mathMB

Release 1.4

unknown

Sep 24, 2020

CONTENTS

mathMB provides mathematical functions designed to be used by nexoclom, although they can be used as stand-alone routines.

CHAPTER

ONE

INTERPU

`interpu()`: 1D linear interpolation using astropy quantities.

This is a wrapper for numpy.interp for use when using astropy quantities. If x and xp have different units, xp is converted to the units of x before interpolation. An exception is raised if the units are not compatible (i.e., the units of xp cannot be converted to the units of x).

Author Matthew Burger

`mathMB.interpu.`**`interpu`**(*x*, *xp*, *fp*, ***kwargs*)

Return one dimensional interpolated astropy quantities.

Parameters

x The x-coordinates at which to evaluate the interpolated values

xp The x-coordinates of the data points.

fp The y-coordinates of the data points

Notes

x and xp must have compatible units. See numpy.interp for details on interpolation.

CHAPTER

TWO

MINMAXMEAN

`minmaxmean()` : Print np.min(), np.max(), np.mean() at once.

`mathMB.minmaxmean.`**`minmaxmean`**`(`*`x`*`)`

Print np.min(), np.max(), np.mean() at once.

CHAPTER

THREE

RANDOMDEVIATES

Compute random deviates from arbitrary 1D and 2D distributions.

mathMB.randomdeviates.**random_deviates_1d**(*x*, *f_x*, *num*)

Compute random deviates from arbitrary 1D distribution.

f_x does not need to integrate to 1. The function nomralizes the distribution. Uses Transformation method (Numerical Recepies, 7.3.2)

Parameters

x The x values of the distribution

f_x The relative probability of the value being in x and x+dx

num The number of random deviates to compute

Returns

numpy array of length num chosen from the distribution f_x.

mathMB.randomdeviates.**random_deviates_2d**(*fdist*, *x0*, *y0*, *num*)

Compute random deviates from arbitrary 2D distribution.

Uses acceptance/rejection method. **Parameters**

fdist 2d array of relative probability

x0 xaxis

y0 yaxis

num number of points to choose

Outputs

x, y vectors of length num

CHAPTER

FOUR

ROTATION_MATRIX

`rotation_matrix()`: Compute the rotation matrix about an axis.

`mathMB.rotation_matrix.`**`rotation_matrix`**(*theta*, *axis*)

Compute the rotation matrix for a rotation of theta about axis.

Authors Matthew Burger

License LICENSE

PYTHON MODULE INDEX

m

INDEX