first box from $\mathcal{G}$. Call it $\boldsymbol{\phi}^I$

  Bisect $\boldsymbol{\phi}^I$ along the direction of maximum width, creating $\boldsymbol{\phi}_1^I$ and $\boldsymbol{\phi}_2^I$

  If $0 \in Q_2'(\boldsymbol{\phi}_k^I|\boldsymbol{\phi}_k^I)$, prepend $\boldsymbol{\phi}_k^I$ to $\mathcal{G}$, $k = 1, 2$.

  UNTIL $\mathcal{G}$ is empty or the first box of $\mathcal{G}$ has maximum diameter $\leq \epsilon$

## 4.5 Examples

Several examples are now presented to illustrate use of the method described above. Note that the following examples each have an algebraic, real expression for $q(\boldsymbol{\phi}|\boldsymbol{\phi}_k)$. This is consistent with traditional EM notation. Though not shown, a person would then determine an expression for the gradient of this function with respect to $\boldsymbol{\phi}$, $q'(\boldsymbol{\phi}|\boldsymbol{\phi}_k)$,

and then express $q'(\boldsymbol{\phi}_k|\boldsymbol{\phi}_k)$ in as simple a way as possible. This is coded in the program as $Q_2'(\boldsymbol{\phi}_k|\boldsymbol{\phi}_k)$.

When numerical results are reported, sub/superscript notation will sometimes be used to simplify the representation of an interval, e.g. $[2.33, 2.35] = 2.3^5_3$.

### 4.5.1 Multinomial Example

The following example from Dempster et al. (1977) is frequently used to introduce the EM algorithm. Consider a set of 197 animals which are classified into four categories. The observed classification counts are $\mathbf{y} = (y_1, y_2, y_3, y_4) = (125, 18, 20, 34)$. The classification of the random variable $Y$ is modeled as following a multinomial distribution:

$$Y \sim Multinomial\left(197, \frac{1}{2} + \frac{1}{4}p, \frac{1}{4} - \frac{1}{4}p, \frac{1}{4} - \frac{1}{4}p, \frac{1}{4}p\right)$$

where $p$ is unknown and to be estimated. There is no missing data in this problem and $p$ is easily estimated by a maximum likelihood approach. For illustration purposes, the problem is reformulated with missing data. Suppose the first classification category $Y_1$ is split into two categories and a new random variable $X$ is modeled:

$$X \sim Multinomial\left(197, \frac{1}{2}, \frac{1}{4}p, \frac{1}{4} - \frac{1}{4}p, \frac{1}{4} - \frac{1}{4}p, \frac{1}{4}p\right).$$

The incomplete data vector $\mathbf{x}$ is $(x_1, x_2, x_3, x_4, x_5)$ and thus $\mathbf{y}$ can be written $\mathbf{y}(\mathbf{x}) = (x_1 + x_2, x_3, x_4, x_5)$. Here $x_1$ and $x_2$ are unobserved except through their sum $x_1 + x_2$. It can be shown that

$$q(p|p_k) = k(\mathbf{x}) + \left[125\frac{\frac{p_k}{4}}{\frac{1}{2} + \frac{p_k}{4}} + x_5\right]\frac{1}{p} - (x_3 + x_4)\frac{1}{1-p} \tag{4.1}$$

where $k(\mathbf{x})$ does not depend on p and can be ignored in the maximization step. Figure 4.2 shows a plot of the corresponding interval extension, $Q(p|p_k^I)$. An accurate interpretation of this interval-valued function can be had in this case by actually overlaying plots of $q(p|p_k)$ for various $p \in p_k^I$, in this case $p = 0.1(0.1)0.9$.

The initial interval selected is $p_0^I = [.00001, .99999]$. While a wider interval can be used, the maximum likelihood estimate of $p$ is certainly contained in $[.00001, .99999]$. Furthermore, the values of $p = 0$ and $p = 1$ are excluded by equation (4.1). If the user selected an inappropriate value for the initial interval, such as $p^I = [0.1, 0.2]$, then the algorithm terminates with the message:

```
Gradient of Q(Phi|Phi_k) = ([152.262,411.414])
Gradient of likelihood does not contain zero.
No stationary point in ([0.1,0.2])
```

The bisection algorithm applied to this problem using initial interval $p_0^I$ produces a list $\mathcal{G}$ which contains two interval boxes,

$$y_1 = 0.6268214978709824^4_3$$

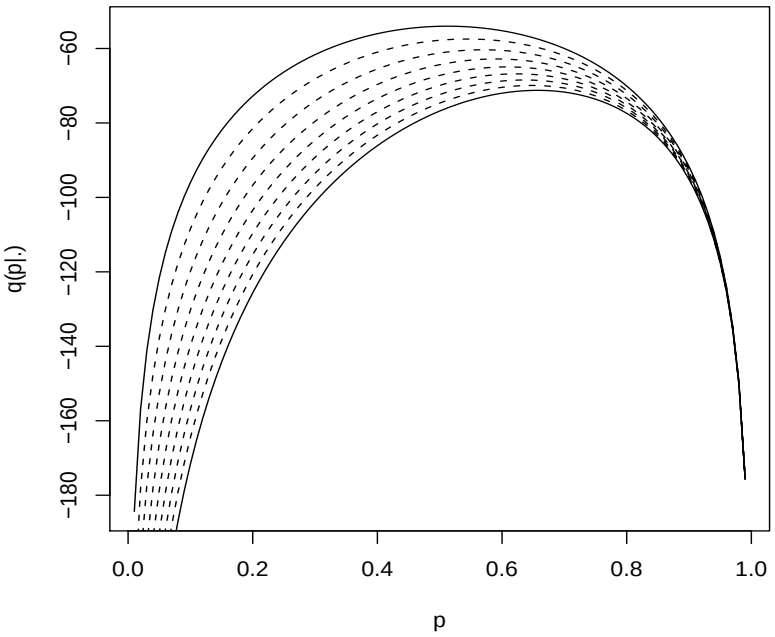


Figure 4.2: Plot of $Q(p|p_0^I)$ versus $p$ for $p_0^I = [0.1, 0.9]$.

$$y_2 = 0.626821497870982_4^5.$$

Any stationary points of the log-likelihood are guaranteed to be contained in the hull of the boxes in the list $\mathcal{G}$. If a scalar estimate is desired, the midpoint of the hull can be given: $\hat{p} = 0.6268214978709824$.

### 4.5.2 Univariate t Example

McLachlan and Krishnan (1997) give an example by Arslan et al. (1993) where the EM algorithm can converge to a local *minimum*. A $p$-dimensional random variable $\mathbf{W}$ is said to have a multivariate $t$-distribution $t_p(\boldsymbol{\mu}, \Sigma, \nu)$ with location $\boldsymbol{\mu}$, positive definite inner product matrix $\Sigma$, and degrees of freedom $\nu$ when the density of $\mathbf{W}$ is given by

$$f_p(\mathbf{w}|\boldsymbol{\mu}, \Sigma, \nu) = \frac{\Gamma(\frac{p+\nu}{2})|\Sigma|^{-1/2}}{(\pi\nu)^{p/2}\Gamma(\frac{\nu}{2})\{1 + (\mathbf{w}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{w}-\boldsymbol{\mu})/\nu\}^{(p+\nu)/2}}. \tag{4.2}$$

The example considered is a univariate case of the $t$-distribution where $\nu = 0.05$, $\Sigma = 1$, and $\mu$ is taken as unknown. The observed data is $\mathbf{w} = (-20, 1, 2, 3)$. Ignoring additive and multiplicative constants, the log likelihood is $\log L(\mu) \propto -\sum_i \log\{1 + 20(w_i - \mu)\}$. A plot showing the shape of this log likelihood appears in Figure 4.3.

The function has seven stationary points. The most interesting are the local maxima at $\mu_2 = 1.086$, $\mu_3 = 1.997$, and $\mu_4 = 2.906$. In this complete-data problem it is possible

to graph the log-likelihood and visually choose starting values that will cause a scalar

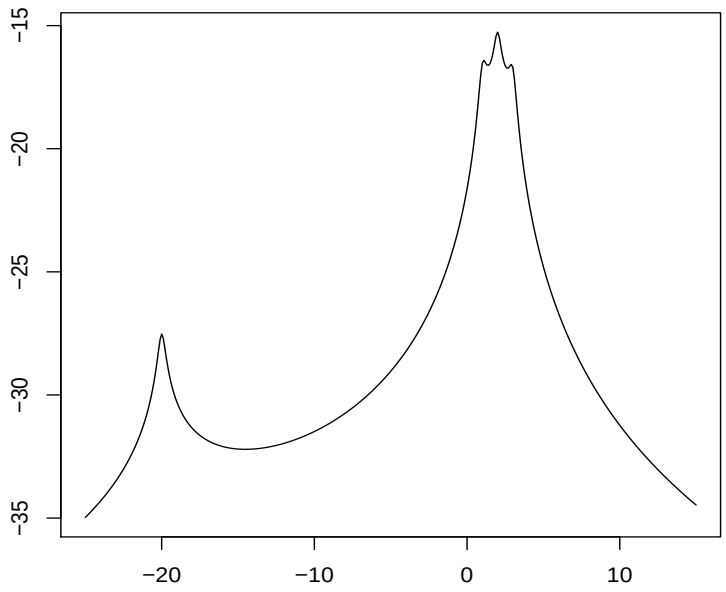


Figure 4.3: Plot of log likelihood function log $L(\mu)$ versus $\mu$. Local maxima occur at $\mu_1 = -19.993$, $\mu_2 = 1.086$, $\mu_3 = 1.997$, and $\mu_4 = 2.906$.

EM algorithm to converge to each of the local maxima, *and even to a local minimum.* However, the domain of attraction for each stationary point is not necessarily a contiguous region.

Using $\mu_0 = [-1000, 1000]$, the bisection algorithm completes 59 iterations (bisections), during which the length of $\mathcal{G}$ is scarcely longer than the 20 boxes at the final step. These boxes occur in distinct groupings around each of the seven stationary points. While the algorithm actually outputs the list of boxes from $\mathcal{G}$, for brevity, the hull of each group of boxes and the hull of the associated enclosures of the $q$ functions are given in Table 4.1.

Looking at this table, the nature of each stationary point is not immediately clear. Since this is a univariate case, it would be possible to evaluate the gradient on either side of each $\boldsymbol{\phi}_{S_i}$ and thereby determine which stationary points are local maxima and which are local minima. However, it is immediately clear from the table that $\boldsymbol{\phi}_{S_5}$ gives the largest value of $Q(\boldsymbol{\phi}_{S_i}|\boldsymbol{\phi}_{S_i})$ and contains the global maximum of the log-likelihood as displayed by Figure 4.3.

Table 4.1: Enclosures of the stationary points for the univariate $t$ example.

| $i$ | $\boldsymbol{\phi}_{S_i}$ | $Q(\boldsymbol{\phi}_{S_i}\|\boldsymbol{\phi}_{S_i})$ |
|---|---|---|
| 1 | $-19.9931646088712^{29}_{30}$ | $-1.5753266627959^{4}_{7}$ |
| 2 | $-14.5161774794253^{0}_{2}$ | $-2.0988377864^{297}_{302}$ |
| 3 | $1.0861678063107^{7}_{0}$ | $-1.6060938703^{397}_{426}$ |
| 4 | $1.3731761015634^{32}_{18}$ | $-1.8922427508^{2981}_{3016}$ |
| 5 | $1.9975126089118^{24}_{17}$ | $-1.5250098867^{386}_{402}$ |
| 6 | $2.6468546770426^{35}_{20}$ | $-1.8841583622^{176}_{208}$ |
| 7 | $2.9056308944679^{85}_{75}$ | $-1.6170241742^{677}_{707}$ |

### 4.5.3 Binomial-Poisson Mixture Example

This example from Thisted (1988) presents a simple multivariate-parameter example dealing with the number of children per widow in a pension fund.

| Children per widow, $i$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Number of widows, $n_i$ | 3062 | 587 | 284 | 103 | 33 | 4 | 2 |

Since the number of widows with no children is larger than would be expected for a Poisson distribution, it is assumed that there are actually two underlying populations. The number of children $Y$ for a widow is modeled as

$$Y \sim \begin{cases} 0 & \text{with probability } \xi \\ \text{Poisson}(\lambda) & \text{with probability } 1-\xi \end{cases} \tag{4.3}$$

With $\boldsymbol{\phi} = (\lambda, \xi)$, the function to be maximized in the M-step is:

$$q(\boldsymbol{\phi}|\boldsymbol{\phi}_k) = \frac{n_0 \xi_k}{\xi_k + (1-\xi_k)\exp(-\lambda_k)} \{\log \xi - \log(1-\xi) + \lambda\} + N\{\log(1-\xi) - \lambda\} + \sum_{i=1}^{6} \{i n_i \log \lambda - n_i \log i!\}. \tag{4.4}$$

Based on a visual examination of the data, the starting values of

$$\boldsymbol{\phi}_0 = (\lambda_0^I, \xi_0^I) = ([0.001, 10], [0.001, 0.999])$$

were chosen as being certain to contain the true parameter values.

Applying the Bisection search, after 52 iterations of bisecting $\boldsymbol{\phi}$ in both directions, the list $\mathcal{G}$ contains 82 boxes, the first and last of which are

$$y_1 = (1.037839078989^{60}_{57}, 0.615056697573^{14}_{12})$$

$$y_{82} = (1.037839078989\mathbf{7}^{\mathbf{80}}_{\mathbf{77}}, 0.615056697573\mathbf{12}^{\mathbf{90}}_{\mathbf{88}}).$$

The hull of the boxes on this list is: $\boldsymbol{\phi}_{S_1} = (1.037389078989\mathbf{7}^{\mathbf{80}}_{\mathbf{57}}, 0.615056697573\mathbf{12}^{\mathbf{90}}_{\mathbf{12}})$.

In this problem, what is important is not the extremely narrow (and hence) high degree of accuracy of $\boldsymbol{\phi}_{S_1}$, but the guarantee that considered over the initial parameter space $\boldsymbol{\phi}_0$, the only stationary points of the log-likelihood (if any exist) are guaranteed to be contained in the interval box $\boldsymbol{\phi}_{S_1}$. Moreover, if a scalar EM algorithm converges to some stationary point in $\boldsymbol{\phi}_0$, that point will be inside $\boldsymbol{\phi}_{S_1}$. Using $\epsilon = 10^{-15}$, the Quick search returns $(1.0378390789897^{\mathbf{9}}_{\mathbf{8}}, 0.615056697573\mathbf{13}^{\mathbf{3}}_{\mathbf{2}})$.

### 4.5.4 Genetic Example

This example is also taken from McLachlan and Krishnan (1997).

Suppose there are 435 observations from a multinomial distribution as given in Table 4.2 where $r = 1 - p - q$. The observed data is $(n_O, n_A, n_B, n_{AB})$ and the unknown

Table 4.2: Distribution of data in the genetic example.

| Cell | Cell Probability | Observed Frequency |
|---|---|---|
| O | $r^2$ | $n_O = 176$ |
| A | $p^2 + 2pr$ | $n_A = 182$ |
| B | $q^2 + 2qr$ | $n_B = 60$ |
| AB | $2pq$ | $n_{AB} = 17$ |

parameters are $\boldsymbol{\phi} = (p, q)$. As in the multinomial example above, missing data is introduced by splitting the A and B cells across the sum in the cell probability. The $q$ function is given by

$$\begin{aligned} q(\boldsymbol{\phi}|\boldsymbol{\phi}_k) = &\left(\frac{182}{1 + 2(1 - p_k - q_k)/p_k} + 199\right)\log(p) + \\ &\left(\frac{60}{1 + 2(1 - p_k - q_k)/q_k} + 77\right)\log(q) + \\ &\left(594 - \frac{182}{1 + 2(1 - p_k - q_k)/p_k} - \frac{60}{1 + 2(1 - p_k - q_k)/q_k}\right)\log(1 - p - q). \end{aligned} \quad (4.5)$$

It is not always possible to search the entire portion of the parameter space with one application of the bisection algorithm. In this example, certain combinations of $p^I$ and $q^I$ cause a division by zero error. Specifically, as illustrated in figure 4.4, the gradient does not exist along the lines $p = 0$, $q = 0$, $1-p-q = 0$, $q = 2-2p$, and $2q = 2-p$. The software can be written to catch division by zero errors and mark a box as containing such until further subdivision occurs. Alternatively, the user can specify a smaller initial region. The only stationary point located inside $\boldsymbol{\phi}_0 = (p_0, q_0) = ([0.00001, 0.45], [0.00001, 0.45]$ is found to be located inside $\boldsymbol{\phi}_S = (0.2644443138466^{\mathbf{706}}_{\mathbf{694}}; 0.093168811815\mathbf{68}^{\mathbf{200}}_{\mathbf{122}})$.

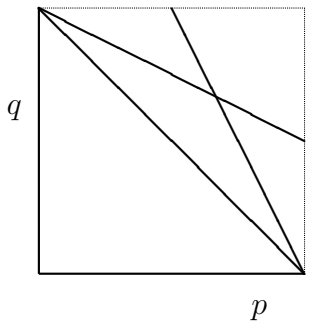


Figure 4.4: The unit square denotes the parameter space for $(p, q)$. Thick lines denote values for which a division by zero error will occur in calculating the gradient.

### 4.5.5 Multinomial Example Continued

The multinomial example considered above is presented again in two different ways to illustrate other ways to employ interval analysis to good advantage.

#### Intervalized Scalar EM Algorithm

For the incomplete-data problem, it is easy to show that

$$(X_2|X_1 + X_2 = 125) \sim Binomial\left(125, \frac{\frac{1}{4}p}{\frac{1}{2} + \frac{1}{4}p}\right)$$

and the E-Step in the usual scalar EM algorithm becomes $x_{1,k} = 125(\frac{1}{2})/(\frac{1}{2} + \frac{1}{4}p)$ and $x_{2,k} = 125(\frac{1}{4}p)/(\frac{1}{2} + \frac{1}{4}p)$. From the complete-data likelihood of $p$,

$$f(p|\mathbf{x}) \propto \left(\frac{1}{2}\right)^{x_{1,k}} \left(\frac{1}{4}p\right)^{x_{2,k}} \left(\frac{1}{4} - \frac{1}{4}p\right)^{x_3+x_4} \left(\frac{1}{4}p\right)^{x_5}$$

the M-Step in the usual scalar EM algorithm is:

$$p_k = \frac{x_{2,k} + x_4}{x_{2,k} + x_3 + x_4 + x_5} = \frac{x_{2,k} + 34}{x_{2,k} + 72}.$$

Using $p_0 = 0.5$ as a starting value and using a convergence tolerance of $\epsilon = 10^{-7}$, the (scalar real) EM algorithm yields:

```
Epsilon:    1e-07
Initial p: 0.5
Iter      p         x2
1     0.608247   25
2     0.624321   29.1502
3     0.626489   29.7373
4     0.626777   29.8159
```

```
5     0.626816   29.8263
6     0.626821   29.8277
7     0.626821   29.8279
8     0.626821   29.8279
```

The algorithm converges at the specified tolerance after 8 iterations. In this case, the starting value of 0.5 for $p$ was chosen simply because 0.5 lies exactly halfway between 0 and 1, which define the bounds for possible starting values. A questioning user may well wonder what results would be obtained for different starting values and how the steps of convergence might change. Interval analysis can be used to answer those questions.

This example can easily be programmed in interval arithmetic, though with a slight modification. Because of the dependency problem, narrower interval enclosures of computed values are more likely to be obtained if each variable appears only once in a calculation. The iterates in the EM algorithm for this particular example are therefore written equivalently as:

$$x_{2,k}^I = \frac{125.0}{2/p_k^I + 1} \quad \text{and} \quad p_k^I = 1 - \frac{38}{x_{2,k}^I + 72}$$

The convergence tolerance remains the same as above, but now $p_0^I = [\delta, 1]$ where $\delta$ is a small machine number greater than zero. The scalar EM algorithm using interval arithmetic produces the following output:

```
Epsilon:   1e-07
Initial p: [4.94066e-324,1]
i           p                  x2
1   [0.472222,0.665689]   [0,41.6667]
2   [0.603656,0.631839]   [23.8764,31.2156]
3   [0.623692,0.627485]   [28.9812,30.0094]
4   [0.626405,0.626910]   [29.7144,29.852]
5   [0.626766,0.626833]   [29.8129,29.8311]
6   [0.626814,0.626823]   [29.8259,29.8284]
7   [0.626821,0.626822]   [29.8277,29.828]
8   [0.626821,0.626822]   [29.8279,29.828]
9   [0.626821,0.626822]   [29.8279,29.8279]
```

It is now easy to see that all scalar starting values of $p_0$ in the scalar EM algorithm will lead to the same point of convergence, and furthermore the number of iterations to convergence is not highly dependent on the starting value of $p$. The use of interval arithmetic has allowed the user to consider all possible values of the input parameter at once. This will not always be the case, but is a beneficial feature for the cases where it is possible.

### Interval Global Optimization

Since interval analysis is scarcely known in the statistical literature, it will also be useful to mention another optimization method here. In the multinomial example, there is no

missing data and the loglikelihood in this case is given by

$$\log f(p|\mathbf{x}) \propto 125 * \log\left(\frac{2+p}{4}\right) + 38 * \log\left(\frac{1-p}{4}\right) + 34 * \log\left(\frac{p}{4}\right).$$

This can be viewed as an ordinary function to optimize, a task for which interval global optimization is well suited. Hansen (1992) is one of several monographs on this topic. Using the PROFIL software (or a similar package) with an initial interval of $p = [0 + \epsilon, 1 - \epsilon]$, a guaranteed enclosure of a stationary point is returned to the user.

## 4.6 Conclusions

Interval analysis first gained noticeable development in the 1960s from the work of R. E. Moore. Interval analysis has a fairly extensive literature in some areas, e.g. global optimization, but has seen little development in statistical settings. This paper takes a step at remediating the current state of knowledge by using interval analysis together with ideas from the EM algorithm. The resulting method is capable of finding multiple stationary points of a loglikelihood to a high degree of accuracy. The EM algorithm cannot be relied upon to do this. Unlike other algorithms for optimization, the method retains the ability of the EM algorithm to handle missing-data problems.

# Chapter 5

# GENERAL CONCLUSIONS

This dissertation has explored various ways in which interval analysis can be utilized in statistical computing. Interval analysis uses intervals instead of real numbers as the basic units of calculation. Interval analysis provides a means to evaluate the range of a function over a given domain. Using this capability, it is possible to obtain numerical answers which are guaranteed to be correct to a certain level of accuracy and to contain the true result. Interval analysis is also able to guarantee finding global optimizers within an initial box.

There are two broad topics covered in this dissertation as related to interval analysis. Several sections of the dissertation are focused on obtaining numerical values which are of a guaranteed accuracy. The application to statistics is in the computation of critical points and tail probabilities of several statistical distributions. For a bivariate chi-square distribution considered in one section, a series expansion is used together with a bound on the truncation error for the series. By evaluating the finite series with intervals and evaluating the truncation error with intervals, an interval enclosure of the true probability is obtained. To find critical points of a distribution, one approach requires solving for the root of an equation. Due to the complicated form of the equation involved, the use of derivatives (as in the Newton-Raphson algorithm) is avoided in favor of a derivative-free root-finding method. An interval secant algorithm (with Illinois modification) is developed and used for finding critical points of the distribution. A bivariate $F$ distribution is also considered using similar techniques with the added complication that each term in the series involves the calculation of an interval Gaussian quadrature rule for evaluating a numerical integral. The methods prove successful for computing guaranteed enclosures of the probabilities to several decimal places (at a minimum). In some cases the guarantee extends to more than a dozen accurate digits. The guaranteed values allowed for the discovery of errors in earlier published tables.

A third section of the dissertation considers the global optimization capabilities of interval analysis. The EM algorithm is widely used in statistics for estimating parameters of a model when data is missing. The original EM paper by Dempster et al. (1977) has been cited in more than 2000 papers since its publication and continues to be an active area of research. Much of the research about the EM algorithm has explored ways to speed up the rate of convergence to a stationary point.

While enjoying enormous popularity, the EM algorithm, like many optimization methods, generally only converges to a stationary point (not necessarily the global optimum). Using special properties of the EM algorithm, an interval enclosure of the gradient of the loglikelihood is derived. The enclosure of the gradient is evaluated over an interval region. Regions where the enclosure of the gradient does not contain zero are therefore known to contain no stationary points. By beginning with an initial region and repeatedly subdividing the region into ever smaller pieces, regions which are known to not contain a stationary point are eliminated from consideration. Any stationary points of the loglikelihood are located in the union of the undiscarded regions. If the initial region is large enough, all stationary points of the loglikelihood will be found. No other known method is capable of achieving the same result.

Interval analysis has been demonstrated to be an effective tool in statistical computing. The unique opportunities and challenges associated with interval analysis promise both opportunities for future research and future rewards.

# Bibliography


Amos, D. E. and Bulgren, W. G. (1972). Computation of a Multivariate F Distribution. *Mathematics of Computation*, 26(117):255–264. 30

ANSI/IEEE (1985). IEEE Standard for Binary Floating-Point Arithmetic. ANSI/IEEE Standard 754-1985. 5

ANSI/IEEE (1987). A Radix-Independent Standard for Floating-Point Arithmetic. ANSI/IEEE Standard 854-1987. 5, 28

Arslan, O., Constable, P. D. L., and Kent, J. T. (1993). Domains of Convergence for the EM Algorithm: a Cautionary Tale in a Location Estimation Problem. *Statistical Computing*, 3:103–108. 42

Bohachevsky, I. O., Johnson, M. E., and Stein, M. L. (1986). Generalized Simulated Annealing for Function Optimization. *Technometrics*, 28:209–217. 10

Bohlender, G. (1990). *Computer Arithmetic and Self-Validating Numerical Methods*, chapter What Do We Need Beyond IEEE Arithmetic?, pages 1–32. Academic Press, Inc., New York. 1

Brane, K. (1988). Standard Functions for Real and Complex Point and Interval Arguments with Dynamic Accuracy. In Kulisch, U. and Stetter, H. J., editors, *Scientific Computation with Automatic Result Verification*, pages 159–184. Springer-Verlag, New York. 7

Cody, W. J. (1988). Floating-Point Standards–Theory and Practice. In Moore, R. E., editor, *Reliability in Computing*, pages 99–107. Academic Press, New York. 5

Corliss, G. F. (1988). Reliability in Computing. In Moore, R. E., editor, *Reliability in Computing*, chapter Applications of Differentiation Arithmetic, pages 127–148. Academic Press, New York. 7

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood From Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B. Methodological*, 39:1–22. 35, 36, 41, 49

Dutt, J. E. and Soms, A. P. (1976). An Integral Representation Technique for Calculating General Multivariate Probabilities with an Application to Multivariate $\chi^2$. *Communications in Statistics–Theory and Methods*, 5:377–388. 24

Gay, D. M. (1988). Interval Least Squares–a Diagnostic Tool. In Moore, R. E., editor, *Reliability in Computing*, pages 183–205. Academic Press, New York. 9

Gessner, M. (1992). Self-Validating Computations of Standard Normal and Incomplete Gamma Probabilities. Master's thesis, Iowa State University. 23

Gunst, R. F. (1973). On Computing Critical Points for a Bivariate Chi-Square Random Variable. *Communications in Statistics*, 2:221–224. 23, 24

Gunst, R. F. and Webster, J. T. (1973). Density Functions of the Bivariate Chi-Square Distribution. *J. Statist. Comput. Simul.*, 2:275–288. 16, 19, 20, 23

Gupta, S. and Sobel, M. (1962). On the Smallest of Several Correlated F Statistics. *Biometrika*, 49:509–523. 13

Hamdy, H., Son, M., and Al-Mahmeed, M. (1988). On the Distribution of [a] Bivariate F-Variable. *Computational Statistics & Data Analysis*, 6:157–164. 12

Hammer, R., Hocks, M., Kulisch, U., and Ratz, D. (1993a). *Numerical Toolbox for Verified Computing I*. Springer-Verlag, New York. 2

Hammer, R., Neaga, M., and Ratz, D. (1993b). PASCAL-XSC: New Concepts for Scientifc Computation and Numerical Data Processing. In Adams, E. and Kulisch, U., editors, *Scientific Computing with Automatic Result Verification*, pages 15–44. Academic Press, New York. 6

Hansen, E. (1988). *Reliability in Computing*, chapter An Overview of Global Optimization Using Interval Analysis, pages 289–307. Academic Press, New York. 10

Hansen, E. (1992). *Global Optimization Using Interval Analysis*. Marcel Dekker, New York. 34, 39, 48

Hansen, E. R. (1997). Sharpness in Interval Computations. *Reliable Computing*, 3:17–29. 4, 34

Jensen, D. R. and Howe, R. B. (1968). Probability Content of the Bivariate Chi-Square Distribution Over Rectangular Regions. *Virginia Journal of Science*, 19:233–239. 16, 24

Jensen, D. R. and Jones, M. Q. (1969). Simultaneous Confidence Intervals for Variances. *Journal of the American Statistical Association*, 64:324–332. 16

Kearfott, R. B. (1996a). *Applications of Interval Computations*, chapter A Review of Techniques in the Verified Solution of Constrained Global Optimization Problems, pages 23–59. Kluwer Academic Publishers, Boston. 6, 10

Kearfott, R. B. (1996b). *Rigorous Global Search: Continuous Problems*. Kluwer Academic Publishers, Boston. 35

Kelch, R. (1993). Numerical Quadrature by Extrapolation with Automatic Result Verification. In Adams, E. and Kulisch, U., editors, *Scientific Computing with Automatic Result Verification*, pages 143–185. Academic Press, New York. 30

Kennedy, W. and Gentle, J. (1980). *Statistical Computing*. Marcel Dekker, New York. 10

Knüppel, O. (1993a). BIAS – Basic Interval Arithmetic Subroutines. Technical Report 93.3, Informationstechnik, Technische Uni. Hamburg–Harburg. 23, 28

Knüppel, O. (1993b). PROFIL – Programmer's Runtime Optimized Fast Interval Library. Technical Report 93.4, Informationstechnik, Technische Uni. Hamburg–Harburg. 23, 28

Knüppel, O. (1993c). PROFIL – Programmer's Runtime Optimized Fast Interval Library. Technical Report 93.4, Informationstechnik, Technische Uni. Hamburg–Harburg. 35

Krämer, W. (1988). Inverse Standard Functions for Real and Complex Point and Interval Arguments with Dynamic Accuracy. In Kulisch, U. and Stetter, H. J., editors, *Scientific Computation with Automatic Result Verification*, pages 185–212. Springer-Verlag, New York. 7

Krämer, W. (1993). Multiple-Precision Computations with Result Verification. In *Scientific Computation with Automatic Result Verification*. Academic Press, New York. 6

Krishnaiah, P. R. (1965). On the Simultaneous ANOVA and MANOVA Tests. *Ann. Inst. Statist. Math.*, 17:35–53. 29

Krishnaiah, P. R. (1980a). *Handbook of Statistics, Vol. 1*, chapter Computations of Some Multivariate Distributions, pages 745–971. North-Holland Publishing Company, New York. 19, 23, 24

Krishnaiah, P. R. (1980b). *Handbook of Statistics, Vol. 1*, chapter Computations of Some Multivariate Distributions, pages 745–971. North-Holland Publishing Company, New York. 27, 28, 29, 31

Lawo, C. (1993). C-XSC: A Programming Environment for Verified Scientific Computing and Numerical Data Processing. In Adams, E. and Kulisch, U., editors, *Scientific Computing with Automatic Result Verification*, pages 71–102. Academic Press, New York. 6

McLachlan, G. J. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. John Wiley & Sons, New York. 42, 45

Metzger, M. (1988). Reliability in Computing. In Moore, R. E., editor, *Reliability in Computing*, chapter FORTRAN-SC: A FORTRAN Extension for Engineering / Scientific Computation with Access to ACRITH, pages 63–79. Academic Press, New York. 6

Montgomery, D. C. (1991). *Design and Analysis of Experiments*. John Wiley & Sons, New York, third ed. edition. pages 148–151. 10

Moore, R. E. (1966). *Interval Analysis*. Prentice-Hall, Englewood Cliffs, N.J. 17, 28

Moore, R. E. (1979). *Methods and Applications of Interval Analysis*. SIAM, Philadelphia. 2, 17, 20, 31, 34

Ralston, M. L. and Jennrich, R. I. (1978). Dud, a Derivative-free Algorithm for Nonlinear Least Squares. *Technometrics*, 20:7–14. 10

Ratscheck, H. and Rokne, J. (1984). *Computer Methods for the Range of Functions*. Ellis Horwood Limited, Chichester, England. 3

Schuurmann, F. J., Krishnaiah, P. R., and Chattopadhyay, A. K. (1975). Tables for a Multivariate F Distribution. *Sankhya*, 37:308–331. 27, 28, 29, 31

Smirnov, N. (1948). Table for Estimating the Goodness of Fit of Empirical Distributions. *Annals of Mathematical Statistics*, 19:279–281. 12

Thisted, R. A. (1988). *Elements of Statistical Computing*. Chapman and Hall, New York. 10, 21, 44

Tuprah, K. and Woodall, W. H. (1986). Bivariate Dispersion Quality Control Charts. *Communications in Statistics – Simulation and Computing*, 15:505–522. 16

Walster, G. W. (1988). Reliability in Computing. In Moore, R. E., editor, *Reliability in Computing*, chapter Philosophy and Practicalities of Interval Arithmetic, pages 309–323. Academic Press, New York. 4, 8

Walter, W. (1988). FORTRAN-SC: A FORTRAN extension for engineering/scientific computation with access to ACRITH. In Moore, R. E., editor, *Reliability in Computing*, pages 43–62. Academic Press, New York. 6

Walter, W. V. (1993). ACRITH-XSC: A Fortran-like Language for Verified Scientific Computing. In Adams, E. and Kulisch, U., editors, *Scientific Computing with Automatic Result Verification*, pages 45–70. Academic Press, New York. 6

Wang, M. C. and Kennedy, W. J. (1990). Comparison of Algorithms for Bivariate Normal Probability Over a Rectangle Based on Self-Validated Results From Interval Analysis. *Journal of Statistical Computation and Simulation*, 37:13–25. 8

Wang, M. C. and Kennedy, W. J. (1992). A Numerical Method for Accurately Approximating Multivariate Normal Probabilities. *Computational Statistics & Data Analysis*, 13:197–210. 8

Wang, M. C. and Kennedy, W. J. (1994a). Self-Validating Computations of Probabilities for Selected Central and Noncentral Univariate Probability Functions. *Journal of the American Statistical Association*, 89:878–887. 9

Wang, M. C. and Kennedy, W. J. (1994b). Self-Validating Computations of Probabilities for Selected Central and Noncentral Univariate Probability Functions. *Journal of the American Statistical Association*, 89:878–887. 17, 23, 28, 30

Wang, M. C. and Kennedy, W. J. (1995). A Self-Validating Numerical Method for Computation of Central and Non-Central F Probabilities and Percentiles. *Statistics & Computing*, 5:155–163. 9

Wang, O. (1994). Applications ofNumerical Interval Analysis to Obtain Self-Validating Results in a Massively Parallel Computing Environment. In *ASAProStCp*. 9, 10

Wright, K. and Kennedy, W. (1999). Self-Validated Computations for the Probabilities of the Central Bivariate Chi-Square Distribution. Submitted for publication. 28, 31

Wu, C. F. J. (1982). On the Convergence Properties of the EM Algorithm. *The Annals of Statistics*, 11(1):95–103. 33

# ACKNOWLEDGEMENTS

I would like to extend my thanks to:

My Ph.D. advisor, William Kennedy, for helpful discussions and direction, words of encouragement and financial support for a research assistantship via National Science Foundation Grant DMS-9500831.

The members of my committee, Mervyn Marasinge, Yasuo Amemiya, Steve Vardeman, and Dale Grosvnor, for helpful suggestions during my preliminary oral and final defense.

Deanne Reber and Juan Jose Goyeneche for proofreading.

Kevin Dodd, Juan Jose Goyeneche, Deanne Reber, Pam Abbitt and Todd Krueger for help with computing issues on everything from TEXto C.

Jon Rokne for personal correspondence regarding complex interval arithmetic.

Robert Dodier and Clint Cummins for correspondence about maximization problems with sensitive starting values.

Sara Nusser for allowing me to take time off from my research assistantship in the survey section.

The faculty of the statistics department at Iowa State University for providing me with an excellent education.