
acsefunctions

Release 0.1

Your Name

Apr 19, 2025

CONTENTS

Python Module Index	3
Index	5

Special functions: factorial, gamma, and Bessel.

`acsefunctions.special.bessel(alpha, x, n_terms=20)`

Compute the Bessel function $J_{\alpha}(x)$ using its series expansion.

Parameters

- **alpha** (*float*) – Order of the Bessel function.
- **x** (*float or numpy.ndarray*) – Input value(s).
- **n_terms** (*int, optional*) – Number of terms in the series (default is 20).

Returns

Computed $J_{\alpha}(x)$.

Return type

float or numpy.ndarray

Examples

```
>>> bessel(0, 0)
1.0
>>> bessel(0, 1) # Approximate value
0.7651976865579666
>>> bessel(0, np.array([0, 1]))
array([1.          , 0.76519769])
```

`acsefunctions.special.factorial(n)`

Compute the factorial $n!$ for non-negative integers.

Parameters

n (*int or numpy.ndarray*) – Non-negative integer input(s).

Returns

Computed $n!$.

Return type

int or numpy.ndarray

Raises

ValueError – If n is negative.

Examples

```
>>> factorial(0)
1
>>> factorial(5)
120
>>> factorial(np.array([0, 1, 2]))
array([1, 1, 2])
```

`acsefunctions.special.gamma(z, T=100, M=1000)`

Compute the gamma function $\gamma(z)$ for $z > 0$ using numerical integration.

Uses trapezoidal rule on $\gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$.

Parameters

- **z** (*float or numpy.ndarray*) – Input value(s), must be positive.
- **T** (*float, optional*) – Upper integration limit (default is 100).

- $M(int, optional)$ – Number of integration points (default is 1000).

Returns

Computed $\gamma(z)$.

Return type

float or numpy.ndarray

Raises

ValueError – If $z \leq 0$.

Examples

```
>>> gamma(1)
1.0
>>> gamma(0.5) # Approximately sqrt(pi)
1.7724538209055159
>>> gamma(np.array([1, 2]))
array([1., 1.])
```

PYTHON MODULE INDEX

a

`acsefunctions.special`, [1](#)

`acsefunctions.transcendental`, [1](#)

A

`acsefunctions.special`
 module, 1
`acsefunctions.transcendental`
 module, 1

B

`bessel()` (*in module `acsefunctions.special`*), 1

F

`factorial()` (*in module `acsefunctions.special`*), 1

G

`gamma()` (*in module `acsefunctions.special`*), 1

M

module
 `acsefunctions.special`, 1
 `acsefunctions.transcendental`, 1