Special Functions

David Eberly
Geometric Tools, LLC
http://www.geometrictools.com/

Created: March 2, 1999
Last Modified: March 2, 2008

Contents

1 Discussion **2**

2 Gamma Function **2**

2.1 Incomplete Gamma Function 3

3 Error Function **3**

4 Modified Bessel Function **4**

1 Discussion

Although there are many special functions used in the mathematical world, the two I have used frequently are the *error function* and the *modified Bessel functions* of orders 0 and 1. In addition I have a couple other special functions, the logarithm of gamma and the incomplete gamma function, which are used for computing the error function. The class declaration for special functions is

```
class mgcSpecialFunction
{
public:
    mgcSpecialFunction () {;}

    // gamma and related functions
    float LogGamma (float x);
    float Gamma (float x) { return exp(LogGamma(x)); }
    float IncompleteGammaS (float a, float x);    // series form
    float IncompleteGammaCF (float a, float x);   // continued fraction form
    float IncompleteGamma (float a, float x);

    // error functions
    float Erf_NRC (float x);  // Numerical Recipes in C, error function
    float Erf (float x);   // polynomial approximation to erf(x)
    float Erfc (float x);  // complementary error function, erfc(x) = 1-erf(x)

    // modified Bessel functions of order 0 and 1
    float ModBessel0 (float z);
    float ModBessel1 (float z);
};
```

As new special functions are needed, they can be added to this class.

2 Gamma Function

The *gamma function* is defined by

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t}\, dt.$$

When z is an integer, then $\Gamma(n+1) = n!$, the factorial function. For large z this function can be quite large, so it is common to work with its logarithm instead. We can use the formula

$$\Gamma(z+1) = \sqrt{2\pi}(z+\alpha+1/2)^{z+1/2} e^{-(z+\gamma+1/2)} \left(\sum_{k=0}^{N} \frac{c_k}{z+k} + \varepsilon \right)$$

for $z > 0$ and where α and N are user–selected. For $\alpha = 5$ and $N = 6$, it can be shown that $|\varepsilon| < 2 \times 10^{-7}$. The implementation of the logarithm of this function is `float LogGamma (float x)` and the gamma function is obtained from it by `float Gamma (float x)`.

2.1 Incomplete Gamma Function

An *incomplete gamma function* is defined by

$$P(a,x) = \frac{\gamma(a,x)}{\Gamma(a)} = \frac{1}{\Gamma(a)} \int_0^x e^{-t} t^{a-1}\, dt$$

for $a > 0$. This equation also defines the function $\gamma(a,x)$. A series representation for $\gamma(a,x)$ is

$$\gamma(a,x) = e^{-x} x^a \sum_{n=0}^{\infty} \frac{\Gamma(a)}{\Gamma(a+1+n)} x^n.$$

The series converges rapidly for $x \le a+1$. Another incomplete gamma function is defined by

$$Q(a,x) = 1 - P(a,x) = \frac{\Gamma(a,x)}{\Gamma(a)}$$

for $a > 0$. This equation also defines the function $\Gamma(a,x)$. A continued fraction representation for $\Gamma(a,x)$ is

$$\Gamma(a,x) = e^{-x} x^a \left(\frac{1}{x+} \frac{1-a}{1+} \frac{1}{x+} \frac{2-a}{1+} \frac{2}{x+} \cdots \right).$$

The continued fraction converges rapidly for $x \ge a+1$. An example of expanding a continued fraction is:

$$\frac{n_1}{d_1+} \frac{n_2}{d_2+} \frac{n_3}{d_3} = \frac{n_1}{d_1 + \frac{n_2}{d_2 + \frac{n_3}{d_3}}}.$$

The Numerical Recipes in C routines `gammp` (evaluates $P(a,x)$), `gammq` (evaluates $Q(a,x)$), `gser` (series computation of $\gamma(a,x)$), and `gcf` (continued fraction computation of $\Gamma(a,x)$), are implemented (with different organization) as special functions `IncompleteGammaS` (series), `IncompleteGammaCF` (continued fractions), and `IncompleteGamma` ($P(a,x)$).

3 Error Function

The *error function* is defined by

$$\operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt.$$

The *complementary error function* is

$$\operatorname{erfc}(x) = 1 - \operatorname{erf}(x).$$

The code I use to calculate the complementary error function comes from Numerical Recipes in C. The formula below comes from Chebyshev fitting of the function.

$$\begin{aligned} t &= \tfrac{1}{1+|x|/2} \\ p(t) &= \textstyle\sum_{k=0}^{9} a_k t^k \\ (1+\varepsilon)\operatorname{erfc}(x) &= t \exp(-z^2 + p(t)) \end{aligned}$$

where

$$a_0 = -1.26551223, a_1 = +1.00002368, a_2 = +0.37409196, a_3 = +0.09678418,$$
$$a_4 = -0.18628806, a_5 = +0.27886807, a_6 = -1.13520398, a_7 = +1.48851587.$$
$$a_8 = -0.82215223, a_9 = +0.17087277, |\varepsilon| < 1.2 \times 10^{-7}$$

The implementation of this is `Erf`. The alternate routine `Erf_NRC` uses the equivalence

$$\mathrm{erf}(x) = P(0.5, x^2), \quad x \geq 0,$$

where P is the incomplete gamma function.

An alternate way to compute the error function is through the use of incomplete gamma functions. In doing so, some other special functions must be implemented.

4 Modified Bessel Function

The modified Bessel function of order zero is an even function given by

$$I_0(z) = \frac{1}{\pi} \int_0^\pi \exp(z \cos\theta)\, d\theta = \sum_{m=0}^{\infty} \frac{(z^2/4)^m}{(m!)^2}.$$

This function is also a solution to the second–order linear differential equation: $z^2 w'' + z w' - z^2 w = 0$. The modified Bessel function of order one is an odd function given by

$$I_1(z) = I_0'(z) = \frac{1}{\pi} \int_0^\pi \cos\theta \exp(z \cos\theta)\, d\theta$$

and it is a solution to the differential equation: $z^2 w'' + z w' - (z^2 + 1) w = 0$. We can rewrite

$$F(R) = 2\pi \exp\left(-\frac{R^2 + 1}{2\rho^2}\right) \left[I_0(R/\rho^2) - R I_1(R/\rho^2)\right].$$

Polynomial approximations to these functions are given below:

$$I_0(z) = \sum_{k=0}^{6} a_k (z/3.75)^{2k} + \varepsilon, \quad 0 \leq z \leq 3.75$$

where

$$a_0 = 1, a_1 = 3.5156229, a_2 = 3.0899424, a_3 = 1.2067492,$$
$$a_4 = 0.2659732, a_5 = 0.0360768, a_6 = 0.0045813, |\varepsilon| < 1.6 \times 10^{-7}$$

and

$$z^{1/2} e^{-z} I_0(z) = \sum_{k=0}^{8} b_k (z/3.75)^{-k} + \varepsilon, \quad z \geq 3.75$$

where

$$b_0 = +0.39894228, b_1 = +0.01328592, b_2 = +0.00225319, b_3 = -0.00157565,$$
$$b_4 = +0.00916281, b_5 = -0.02057706, b_6 = +0.02635537, b_7 = -0.01647633, b_8 = +0.00392377,$$
$$|\varepsilon| < 1.9 \times 10^{-7}$$

and

$$z^{-1} I_1(z) = \sum_{k=0}^{6} c_k (z/3.75)^{2k} + \varepsilon, \quad 0 \leq z \leq 3.75$$

where

$$c_0 = 0.5, c_1 = 0.87890549, c_2 = 0.51498869, c_3 = 0.15084934,$$
$$c_4 = 0.02658733, c_5 = 0.00301532, c_6 = 0.00032411, |\varepsilon| < 8 \times 10^{-9}$$

and

$$z^{1/2} e^{-z} I_1(z) = \sum_{k=0}^{8} d_k (z/3.75)^{-k} + \varepsilon, \quad z \geq 3.75$$

where

$$d_0 = +0.39894228, d_1 = -0.03988024, d_2 = -0.00362018, d_3 = +0.00163801,$$
$$d_4 = -0.01031555, d_5 = +0.02282967, d_6 = -0.02895312, d_7 = +0.01787654, d_8 = -0.00420059,$$
$$|\varepsilon| < 2.2 \times 10^{-7}$$

The formulas can be found in Abramowitz and Stegun, *Handbook of Mathematical Functions*. No derivation is given there. However, the idea is that for inputs close to zero, the function can be fit to any desired accuracy with a polynomial. For inputs far from zero, the function can be fit to any desired accuracy with a rational function. (Of course, the trick is choosing the forms of these functions and choosing parameters of these functions to guarantee the accuracy. Probably a least squares approach can be used which minimizes the maximum difference between function and approximation.)