Special Functions in Mathematics

With Scipy

Authored by : **B D Satish**

Last revised on 06-Nov-2007

Contents

1 Introduction **2**

2 Airy Functions **2**
- 2.1 Ai(z), Bi(z) . . . 2
- 2.2 Zeros . . . 2

3 Elliptic Functions and Integrals **3**
- 3.1 $F(\phi \mid m), K(m)$ . . . 3
- 3.2 $E(\phi \mid m), E(m)$ . . . 3
- 3.3 $\mathrm{sn}(u \mid m), \mathrm{cn}(u \mid m), \mathrm{dn}(u \mid m)$ . . . 3

4 Bessel Functions **3**

5 Spherical Bessel Functions **4**
- 5.1 $j_n(z), y_n(z)$ . . . 4
- 5.2 $i_n(z), k_n(z)$ . . . 4

6 Struve Functions **5**
- 6.1 $\mathbf{H}_\nu(\mathrm{z})$ . . . 5
- 6.2 $\mathbf{L}_\nu(\mathrm{z})$ . . . 5

7 Gamma and Related Functions **5**
- 7.1 $\Gamma(z)$ . . . 5
- 7.2 $P(a,x), Q(a,x)$ . . . 6
- 7.3 $B(a,b), I_x(a,b)$ . . . 6
- 7.4 $\psi(z), \psi^{(n)}(z)$ . . . 6

8 Error Function and Fresnel Integtrals **7**
- 8.1 erf z, erfc z . . . 7
- 8.2 $C(z)$, $S(z)$ . . . 7
- 8.3 $F_+(z), K_+(z), F_-(z), K_-(z)$ . . . 7

9 Hypergeometric Functions **7**
- 9.1 ${}_1F_1$, ${}_0F_1, U$ . . . 7
- 9.2 ${}_2F_1$, ${}_2F_0$, ${}_1F_2$, ${}_3F_0$ . . . 8

10 Orthogonal Polynomials **8**
- 10.1 $P_n(x), P_n^*(x)$ . . . 9
- 10.2 $T_n(x), C_n(x), T_n^*(x)$ . . . 9
- 10.3 $U_n(x), S_n(x), U_n^*(x)$ . . . 9
- 10.4 $P_n^{(a,b)}(x), G_n(p,q,x)$ . . . 10
- 10.5 $L_n(x), L_n^{(a)}(x)$ . . . 10
- 10.6 $C_n^{(a)}(x)$ . . . 10
- 10.7 $H_n(x), He_n(x)$ . . . 11

11 Exponential Integrals **11**
- 11.1 $Ei(x), E_n(x), E_1(x), \zeta(s), Li_2(z)$ . . . 11

12 Zeta and Related Functions **11**
- 12.1 $\zeta(s,a), \zeta(s), \mathrm{Li}_2(z)$ . . . 11

13 Sine and Cosine Integrals **11**
- 13.1 $\mathrm{Si}(z), \mathrm{Ci}(z), \mathrm{Shi}(z), \mathrm{Chi}(z)$ . . . 11

14 License **12**

1 Introduction

I always had a fascination about the so-called "Special functions" in mathematics. Most of them arise as solutions to differential equations. They have a wide range of properties which are very interesting

In this article, I've collected the *definitions* of these functions. Usually it'll be an integral or a power series. I recommend the Scipy (http://www.scipy.org) module which is written in Python (http://www.python.org) for evaluating these functions. Add the following line to your python program:

```
from scipy.special import *
```

and you are ready to go. You'll find all the Python commands in `monospace font`.

The entire document was written in LaTeX and converted to PDF using pdfLaTeX. See the License (section 14) before you proceed.

2 Airy Functions

There are two kinds of **Airy** functions.

2.1 Ai(z), Bi(z)

The function `airy(z)` returns four values respectively as $\mathrm{Ai}(z)$, $\mathrm{Ai}'(z)$, $\mathrm{Bi}(z)$ and $\mathrm{Bi}'(z)$ in accordance with the following definitions:–

$$\begin{aligned} \zeta &= \frac{2}{3} z^{\frac{3}{2}} \\ \mathrm{Ai}(z) &= \pi^{-1} \sqrt{\left(\frac{z}{3}\right)} K_{\frac{1}{3}}(\zeta) \\ &= \frac{1}{3}\sqrt{z}\ [\ I_{-\frac{1}{3}}(\zeta) - I_{\frac{1}{3}}(\zeta)\ ] \end{aligned} \tag{1}$$

$$\mathrm{Bi}(z) \quad = \quad \sqrt{\frac{z}{3}}\ [\ I_{-\frac{1}{3}}(\zeta) + I_{\frac{1}{3}}(\zeta)\ ] \tag{2}$$

$$\begin{aligned} \mathrm{Ai}'(z) &= -\pi^{-1} \frac{z}{\sqrt{3}}\ K_{\frac{2}{3}}(\zeta) \\ \mathrm{Bi}'(z) &= \frac{z}{\sqrt{3}}\ [\ I_{-\frac{2}{3}}(\zeta) + I_{\frac{2}{3}}(\zeta)\ ] \end{aligned} \tag{3}$$

Similarly `airye(z)` returns exponentially-scaled Airy functions in the order $\mathrm{Ai}(z)\exp\left(\frac{2}{3}z\sqrt{z}\right)$, $\mathrm{Ai}'(z)\exp\left(\frac{2}{3}z\sqrt{z}\right)$, $\mathrm{Bi}(z)\exp\left(-\Re\left\{\frac{2}{3}z\sqrt{z}\right\}\right)$ and $\mathrm{Bi}'(z)\exp\left(-\Re\left\{\frac{2}{3}z\sqrt{z}\right\}\right)$

2.2 Zeros

Additionally, to understand the zeros of Airy functions, we define $\mathrm{Ai}(z)$ and $\mathrm{Bi}(z)$ in terms of $f(z)$ and $g(z)$ as follows:

$$\begin{aligned} \mathrm{Ai}(z) &= c_1 f(z) - c_2 g(z) \\ \mathrm{Bi}(z) &= \sqrt{3}\ [c_1 f(z) + c_2 g(z)\ ] \end{aligned} \tag{4}$$

where

$$\begin{aligned} c_1 &= \mathrm{Ai}(0) = 3^{-\frac{2}{3}}/\Gamma(2/3) \\ c_2 &= -\mathrm{Ai}'(0) = 3^{-\frac{1}{3}}/\Gamma(2/3) \\ f(z) &= \sum_{k=0}^{\infty} 3^k \left(\frac{1}{3}\right)_k \frac{z^{3k}}{(3k)!} \\ g(z) &= \sum_{k=0}^{\infty} 3^k \left(\frac{2}{3}\right)_k \frac{z^{3k+1}}{(3k+1)!} \end{aligned} \tag{5}$$

Denote the n^{th} zero of $\mathrm{Ai}(z)$, $\mathrm{Ai}'(z)$, $\mathrm{Bi}(z)$ and $\mathrm{Bi}'(z)$ respectively as a_n, a'_n, b_n and b'_n , where

$$\begin{aligned} a_n &= -f[\ 3\pi(4n-1)/8\ ] \\ a'_n &= -g[\ 3\pi(4n-3)/8\ ] \\ \mathrm{Ai}(a'_n) &= (-1)^{n-1} f_1[\ 3\pi(4n-1)/8\ ] \\ \mathrm{Ai}'(a_n) &= (-1)^{n-1} g_1[\ 3\pi(4n-3)/8\ ] \\ b_n &= -f[\ 3\pi(4n-3)/8\ ] \\ b'_n &= -g[\ 3\pi(4n-1)/8\ ] \\ \mathrm{Bi}(b'_n) &= (-1)^{n-1} g_1[\ 3\pi(4n-1)/8\ ] \\ \mathrm{Bi}'(b_n) &= (-1)^{n-1} f_1[\ 3\pi(4n-3)/8\ ] \end{aligned} \tag{6}$$

The function `ai_zeros(n)` returns a tuple of four arrays. Each array is of length n containing the first n values of a_n, a'_n, $\mathrm{Ai}(a'_n)$ and $\mathrm{Ai}'(a_n)$. Similarly `bi_zeros(n)` also returns a tuple of four arrays. Each array is of length n containing the first n values of b_n, b'_n, $\mathrm{Bi}(b'_n)$ and $\mathrm{Bi}'(b_n)$.

3 Elliptic Functions and Integrals

3.1 $F(\phi \,|\, m), K(m)$

The **incomplete elliptic integral** of first kind of parameter m and amplitude ϕ is given by

$$u = F(\phi \,|\, m) = \int_0^{\phi} \frac{1}{\sqrt{1 - m\sin^2\theta}}\, d\theta \qquad (7)$$

where $-\pi/2 < \phi < \pi/2$ and $0 \leq m < 1$. When $\phi = \pi/2$, we call it the **complete elliptic integral** of first kind:

$$K(m) = \int_0^{\frac{\pi}{2}} \frac{1}{\sqrt{1 - m\sin^2\theta}}\, d\theta \qquad (8)$$

The functions `ellipk(m)` and `ellipkinc(phi,m)` return $K(m)$ and $F(\phi \mid m)$ repectively.

3.2 $E(\phi \,|\, m), E(m)$

The incomplete elliptic integral of second kind is given by

$$E(\phi \,|\, m) = \int_0^{\phi} \sqrt{1 - m\sin^2\theta}\, d\theta \qquad (9)$$

The complete elliptic integral of second kind is, simply, $E(\pi/2 \,|\, m)$, often denoted as $E(m)$. The functions `ellipe(m)` and `ellipeinc(phi,m)` return $E(m)$ and $E(\phi \,|\, m)$ repectively.

3.3 $\mathrm{sn}(u \,|\, m), \mathrm{cn}(u \,|\, m), \mathrm{dn}(u \,|\, m)$

The **Jacobian elliptic** *functions* are based on the elliptic *integrals* of first kind. The *amplitude* for Jacobi functions is the inverse of the elliptic integral of first kind.

If $u = F(\phi \,|\, m)$, then $\phi = \mathrm{am}(u \,|\, m)$

The Jacobi elliptic functions are:

$$\begin{aligned} \phi &= \mathrm{am}(u \,|\, m) \\ \mathrm{sn}(u \,|\, m) &= \sin(\phi) \\ \mathrm{cn}(u \,|\, m) &= \cos(\phi) \\ \mathrm{dn}(u \,|\, m) &= \sqrt{1 - m\sin^2\phi} = \Delta(\phi) \end{aligned} \qquad (10)$$

The `ellipj(u,m)` returns $\mathrm{sn}(u \mid m)$, $\mathrm{cn}(u \mid m)$, $\mathrm{dn}(u \,|\, m)$ and $\mathrm{am}(u \,|\, m)$ respectively.

4 Bessel Functions

In this section, we assume that n is an integer and ν is a real number. The Bessel functions $J_n(x)$ and $Y_n(x)$ are linerly independent solutions to the differential equation

$$x^2y'' + xy' + (z^2 - n^2)y = 0 \qquad (11)$$

$J_n(x)$ is often called the Bessel function of the first kind, or simply *the* Bessel function. $Y_n(x)$ is referred to as the Bessel function of second kind. The n is called the *order* of Bessel function. $J_n(x)$, when generalized to real order ν and complex argument z, is denoted as $J_\nu(z)$. Similarly, $Y_\nu(z)$ is an extension of $Y_n(x)$ for real order and complex input.

$$J_n(z) = \frac{1}{\pi}\int_0^{\pi} \cos(z\sin\theta - n\theta)\, d\theta \qquad (12)$$

$$\begin{aligned} J_\nu(z) = \frac{1}{\pi}\int_0^{\pi} \cos(z\sin\theta - \nu\theta)\, d\theta \\ -\frac{\sin(\nu\pi)}{\pi}\int_0^{\infty} \exp(-z\sinh t - \nu t)\, dt \end{aligned} \qquad (13)$$

$$J_\nu(z) = z^\nu \sum_{k=0}^{\infty} \frac{(-1)^k z^{2k}}{2^{2k+\nu}\; k!\; \Gamma(\nu + k + 1)} \qquad (14)$$

The Bessel function of second kind is defined as:

$$Y_\nu(z) = \frac{1}{\sin\nu\pi}\,[\; J_\nu(z)\cos\nu\pi - J_{-\nu}(z)\;] \qquad (15)$$

$$Y_n(x) = \lim_{\nu \to n} Y_\nu(x) \qquad (16)$$

The functions $\mathbf{jn}(n, x)$, $\mathbf{jv}(v, z)$, $\mathbf{yn}(n, x)$ and $\mathbf{jv}(v, z)$ respectively evaluate $J_n(x)$, $J_\nu(z)$, $Y_n(x)$ and $J_\nu(z)$.

The functions $\mathbf{jve}(v, z)$ and $\mathbf{yve}(v, z)$ scale the values of $\mathbf{jv}(v, z)$ and $\mathbf{yv}(v, z)$ by an exponential factor, $\exp(-|\Im\{z\}|)$, respectively.

The modified Bessel functions $I_\nu(z)$ and $K_\nu(z)$ are solutions to the differential equation

$$z^2y'' + zy' - (z^2 + n^2)y = 0 \qquad (17)$$

They are given by the following equations:

$$I_n(z) = \frac{1}{\pi}\int_0^\pi \exp(z\cos\theta)\cos(n\theta)\,d\theta \quad (18)$$

$$\begin{aligned} I_\nu(z) = {} & \frac{1}{\pi}\int_0^\pi \exp(z\cos\theta)\cos(\nu\theta)\,d\theta \\ & -\frac{\sin(\nu\pi)}{\pi}\int_0^\infty \exp(-z\cosh t - \nu t)\,dt \end{aligned} \quad (19)$$

$$I_\nu(z) = \sum_{k=0}^\infty \frac{z^{2k+\nu}}{2^{2k+\nu}\,k!\,\Gamma(k+\nu+1)} \quad (20)$$

The modified function $K_\nu(z)$ can be computed as:

$$K_\nu(z) = \frac{\pi}{2\sin\nu\pi}\left[\, I_{-\nu}(z) - I_\nu(z)\,\right] \quad (21)$$

$$K_n(x) = \lim_{\nu\to n} K_\nu(x) \quad (22)$$

The functions `iv(v,z)`, `kv(v,z)` and `kn(n,x)` respectively evaluate $I_\nu(z)$, $K_\nu(z)$ and $K_n(x)$.

The functions `ive(v,z)` and `kve(v,z)` scale the values of `iv(v,z)` and `kv(v,z)` by the exponential factors, $\exp(-|\Re\{z\}|)$ and $\exp(z)$ respectively.

We finally mention that there is a practical need for solutions of Bessel's equation that are complex even for real values of x. For this purpose the solutions

$$H_\nu^{(1)}(z) = J_\nu(z) + jY_\nu(z) \quad (23)$$

$$H_\nu^{(2)}(z) = J_\nu(z) - jY_\nu(z) \quad (24)$$

called **Hankel functions** of order ν. The functions `hankel1(v,z)` and `hankel2(v,z)` compute $H_\nu^{(1)}(z)$ and $H_\nu^{(2)}(z)$ respectively. The exponentially scaled Hankel functions are also available:

$$\texttt{hankel1e(v,z)} = \texttt{hankel1(v,z)} * \texttt{exp}(-1\texttt{j} * \texttt{z}) \quad (25)$$

$$\texttt{hankel2e(v,z)} = \texttt{hankel2(v,z)} * \texttt{exp}(+1\texttt{j} * \texttt{z}) \quad (26)$$

5 Spherical Bessel Functions

5.1 $j_n(z), y_n(z)$

The **spherical Bessel** functions solve the differntial equation:

$$z^2y'' + 2zy' + [z^2 - n(n+1)]y = 0 \quad (27)$$

for $n = 0, \pm1, \pm2, \ldots$. Particular solutions are the Spherical Bessel functions of the first kind:

$$j_n(z) = \sqrt{\frac{\pi}{2z}}J_{n+1/2}(z), \quad (28)$$

the Spherical Bessel functions of the second kind

$$y_n(z) = \sqrt{\frac{\pi}{2z}}Y_{n+1/2}(z), \quad (29)$$

and the Spherical Bessel functions of the third kind,

$$h_n^{(1)}(z) = j_n(z) + jy_n(z) = \sqrt{\frac{\pi}{2z}}H_{n+1/2}^{(1)}(z), \quad (30)$$

$$h_n^{(2)}(z) = j_n(z) - jy_n(z) = \sqrt{\frac{\pi}{2z}}H_{n+1/2}^{(2)}(z) \quad (31)$$

Let $f_n(z)$ be any of $j_n(z)$, $y_n(z)$, $h_n^{(1)}(z)$, $h_n^{(1)}(z)$,

$$\begin{aligned} \left(\frac{1}{z}\frac{d}{dz}\right)^m [z^{-n}f_n(z)] & = (-1)^m z^{-n-m} f_{n+m}(z) \\ \left(\frac{1}{z}\frac{d}{dz}\right)^m [z^{n+1}f_n(z)] & = z^{n-m+1} f_{n-m}(z) \end{aligned} \quad (32)$$

$n = 0, \pm1, \pm2, \ldots$; $m = 0, 1, 2, \ldots$

The function `sph_jn(n,z)` returns two arrays. The first array returns the values of $j_n(z)$ for all orders upto and including n, i.e., `n` $= 0, 1, 2, \ldots n$. The second array returns the first derivative of $j_n(z)$ for all orders upto and including n, i.e., `n` $= 0, 1, 2, \ldots n$. Similar comments apply for `sph_yn(n,z)` and $y_n(z)$. Another function, `sph_jnyn(n,z)`, combines the above two cases. It returns a tuple of four arrays: the first two are obtained from `sph_jn(n,z)` and the last two are obtained from `sph_yn(n,z)`.

5.2 $i_n(z), k_n(z)$

The **modified spherical Bessel** functions satisfy the ODE:

$$z^2y'' + 2zy' - [z^2 + n(n+1)]y = 0 \quad (33)$$

Particular solutions are the Modified Spherical Bessel functions of the first kind,

$$\begin{aligned} i_n(z) &= e^{-jn\frac{\pi}{2}} j_n(ze^{j\frac{\pi}{2}}); -\pi < \measuredangle z \le \frac{\pi}{2} \\ &= e^{j3n\frac{\pi}{2}} j_n(ze^{-j\frac{\pi}{2}}); \frac{\pi}{2} < \measuredangle z \le \pi \end{aligned} \quad (34)$$

of the second kind:

$$\begin{aligned} k_n(z) &= -\frac{\pi}{2} e^{jn\frac{\pi}{2}} h_n^{(1)}(ze^{j\frac{\pi}{2}}); -\pi < \measuredangle(z) \le \frac{\pi}{2} \\ &= -\frac{\pi}{2} e^{-jn\frac{\pi}{2}} h_n^{(2)}(ze^{-j\frac{\pi}{2}}); \frac{\pi}{2} < \measuredangle z \le \pi \end{aligned} \quad (35)$$

The function `sph_in(n,z)` returns two arrays. The first array returns the values of $i_n(z)$ for all orders upto and including n, i.e., $\mathtt{n} = 0, 1, 2, \ldots n$. The second array returns the first derivative of $i_n(z)$ for all orders upto and including n, i.e., $\mathtt{n} = 0, 1, 2, \ldots n$. Similar comments apply for `sph_kn(n,z)` and $k_n(z)$. Another function, `sph_inkn(n,z)`, combines the above two cases. It returns a tuple of four arrays: the first two are obtained from `sph_in(n,z)` and the last two are obtained from `sph_kn(n,z)`.

The functions $zj_n(z)$ and $zy_n(z)$ are called Ricatti-Bessel functions, computed respectively by `ricatti_jn(n,z)` and `ricatti_yn(n,z)`.

6 Struve Functions

6.1 $\mathbf{H}_\nu$(z)

The **Struve function** $\mathbf{H}_\nu$(z) appears in the solution of the inhomogeneous Bessel equation which has the form

$$z^2y'' + zy' + (z^2 - \nu^2)y = \frac{4}{\sqrt{\pi}} \frac{(z/2)^{\nu+1}}{\Gamma(\nu + 1/2)} \quad (36)$$

The general solution to this equation is:

$$y = aJ_\nu(z) + bY_\nu(z) + \mathbf{H}_\nu(z) \quad (37)$$

where a and b are constants. The power series expansion is

$$\mathbf{H}_\nu(z) = \left(\frac{z}{2}\right)^{\nu+1} \sum_{k=0}^{\infty} \frac{(-1)^k (z/2)^{2k}}{\Gamma(k+\frac{3}{2})\Gamma(k+\nu+\frac{3}{2})} \quad (38)$$

A common integral representation is

$$\mathbf{H}_\nu(z) = \frac{2(z/2)^\nu}{\sqrt{\pi}} \Gamma(\nu + \frac{1}{2}) \int_0^{\frac{\pi}{2}} \sin(z\cos\theta)\sin^{2\nu}\theta\, d\theta \quad (39)$$

The Python function `struve(v,x)` returns $\mathbf{H}_\nu$(z) of order `v` at `x`, `x` must be positive unless `v` is an integer.

6.2 $\mathbf{L}_\nu$(z)

The **Modified Struve function** $\mathbf{H_n}$(z) is

$$\begin{aligned} \mathbf{L}_\nu(z) &= -je^{-\frac{j\nu\pi}{2}} \mathbf{H}_\nu(jz) \\ &= \left(\frac{z}{2}\right)^{\nu+1} \sum_{k=0}^{\infty} \frac{(z/2)^{2k}}{\Gamma(k+\frac{3}{2})\Gamma(k+\nu+\frac{3}{2})} \end{aligned} \quad (40)$$

An integral representation for $\Re\{\nu\} > -\frac{1}{2}$ is

$$\mathbf{L}_\nu(z) = \frac{2(z/2)^\nu}{\sqrt{\pi}} \Gamma(\nu + \frac{1}{2}) \int_0^{\frac{\pi}{2}} \sinh(z\cos\theta)\sin^{2\nu}\theta\, d\theta \quad (41)$$

7 Gamma and Related Functions

7.1 $\Gamma(z)$

$\Gamma(z)$ is a generalization of the factorial function. It is defined by Euler's integral

$$\begin{aligned} \Gamma(z) &= \int_0^\infty t^{z-1} e^{-t}\, dt \quad (\Re\{z\} > 0) \\ &= k^z \int_0^\infty t^{z-1} e^{-kt}\, dt \; (\Re\{z\}, \Re\{k\} > 0) \end{aligned} \quad (42)$$

If $z = n, n > 0$ is an integer then $\Gamma(n) = (n-1)!$. Note that $\Gamma(0) = \infty$. If $z \notin \mathbb{I}^-$,

$$\Gamma(z) = \lim_{n\to\infty} \frac{n!n^z}{z(z+1)\cdots(z+n)} \quad (43)$$

`gamma(z)` evaluates $\Gamma(z)$. The `gammaln(z)` computes the natural logarithm of $\Gamma(z)$ with a single branch cut along the negative real axis. Provided ($\Re\{z\} > 0$)

$$\ln\Gamma(z) = \int_0^\infty \left[(z-1)e^{-t} - \frac{e^{-t}-e^{-zt}}{1-e^{-t}}\right]\frac{dt}{t} \quad (44)$$

A point to be noted is that, according to Scipy's implementation:

$$\texttt{gammaln(z)} = \texttt{ln(abs(gamma(z)))} \quad (45)$$

The reciprocal of $\Gamma(z)$ is computed by `rgamma(z)` according to *Euler's product*:

$$\frac{1}{\Gamma(z)} = ze^{\gamma z}\prod_{n=1}^{\infty}\left[\left(1+\frac{z}{n}\right)e^{-z/n}\right] \quad (46)$$

The Euler's constant γ is defined by

$$\begin{aligned}\gamma &= \lim_{m\to\infty}\left[1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{m}-\ln m\right] \\ &= 0.57721566499\ldots\end{aligned} \quad (47)$$

Also,

$$\gamma = \int_0^\infty \left(\frac{1}{1+t} - e^{-t}\right)\frac{dt}{t} \quad (48)$$

As an aside, we define **Pochhammer's rising factorial**,

$$\begin{aligned}(a)_n &= a(a+1)\cdots(a+n-1) \\ &= \frac{\Gamma(a+n)}{\Gamma(a)}\end{aligned} \quad (49)$$

7.2 $P(a,x), Q(a,x)$

The **Incomplete Gamma** function is defined by

$$P(a,x) = \frac{1}{\Gamma(a)}\int_0^x e^{-t}t^{a-1}\,dt \quad (50)$$

and the **Complementary Incomplete Gamma** function is evaluated as

$$\begin{aligned}Q(a,x) &= 1-P(a,x) \\ &= \frac{1}{\Gamma(a)}\int_x^\infty e^{-t}t^{a-1}\,dt\end{aligned} \quad (51)$$

Scipy evaluates $P(a,x)$ using `gammainc(a,x)` and $Q(a,x)$ as `gammaincc(a,x)` only when $a, x > 0$, even though the arguments in Equation 50 and Equation 51 can be complex-valued. These forms are also called the *regularized* gamma integrals.

The inverse gamma function `gammaincinv(a,s)` is the solution for x in the equation $s = P(a,x)$. Similarly `gammainccinv(a,s)` solves for x in $s = Q(a,x)$.

7.3 $B(a,b), I_x(a,b)$

The **Euler Beta** function

$$\begin{aligned}B(a,b) &= \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)} \\ &= \int_0^1 t^{a-1}(1-t)^{b-1}\,dt \\ &= 2\int_0^{\frac{\pi}{2}}(\sin t)^{2a-1}(\cos t)^{2b-1}\,dt\end{aligned} \quad (52)$$

is returned by `beta(a,b)`. As with `gammaln(a)`,

$$\texttt{betaln(a,b)} = \texttt{ln(abs(beta(a,b)))} \quad (53)$$

The **regularized incomplete beta** function, defined as,

$$I_x(a,b) = \frac{1}{B(a,b)}\int_0^z t^{a-1}(1-t)^{b-1}\,dt \quad (54)$$

is computed by `betainc(a,b,x)`.

There are three inverse beta functions available in Scipy. If `s=betainc(a,b,x)`, then

$$\texttt{x} = \texttt{betaincinv(a,b,s)} \quad (55)$$

$$\texttt{a} = \texttt{betaincinva(s,b,x)} \quad (56)$$

$$\texttt{b} = \texttt{betaincinvb(a,s,x)} \quad (57)$$

7.4 $\psi(z), \psi^{(n)}(z)$

The *digamma* function is the logarithmic derivative of the Gamma function:

$$\psi(z) = \frac{d}{dz}\ln\Gamma(z) = \frac{\Gamma'(z)}{\Gamma(z)} \quad (58)$$

For $z = n \in \mathbb{I}^+$ and $n \geq 2$

$$\psi(1) = -\gamma; \psi(n) = -\gamma + \sum_{k=1}^{n-1} k^{-1} \tag{59}$$

where γ is the Equation 47. For complex z, $\Re\{z\} \geq 0$

$$\begin{aligned} \psi(1+z) &= -\gamma + \sum_{n=1}^{\infty} \frac{z}{n(n+z)} \\ \psi(z) &= \int_0^\infty \left[\frac{e^{-t}}{t} - \frac{e^{-zt}}{1-e^{-t}} \right] dt \\ \psi(z) &= -\gamma + \int_0^\infty \frac{e^{-t} - e^{-zt}}{1-e^{-t}} dt \end{aligned} \tag{60}$$

It is computed by `psi(z)` which has an alias `digamma(z)`.

The n^{th} derivative of $\psi(z)$ gives the **Polygamma** functions. For complex z, $\Re\{z\} \geq 0$

$$\begin{aligned} \psi^{(n)}(z) &= \frac{d^n}{dz^n} \psi(z) \\ &= \frac{d^{n+1}}{dz^{n+1}} \ln \Gamma(z) \\ &= (-1)^{n+1}\, n! \sum_{k=0}^{\infty} \frac{1}{(z+k)^{n+1}} \end{aligned} \tag{61}$$

which you can evaluate by `polygamma(n,z)`

8 Error Function and Fresnel Integtrals

8.1 erf z, erfc z

The **error function** is defined by the integral

$$\begin{aligned} \operatorname{erf}(z) &= \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2}\, dt \\ &= \frac{2}{\sqrt{\pi}} \sum_{n=0}^{\infty} \frac{(-1)^n z^{2n+1}}{n!\,(2n+1)} \end{aligned} \tag{62}$$

The **complementary error function** is defined by the integral

$$\begin{aligned} \operatorname{erfc}(z) &= 1 - \operatorname{erf}(z) \\ &= \frac{2}{\sqrt{\pi}} \int_z^\infty e^{-t^2}\, dt \end{aligned} \tag{63}$$

The Scipy's functions `erf(z)` and `erfc(z)` respectively compute $\operatorname{erf}(z)$ and $\operatorname{erfc}(z)$.

8.2 $C(z)$, $S(z)$

The **Fresnel cosine integral** defined by

$$\begin{aligned} C(z) &= \int_0^z \cos\left(\frac{\pi}{2} t^2\right) dt \\ &= \sum_{n=0}^{\infty} \frac{(-1)^n (\pi/2)^{2n}}{(2n)!\,(4n+1)} z^{4n+1} \end{aligned} \tag{64}$$

and the **Fresnel sine integral** by

$$\begin{aligned} S(z) &= \int_0^z \sin\left(\frac{\pi}{2} t^2\right) dt \\ &= \sum_{n=0}^{\infty} \frac{(-1)^n (\pi/2)^{2n+1}}{(2n+1)!\,(4n+3)} z^{4n+3} \end{aligned} \tag{65}$$

are computed by `fresnel(z)` which returns a two-element tuple containing $S(z)$ and $C(z)$ respectively.

8.3 $F_+(z), K_+(z), F_-(z), K_-(z)$

The **modified Fresnel integrals**

$$\begin{aligned} F_+(z) &= \int_x^\infty e^{jt^2}\, dt \\ K_+(z) &= \frac{1}{\sqrt{\pi}} e^{-j(x^2+\pi/4)} F_+(z) \end{aligned} \tag{66}$$

are returned as a tuple by `modfresnelp(x)` and

$$\begin{aligned} F_-(z) &= \int_x^\infty e^{-jt^2}\, dt \\ K_-(z) &= \frac{1}{\sqrt{\pi}} e^{-j(x^2+\pi/4)} F_-(z) \end{aligned} \tag{67}$$

are returned as a tuple by `modfresnelm(x)`.

9 Hypergeometric Functions

9.1 ${}_1F_1$, ${}_0F_1, U$

The independent solutions of Kummer's differential equation,

$$zy'' + (b-z)y' - ay = 0 \tag{68}$$

are denoted by ${}_1F_1(a;b;z)$ and $U(a,b,z)$. These are called the **confluent hypergeometric** functions of first and second kind respectively. The power series is

$$ {}_1F_1(a;b;z) = \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{z^k}{k!} \tag{69} $$

where $(a)_k$ is defined in Equation 49; and the integral representation is

$$ {}_1F_1(a;b;z) = \frac{\Gamma(b)}{\Gamma(b-a)\Gamma(a)} \int_0^1 e^{zt} t^{a-1} (1-t)^{b-a-1}\, dt \tag{70} $$

The integral representation of $U(a,b,z)$ is

$$ U(a,b,z) = \frac{1}{\Gamma(a)} \int_0^{\infty} e^{-zt} t^{a-1} (1+t)^{b-a-1}\, dt \tag{71} $$

A limiting form of the confluent hypergeometric function which often appears is ${}_0F_1(;a;z)$ defined as:

$$ \begin{aligned} {}_0F_1(;a;z) &= \lim_{q\to\infty} {}_1F_1(q;a;z/q) \\ &= \sum_{k=0}^{\infty} \frac{1}{(a)_k} \frac{z^k}{k!} \end{aligned} \tag{72} $$

and satisfies the differential equation,

$$ zy'' + ay' - y = 0 $$

Scipy's `hyp1f1(a,b,x)`, `hyperu(a,b,x)` and `hyp0f1(a,z)` respectively evaluate ${}_1F_1(a;b;z)$, $U(a,b,z)$ and ${}_0F_1(;a;z)$.

9.2 ${}_2F_1$, ${}_2F_0$, ${}_1F_2$, ${}_3F_0$

The Gauss' **hypergeometric** function is a solution of Gauss' differential equation:

$$ z(1-z)y'' + [c-(a+b+1)z]y' - aby = 0 \tag{73} $$

whose power series is

$$ {}_2F_1(a,b;c;z) = \sum_{k=0}^{\infty} \frac{(a)_k (b)_k}{(c)_k} \frac{z^k}{k!} \tag{74} $$

p	q	Function	Scipy
2	0	${}_2F_0(a,b;;z)$	`hyp2f0(a,b,z,T)`
1	2	${}_1F_2(a;b,c;z)$	`hyp1f2(a,b,c,z)`
3	0	${}_3F_0(a,b,c;;z)$	`hyp3f0(a,b,c,z)`

Table 1: Hypergeometric Functions

and integral representation is

$$ {}_2F_1(a,b;c;z) = \frac{\Gamma(c)}{\Gamma(b)\Gamma(c-b)} \times \int_0^1 (1-b)^{c-b-1} (1-tz)^{-a}\, dt \tag{75} $$

Let $\mathbf{a} = (a_1, a_2, \cdots, a_p)$ and $\mathbf{b} = (b_1, b_2, \cdots, a_q)$ be two arrays of length p and q respectively, with $\Re\{a_i\} > 0$, $\Re\{b_j\} > 0$. The Barnes' **generalized hypergeometric** series is given by

$$ {}_pF_q(\mathbf{a};\mathbf{b};z) = \sum_{k=0}^{\infty} \frac{\prod_{i=1}^{p} (a_i)_k}{\prod_{j=1}^{q} (b_j)_k} \frac{z^k}{k!} \tag{76} $$

Table 1 provides the various functions avaialabe in Scipy for different values of p and q. All the functions in Table 1 return a tuple `(y,err)` where `y` is the actual value of the function and `err` is an error estimate. Additionally `hyp2f0(a,b,z,T)` takes a fourth argument, `T`, that determines a convergence factor and can be either 1 or 2 only.

10 Orthogonal Polynomials

A system of polynomials $f_n(x)$, degree $[f_n(x)] = n$, is called **orthogonal** on the interval $a \le x \le b$, with respect to the weight function $w(x)$, if

$$ \int_a^b w(x) f_n(x) f_m(x)\, dx = 0 \qquad (n \ne m; n, m = 0, 1, 2, \ldots) \tag{77} $$

Thw weight function $w(x)[w(x) \geq 0]$ determines the system $f_n(x)$ up to a constant factor in each polynomial. For each of these polynomials, we list out

- Orthogonality relation
- Power series expansion
- Differential equation
- Recurrence equation
- Rodrigues' Formula

All these functions return a polynomial class (`poly1d` object) which can then be evaluated: `vals = chebyt(n)(x)`. This class also has an attribute `'weights'` which return the roots, weights, and total weights for the appropriate form of Gaussian quadrature. These are returned in an $n \times 3$ array with roots in the first column, weights in the second column, and total weights in the final column.

10.1 $P_n(x), P_n^*(x)$

Legendre polynomials, $P_n(x)$, arise in studies of systems with three-dimensional spherical geometry. The satisfy

$$\int_{-1}^{1} P_m(x) P_n(x)\, dx = 0 \tag{78}$$

$$P_n(x) = \frac{1}{2^n} \sum_{m=0}^{[n/2]} \binom{n}{m} \binom{2n-2m}{n} x^{n-2m} \tag{79}$$

$$(1-x^2)y'' - 2xy' + n(n+1)y = 0 \tag{80}$$

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x) \tag{81}$$

$$P_n(x) = \frac{1}{(-1)^n 2^n\, n!} \frac{d^n}{dx^n}(1-x^2)^n \tag{82}$$

which can be evaluated by `legendre(n)(x)`. A shifted version, $P_n^*(x)$, orthogonal over $[0, 1]$ with weighting function 1 can be evaluated using `sh_legendre(n)(x)`

10.2 $T_n(x), C_n(x), T_n^*(x)$

Series of Chebyshev polynomials are often used in making numerical approximations to functions. **Chebyshev polynomial** of *first* kind $T_n(x)$ are specified by

$$\int_{-1}^{1} T_m(x) T_n(x) \frac{1}{\sqrt{1-x^2}}\, dx = 0 \tag{83}$$

$$T_n(x) = \frac{n}{2} \sum_{m=0}^{[n/2]} \frac{n}{2} (-1)^m \frac{(n-m-1)!}{m!\,(n-2m)!} (2x)^{n-2m} \tag{84}$$

$$(1-x^2)y'' - xy' + n^2 y = 0 \tag{85}$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \tag{86}$$

$$T_n(x) = \frac{\sqrt{\pi}(1-x^2)}{(-1)^n 2^n \Gamma(n+1/2)} \frac{d^n}{dx^n} \frac{(1-x^2)^n}{\sqrt{(1-x^2)}} \tag{87}$$

are computed with `chebyt(n)(x)` . Another version of Chebyshev's polynomial of first kind, `chebyc(n)(x)` is given by

$$C_n(x) = 2T_n(x/2) \tag{88}$$

A shifted version is also available, via `sh_chebyt(n)(x)`

$$T_n^*(x) = T_n(2x-1) \tag{89}$$

10.3 $U_n(x), S_n(x), U_n^*(x)$

Chebyshev polynomials of *second* kind, $U_n(x)$, satisfy

$$\int_{-1}^{1} U_m(x) U_n(x) \sqrt{1-x^2}\, dx = 0 \tag{90}$$

$$U_n(x) = \sum_{m=0}^{[n/2]} (-1)^m \frac{(n-m)!}{m!\,(n-2m)!} (2x)^{n-2m} \tag{91}$$

$$(1-x^2)y'' - 3xy' + n(n+2)y = 0 \tag{92}$$

$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x) \tag{93}$$

$$U_n(x) = \frac{(n+1)\sqrt{\pi}}{(-1)^n 2^{n+1} \Gamma(n+3/2) \sqrt{(1-x^2)}} \times \frac{d^n}{dx^n}(1-x^2)^{n+1/2} \tag{94}$$

Another version of Chebyshev's polynomial of second kind, `chebys(n)(x)` is given by

$$S_n(x) = U_n(x/2) \quad (95)$$

A shifted version is also available, via `sh_chebyu(n)(x)`

$$U_n^*(x) = U_n(2x-1) \quad (96)$$

10.4 $P_n^{(a,b)}(x), G_n(p,q,x)$

Jacobi polynomials, $P_n^{(a,b)}(x)$, $(a > -1, b > -1)$ occur in studies of the rotation group, particularly in quantum mechanics.

$$\int_{-1}^{1} P_m^{(a,b)}(x) P_n^{(a,b)}(x)(1-x)^a(1-x)^b\, dx = 0 \quad (97)$$

$$P_n^{(a,b)}(x) = \frac{1}{2^n}\sum_{m=0}^{n}\binom{n+a}{m}\times \binom{n+b}{n-m}(x-1)^{n-m}(x+1)^m \quad (98)$$

$$(1-x^2)y'' + [b-a-(a+b+2)x]y' + n(n+a+b+1)y = 0 \quad (99)$$

$$2(n+1)(n+a+b+1)(2n+a+b)P_{n+1}^{(a,b)}(x) = [(2n+a+b+1)(a^2-b^2)+(2n+a+b)^3x]P_n^{(a,b)}(x) - 2(n+a)(n+b)(2n+a+b+2)P_{n-1}^{(a,b)}(x) \quad (100)$$

$$P_n^{(a,b)}(x) = \frac{(-1)^{-n}2^{-n}}{n!(1-x)^a(1-x)^b}\times \frac{d^n}{dx^n}(1-x)^a(1-x)^b(1-x^2)^n \quad (101)$$

$P_n^{(a,b)}(x)$ is evaluated by `jacobi(n,a,b)(x)`. A shifted Jacobi version, defined by,

$$G_n(p,q,x) = \frac{n!\,\Gamma(n+p)}{\Gamma(2n+p)}P_n^{(p-q,q-1)}(2x-1) \quad (102)$$

where $p-q > -1; q > 0$, can be evaluated using `sh_jacobi(n,p,q)`.

10.5 $L_n(x), L_n^{(a)}(x)$

Generalized Laguerre polynomials, $L_n^{(a)}(x), a > -1$ are related to hydrogen atom wave functions in quantum mechanics.

$$\int_0^{\infty} L_m^a(x)L_n^a(x)x^a e^{-x}\, dx = 0 \quad (103)$$

$$L_n^a(x) = \sum_{m=0}^{n}\frac{(-1)^m}{m!}\binom{n+a}{n-m}x^m \quad (104)$$

$$xy'' + (a+1-x)y' + ny = 0 \quad (105)$$

$$(n+1)L_{n+1}^a(x) = (2n+a+1-x)L_n^{(a)}(x) - (n+a)L_{n-1}^{(a)}(x) \quad (106)$$

$$L_n^{(a)}(x) = \frac{e^x}{n!x^a}\frac{d^n}{dx^n}e^{-x}x^a x^n \quad (107)$$

$L_n^{(a)}(x)$ is given by `genlaguerre(n,a)(x)`. A special case occurs when $a = 0$; the **Laguerre polynomials** $L_n(x)$

$$L_n(x) = L_n^{(0)}(x) \quad (108)$$

are evaluated by `laguerre(n)(x)`.

10.6 $C_n^{(a)}(x)$

Gegenbauer polynomials $C_n^{(a)}(x), a > -\frac{1}{2}$ can be viewed as generalizations of the Legendre polynomials to systems with $(m+2)-$dimensional spherical symmetry. They are sometimes known as **ultraspherical polynomials**.

$$\int_{-1}^{1} C_m^{(a)}(x)C_n^{(a)}(x)\frac{(1-x^2)^a}{\sqrt{1-x^2}}\, dx = 0 \quad (109)$$

$$C_n^{(a)}(x) = \frac{1}{\Gamma(a)}\times \sum_{m=0}^{[n/2]}\frac{(-1)^m\Gamma(a+n-m)}{m!(n-2m)!}(2x)^{n-2m} \quad (110)$$

$$(1-x^2)y'' - (2a+1)xy' + n(n+2a)y = 0 \tag{111}$$

$$(n+1)C_{n+1}^{(a)}(x) = xC_n^{(a)}(x) - C_{n-1}^{(a)}(x) \tag{112}$$

$$C_n^{(a)}(x) = \frac{(-1)^n\Gamma(a+\frac{1}{2})\Gamma(n+2a)(1-x^2)^{\frac{1}{2}}}{2^n n!\Gamma(2a)\Gamma(a+n+\frac{1}{2})(1-x^2)^a} \times \frac{d^n}{dx^n}(1-x^2)^{n+a-\frac{1}{2}} \tag{113}$$

$C_n^{(a)}(x)$ is evaluated by `gegenbauer(n,a)(x)`.

10.7 $H_n(x), He_n(x)$

Hermite polynomials $H_n(x)$ arise as the quantum-mechanical wave functions for a harmonic oscillator.

$$\int_{-\infty}^{\infty} H_m(x)H_n(x)e^{-x^2}\,dx = 0 \tag{114}$$

$$H_n(x) = n!\sum_{m=0}^{[n/2]} \frac{(-1)^m(2x)^{n-2m}}{m!(n-2m)!} \tag{115}$$

$$y'' - 2xy' + 2ny = 0 \tag{116}$$

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x) \tag{117}$$

$$H_n(x) = (-1)^n e^{x^2}\frac{d^n}{dx^n}e^{-x^2} \tag{118}$$

$H_n(x)$ is evaluated by `hermite(n)(x)`. The **normalized Hermite** polynomials, $He_n(x)$, given by

$$He_n(x) = 2^{-n/2}H_n(x/\sqrt{2}) \tag{119}$$

are available with `hermitenorm(n)(x)`.

11 Exponential Integrals

11.1 $Ei(x), E_n(x), E_1(x), \zeta(s), Li_2(z)$

Exponential integrals of various types,

$$Ei(x) = \int_{-\infty}^{x} \frac{e^t}{t}\,dt \quad x > 0$$

$$E_n(z) = \int_1^{\infty} \frac{e^{-zt}}{t^n}\,dt \quad \Re\{z > 0\} \tag{120}$$

$$E_1(z) = \int_z^{\infty} \frac{e^{-t}}{t}\,dt$$

are respectively evaluated by `expi(x)`, `expn(n,x)` and `exp1(z)`.

12 Zeta and Related Functions

12.1 $\zeta(s,a), \zeta(s), \mathrm{Li}_2(z)$

The **generalized Riemann zeta** function or **Hurwitz zeta** function:

$$\zeta(s,a) = \sum_{k=0}^{\infty} \frac{1}{(k+a)^s} \tag{121}$$

where any term with $k+a=0$ is excluded. This is given by `zeta(s,a)`. The standard **Riemann zeta** function is a special case where $a=0$.

$$\zeta(s) = \sum_{k=1}^{\infty} \frac{1}{k^s} \tag{122}$$

$$= 1 + \sum_{k=2}^{\infty} \frac{1}{k^s} \tag{123}$$

Scipy provides `zetac(s)` which evaluates $\{\zeta(s) - 1.\}$ So actually, `1+zetac(s)` gives $\zeta(s)$.

The **Spence's integral**, also called the **dilogarithm** function,

$$\mathrm{Li}_2(z) = \int_1^{\infty} \frac{\ln t}{t-1}\,dt \tag{124}$$

$$= \sum_{k=1}^{\infty} \frac{(1-z)^k}{k^2} \tag{125}$$

can be evaluated using `spence(z)`.

13 Sine and Cosine Integrals

13.1 $\mathrm{Si}(z), \mathrm{Ci}(z), \mathrm{Shi}(z), \mathrm{Chi}(z)$

The **sine** integral:

$$\mathrm{Si}(z) = \int_0^z \frac{\sin t}{t}\,dt \tag{126}$$

The **cosine** integral:

$$\mathrm{Ci}(z) = \gamma + \ln z + \int_0^z \frac{\cos t - 1}{t}\,dt \tag{127}$$

The command `sici(z)` returns a two-element tuple, having $\mathrm{Si}(z)$ and $\mathrm{Ci}(z)$. The **hyperbolic sine** integral:

$$\mathrm{Shi}(z) = \int_0^z \frac{\sinh t}{t} \tag{128}$$

The **hyperbolic cosine** integral:

$$\mathrm{Chi}(z) = \gamma + \ln z + \int_0^t \frac{\cosh t - 1}{t}\, dt \tag{129}$$

The command `shichi(z)` returns a two-element tuple, having $\mathrm{Shi}(z)$ and $\mathrm{Chi}(z)$.

14 License

This work is licensed under the Creative Commons Attribution-Noncommercial-ShareAlike-3.0 License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-sa/3.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

Attribution: You can view and modify the content of this document provided you retain my name and my email id mailto:bdsatish@techgeek.co.in in all of your derivative works.

Noncommercial: You cannot use this article for *any* commercial purpose.

Share Alike: You can adapt this document provided the derivative work is distributed under the same license as this.

All the functions in `scipy.special` are not covered here. I have left out some twenty or so functions. If you want any additions or have any suggestions, do let me know! [[1]]

[1] Retreived from http://bdsatish.googlepages.com