NCL functions & procedures science reference card

NCL version 6.2.1   Karin Meier-Fleischer, DKRZ
December 04, 2014   Mary Haley, NCAR

Read the corresponding web page to get more information about a function or procedure:

http://ncl.ucar.edu/Document/Functions/index.shtml

Statistics

dtrend
Estimates and removes the least squares linear trend of the rightmost dimension from all grid points.

dtrend_leftdim
Estimates and removes the least squares linear trend of the leftmost dimension from all grid points and retains meta data.

dtrend_msg
Estimates and removes the least squares linear trend of the rightmost dimension from all grid points (missing values allowed).

dtrend_msg_n
Estimates and removes the least squares linear trend of the dim-th dimension from all grid points (missing values allowed).

dtrend_n
Estimates and removes the least squares linear trend of the given dimension from all grid points.

dtrend_quadratic
Estimates and removes the least squares quadratic trend of the rightmost dimension from all grid points.

dtrend_quadratic_msg_n
Estimates and removes the least squares quadratic trend of the dim-th dimension from all grid points (missing values allowed).

General applied math

arrayshift
Rearrange an array in a manner similar to Matlab's fftshift.

betainc
Evaluates the incomplete beta function.

calculate_monthly_values
Calculate monthly values [avg, sum, min, max] from high frequency temporal values.

cancor
Performs canonical correlation analysis between two sets of variables.

cdft_p
Calculates the one-sided probability given a t-value and the degrees of freedom.

cdft_t
Calculates the t-value given the one-sided probability and the degrees of freedom.

center_finite_diff
Performs a centered finite difference operation on the rightmost dimension.

center_finite_diff_n
Performs a centered finite difference operation on the given dimension.

cfftb
Performs a backward complex discrete fourier transform [Fourier Synthesis].

cfftf
Performs a forward complex discrete fourier transform of a real periodic sequence.

cfftf_frq_reorder
Reorders the data returned by cfftf.

chiinv
Evaluates the inverse chi-squared distribution function.

covcorm
Calculates a covariance or correlation matrix.

covcorm_xy
Calculates a covariance or correlation matrix given two separate arrays.

cumsum
Calculates the cumulative sum.

decimalPlaces
Truncates or rounds to the number of decimal places specified

equiv_sample_size
Estimates the number of independent values of a series of correlated observations.

erf
Evaluates the real error function.

erfc
Evaluates the real complementary error function.

esacr
Computes sample auto-correlations

esacv
Computes sample auto-covariances

esccr
Computes sample cross-correlations.

esccv
Computes sample cross-covariances.

escorc
Computes sample linear cross-correlations at lag 0 only.

escorc_n
Computes sample cross-correlations at lag 0 only, across the given dimensions.

escovc
Computes sample cross-covariances at lag 0 only.

ezfftb
Perform a Fourier synthesis from real and imaginary coefficients.

ezfftf
Perform a Fourier analysis on a real periodic sequence.

fft2db
Performs a two-dimensional discrete backward Fourier transform (Fourier synthesis).

fft2df
Performs a two-dimensional forward real discrete Fourier transform (i.e., Fourier analysis) of a real periodic array.

filwgts_lanczos
Calculates one-dimensional filter weights.

filwgts_normal
Calculates one-dimensional filter weights based upon the normal (gaussian) distribution.

fourier_info
Performs fourier analysis on one or more periodic series.

ftest
Applies F-test for variances and returns an estimate of the statistical significance.

gamma
Evaluates the complete gamma function.

gammainc
Evaluates the incomplete gamma function.

genNormalDist
Generates a normal distribution.

inverse_matrix
Computes the inverse of a general matrix using LU factorization.

kf_filter
Extract equatorial waves by filtering in the Wheeler-Kiladis wavenumber-frequency domain.

kolsm2_n
Use the Kolmogorov-Smirnov two-sample test to determine if two samples are from the same distribution.

kron_product
Computes the Kronecker product for two-dimensional matrices.

linrood_latwgt
Computes the latitudes and weights used by the Lin-Rood Model.

linrood_wgt
Computes the weights used by the Lin-Rood Model.

local_max
Determines the relative maxima for a 2-dimensional array.

local_min
Determines the relative minima for a 2-dimensional array.

lspoly
Calculates a set of coefficients for a weighted least squares polynomial fit to the given data.

lspoly_n
Calculates a set of coefficients for a weighted least squares polynomial fit to the given data on the given dimension.

NewCosWeight
Performs cosine of the latitude weighting on the given array.

pattern_cor
Compute centered or uncentered pattern correlation.

pdfx
Generates a univariate probability density distribution (PDF).

pdfxy
Generates a joint probability density distribution. (Please use pdfxy_conform.)

pdfxy_bin

Performs looping necessary to calculate the bivariate (joint) probability distribution (see pdfxy).

pdfxy_conform

An interface to pdfxy that allows the input arrays to be different sizes.

quadroots

Determine roots of a quadratic equation [a*x^2 + b*x + c].

reg_multlin

Performs basic multiple linear regression analysis.

reg_multlin_stats

Performs multiple linear regression analysis including confidence estimates and creates an ANOVA table.

regcoef

Calculates the linear regression coefficient between two variables.

regCoef

Calculates the linear regression coefficient between two variables.

regCoef_n

Calculates the linear regression coefficient between two variables on the given dimensions.

regline

Calculates the linear regression coefficient between two series.

regline_stats

Performs simple linear regression including confidence estimates and creates an ANOVA table.

rmInsufData

Sets all instances (i.e. time) of a grid point to missing if a user-prescribed percentage of points is missing.

run_cor

Calculates a running correlation.

runave

Calculates an unweighted running average on the rightmost dimension.

runave_n

Calculates an unweighted running average on the given dimension.

runave_n_Wrap

Calculates an unweighted running average on the given dimension and retains metadata.

runave_Wrap

Calculates an unweighted running average on the rightmost dimension and retains metadata.

sign_f90

Mimic the behavior of fortran-90's sign transfer function.

sign_matlab

Mimic the behavior of Matlab's sign function.

simpeq

Integrates a sequence of equally spaced points using Simpson's Rule.

simpne

Integrates a sequence of unequally spaced points using Simpson's three-point formula.

sindex_yrmo

Calculates the Southern Oscillation Index given two series of year-month values.

smth9

Performs nine point local smoothing on one or more 2D grids.

smth9_Wrap

Performs nine point local smoothing on one or more 2D grids and retains metadata.

snindex_yrmo

Calculates the Southern Oscillation Index and the noise index given two series of year-month values.

solve_linsys

Computes the solution to a real system of linear equations.

sparse_matrix_mult

Multiples a sparse matrix with a dense matrix. Function under construction. Calling arguments may change.

spcorr

Computes Spearman rank order correlation coefficient.

spcorr_n

Computes Spearman rank order correlation coefficient across the given dimension.

specx_anal

Calculates spectra of a series.

specx_ci

Calculates the theoretical Markov spectrum and the lower and upper confidence curves.

specxy_anal

Calculates cross spectra quantities of a series.

SqrtCosWeight

Performs square-root of the cosine of the latitude weighting on the given array.

sqsort

Sorts a singly dimensioned arrays of strings.

stat2

Calculates the first two moments of the given input.

stat4

Calculates estimates of the first four moments (mean, variance, skewness, and kurtosis) of the given input.

stat_dispersion

Computes a number of robust statistics.

stat_medrng

Calculates median, range, and mid-range of the given input.

stat_trim

Calculates trimmed estimates of the first two moments of the given input.

stddev

Calculates the sample standard deviation.

student_t

Calculates the two-tailed probability of the Student-t distribution.

taper

Applies split-cosine-bell tapering to one or more series across the rightmost dimension.

taper_n

Applies split-cosine-bell tapering to one or more series across the given dimension.

trend_manken_n

Mann-Kendall non-parametric test for monotonic trend plus the Theil-Sen robust estimate of linear trend.

ttest

Returns an estimate of the statistical significance and, optionally, the t-values.

variance

Computes an unbiased estimate the variance of all input points.

wave_number_spc

Computes the total power spectrum as a function of latitudinal wave number.

wavelet

Calculates the wavelet transform of a time series and significance levels.

wavelet_default

Calculates the wavelet transform of a time series and significance levels.

wgt_area_smooth

Smooths an array of data using a 5-point 2D area-weighted smoothing algorithm.

wgt_areaave

Calculates the area average of a quantity using weights.

wgt_areaave2

Calculates the area average of a quantity using two-dimensional weights.

wgt_areaave_Wrap

Calculates the area average of a quantity using weights and retains metadata.

wgt_arearmse

Calculates a weighted area root-mean-square-difference between two variables.

wgt_arearmse2

Calculates a weighted area root-mean-square-difference (rmse) between two variables using two-dimensional weights.

wgt_areasum2

Calculates the area sum (total) of a quantity using two-dimensional weights.

wgt_runave

Calculates a weighted running average across the rightmost dimension.

wgt_runave_leftdim

Calculate a weighted running average over the leftmost dimenaion (usually, "time") and return in the original order with meta data.

wgt_runave_n

Calculates a weighted running average across the given dimension.

wgt_runave_n_Wrap

Calculates a weighted running average on the given dimension and retains metadata.

wgt_runave_Wrap

Calculates a weighted running average on the rightmost dimension and retains metadata.

wgt_volave

Calculates the volume average of a quantity using weights.

wgt_volave_ccm

Calculates the volume average of a quantity from the CCM using weights.

wgt_volrmse

Calculates a weighted volume root-mean-square-difference between two variables.

wgt_volrmse_ccm

Calculates a weighted volume root-mean-square-difference between two variables from the CCM.

wk_smooth121

Performs a specialized 1-2-1 filter for Wheeler-Kiladis plots.

zonalAve

Computes a zonal average of the input array

Dimension routines

dim_acumrun_n

Calculates individual accumulated sums of sequences ('runs') of a specified length.

dim_avg

Computes the average of a variable's rightmost dimension at all other dimensions.

dim_avg_n

Computes the average of a variable's given dimension(s) at all other dimensions.

dim_avg_n_Wrap

Computes the average of a variable's given dimensions at all other dimensions and retains metadata.

dim_avg_wgt

Computes the weighted average of a variable's rightmost dimension at all other dimensions.

dim_avg_wgt_n

Computes the weighted average of a variable's given dimension at all other dimensions.

dim_avg_wgt_n_Wrap

Computes the weighted average of a variable's given dimension at all other dimensions and retains metadata.

dim_avg_wgt_Wrap

Computes the weighted average of a variable's rightmost dimension at all other dimensions and retains metadata.

dim_avg_Wrap

Computes the average of a variable's rightmost dimension at all other dimensions and retains metadata.

dim_cumsum

Calculates the cumulative sum along the rightmost dimension.

dim_cumsum_n

Calculates the cumulative sum along the given dimension(s).

dim_cumsum_n_Wrap

Calculates the cumulative sum along the given dimension(s) and retains meta data.

dim_cumsum_Wrap

Calculates the cumulative sum along the rightmost

dimension and retains meta data.

dim_gamfit_n

Fit data to the two parameter gamma distribution.

dim_max

Finds the maximum of a variable's rightmost dimension at all other dimensions.

dim_max_n

Finds the maximum of a variable's given dimensions at all other dimensions.

dim_max_n_Wrap

Computes the maximum of a variable's given dimensions at all other dimensions and retains metadata.

dim_median

Computes the median of a variable's rightmost dimension at all other dimensions.

dim_median_n

Computes the median of a variable's given dimensions at all other dimensions.

dim_min

Finds the minimum of a variable's rightmost dimension at all other dimensions.

dim_min_n

Finds the minimum of a variable's given dimensions at all other dimensions.

dim_min_n_Wrap

Computes the minimum of a variable's given dimensions at all other dimensions and retains metadata.

dim_num

Calculates the number of True values of a variable's rightmost dimension at all other dimensions.

dim_num_n

Calculates the number of True values of a variable's given dimensions at all other dimensions.

dim_numrun_n

Counts the number of "runs" (sequences) within a series containing zeros and ones.

dim_pqsort

Computes the permutation vector generated by sorting the n - 1th (rightmost) dimension.

dim_pqsort_n

Computes the permutation vector generated by sorting the given dimension.

dim_product

Computes the product of a variable's rightmost dimension at all other dimensions.

dim_product_n

Computes the product of a variable's given dimension(s) at all other dimensions.

dim_rmsd

Computes the root-mean-square-difference between two variables' rightmost dimension at all other dimensions.

dim_rmsd_n

Computes the root-mean-square-difference between two variables' given dimensions at all other dimensions.

dim_rmsd_n_Wrap

Computes the root-mean-square-difference between

two variables' given dimensions at all other dimensions.

dim_rmsd_Wrap

Computes the root-mean-square-difference between two variables' rightmost dimension at all other dimensions.

dim_rmvmean

Calculates and removes the mean of the (rightmost) dimension at all other dimensions.

dim_rmvmean_n

Calculates and removes the mean of the given dimension(s) at all other dimensions.

dim_rmvmean_n_Wrap

Calculates and removes the mean of the given dimensions at all other dimensions and retains metadata.

dim_rmvmean_Wrap

Calculates and removes the mean of the (rightmost) dimension at all other dimensions and retains metadata.

dim_rmvmed

Calculates and removes the median of the (rightmost) dimension at all other dimensions.

dim_rmvmed_n

Calculates and removes the median of the given dimension(s) at all other dimensions.

dim_rmvmed_n_Wrap

Calculates and removes the median of the given dimensions at all other dimensions and retains metadata.

dim_rmvmed_Wrap

Calculates and removes the median of the (rightmost) dimension at all other dimensions and retains metadata.

dim_standardize

Calculates standardized anomalies of the rightmost dimension at all other dimensions.

dim_standardize_n

Calculates standardized anomalies of the given dimension(s) at all other dimensions.

dim_standardize_n_Wrap

Calculates standardized anomalies of the given dimensions at all other dimensions and retains metadata.

dim_standardize_Wrap

Calculates standardized anomalies of the rightmost dimension at all other dimensions and retains metadata.

dim_stat4

Computes the first four moments (average, variance, skewness, and kurtosis) of the rightmost dimension for all other dimensions.

dim_stat4_n

Computes the first four moments (average, variance, skewness, and kurtosis) of the given dimension(s) for all other dimensions.

dim_stddev

Computes the population standard deviation of a variable's rightmost dimension at all other dimensions.

dim_stddev_n

Computes the population standard deviation of a variable's given dimension(s) at all other dimensions.

dim_stddev_n_Wrap

Computes the population standard deviation of a variable's given dimension(s) at all other dimensions and retains metadata.

dim_stddev_Wrap

Computes the population standard deviation of a variable's rightmost dimension at all other dimensions and retains metadata.

dim_sum

Computes the arithmetic sum of a variable's rightmost dimension at all other dimensions.

dim_sum_n

Computes the arithmetic sum of a variable's given dimension(s) at all other dimensions.

dim_sum_n_Wrap

Computes the arithmetic sum of a variable's given dimensions at all other dimensions and retains metadata.

dim_sum_wgt

Computes the weighted sum of a variable's rightmost dimension at all other dimensions.

dim_sum_wgt_n

Computes the weighted sum of a variable's given dimension at all other dimensions.

dim_sum_wgt_n_Wrap

Computes the weighted sum of a variable's given dimension at all other dimensions and retains metadata.

dim_sum_wgt_Wrap

Computes the weighted sum of a variable's rightmost dimension at all other dimensions and retains metadata.

dim_sum_Wrap

Computes the arithmetic sum of a variable's rightmost dimension at all other dimensions and retains metadata.

dim_variance

Computes the unbiased estimates of the variance of a variable's rightmost dimension.

dim_variance_n

Computes the unbiased estimates of the variance of a variable's given dimension(s) at all other dimensions.

dim_variance_n_Wrap

Computes unbiased estimates of the variance of a variable's given dimension(s) at all other dimensions and retains metadata.

dim_variance_Wrap

Computes unbiased estimates of the variance of a variable's rightmost dimension at all other dimensions and retains metadata.

Cumulative distribution functions

cdfbin_p

Calculates the binomial density of a cumulative

distribution function.

cdfbin_pr

Calculates the probability of success of each trial of a cumulative distribution function.

cdfbin_s

Calculates the number of successes of a cumulative distribution function.

cdfbin_xn

Calculates the number of binomial trials of a cumulative distribution function.

cdfchi_p

Calculates the integral of a cumulative chi-square distribution function.

cdfchi_x

Calculates the upper limit of integration of a cumulative chi-square distribution function.

cdfgam_p

Calculates the integral of a cumulative gamma distribution function.

cdfgam_x

Calculates the upper limit of integration of a cumulative gamma distribution function.

cdfnor_p

Calculates the integral of a cumulative normal distribution function.

cdfnor_x

Calculates the upper limit of integration of a cumulative normal distribution function.

Empirical orthogonal functions

eof2data

Reconstructs a data set from EOFs and EOF time series.

eofcor

Calculates empirical orthogonal functions via a correlation matrix (NCL's original function).

eofcor_ts

Calculates the time series of the amplitudes associated with each eigenvalue in an EOF which was calculated using a correlation matrix.

eofcor_Wrap

Calculates empirical orthogonal functions and retains meta data. (NCL's original function).

eofcov

Calculates empirical orthogonal functions via a covariance matrix (original version).

eofcov_ts

Calculates the time series of the amplitudes associated with each eigenvalue in an EOF which was calculated using a covariance matrix.

eofcov_Wrap

Calculates empirical orthogonal functions and retains meta data. (NCL's original EOF function).

eofunc

Compute empirical orthogonal functions (EOFs, aka: Principal Component Analysis).

eofunc_ts

Calculates the time series of the amplitudes associated with each eigenvalue in an EOF.

eofunc_ts_Wrap

Calculates the time series of the amplitudes associated with each eigenvalue in an EOF and retains metadata.

eofunc_varimax

Rotates EOFs using the using Kaiser row normalization and the varimax criterion.

eofunc_varimax_reorder

Reorder the results returned by eof_varimax into descending order by percent variance explained.

eofunc_varimax_Wrap

Rotates EOFs using the using Kaiser row normalization and the varimax criterion and retains metadata.

eofunc_Wrap

Computes empirical orthogonal functions (aka: Principal Component Analysis, Eigen Analysis) and retains metadata.

Group creators and query

filegrpdef

Defines a list of group (names), for a supported file, or group.

getfilegrpnames

Returns an array of file group names in the specified supported file.

Interpolation

area_conserve_remap

Performs areal conservative remapping from one rectilinear grid to another.

area_conserve_remap_Wrap

Performs areal conservative remapping from one rectilinear grid to another and preserves meta data.

area_hi2lores

Interpolates from high resolution rectilinear grids to low resolution recitlinear grids using local area averaging.

area_hi2lores_Wrap

Interpolates from high resolution rectilinear grids to low resolution rectilinear grids using local area averaging. (retains meta data)

bin_avg

Calculates gridded binned averages and counts onto a rectilinear grid using randomly spaced data.

csa1

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa1d

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa1s

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa1x

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa1xd

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa1xs

Calculates an approximating cubic spline for the input data, one 1-dimensional section at a time.

csa2

Calculates an approximating cubic spline for two-dimensional input data.

csa2d

Calculates an approximating cubic spline for two-dimensional input data.

csa2l

Calculates an approximating cubic spline for two-dimensional input data.

csa2ld

Calculates an approximating cubic spline for two-dimensional input data.

csa2ls

Calculates an approximating cubic spline for two-dimensional input data.

csa2lx

Calculates an approximating cubic spline for two-dimensional input data.

csa2lxd

Calculates an approximating cubic spline for two-dimensional input data.

csa2lxs

Calculates an approximating cubic spline for two-dimensional input data.

csa2s

Calculates an approximating cubic spline for two-dimensional input data.

csa2x

Calculates an approximating cubic spline for two-dimensional input data.

csa2xd

Calculates an approximating cubic spline for two-dimensional input data.

csa2xs

Calculates an approximating cubic spline for two-dimensional input data.

csa3

Calculates an approximating cubic spline for three-dimensional input data.

csa3d

Calculates an approximating cubic spline for three-dimensional input data.

csa3l

Calculates an approximating cubic spline for three-dimensional input data.

csa3ld

Calculates an approximating cubic spline for three-dimensional input data.

csa3ls

Calculates an approximating cubic spline for three-dimensional input data.

csa3lx

Calculates an approximating cubic spline for three-dimensional input data.

csa3lxd

Calculates an approximating cubic spline for three-dimensional input data.

csa3lxs

Calculates an approximating cubic spline for three-dimensional input data.

csa3s

Calculates an approximating cubic spline for three-dimensional input data.

csa3x

Calculates an approximating cubic spline for three-dimensional input data.

csa3xd

Calculates an approximating cubic spline for three-dimensional input data.

csa3xs

Calculates an approximating cubic spline for three-dimensional input data.

csc2s

Converts Cartesian coordinates on a unit sphere to spherical coordinates (lat/lon).

csgetp

Retrieves control parameters for Cssgrid routines.

css2c

Converts spherical coordinates (lat/lon) to Cartesian coordinates on a unit sphere.

cssetp

Sets control parameters for Cssgrid routines.

cssgrid

Uses tension splines to interpolate unstructured (randomly-spaced) data on a unit sphere to data values on a rectilinear grid.

cssgrid_Wrap

Uses tension splines to interpolate unstructrued (randomly-spaced) data on a unit sphere to data values on a rectilinear grid (retains metadata).

csstri

Calculates a Delaunay triangulation of data randomly positioned on the surface of a sphere.

csvoro

Determines Voronoi polygons for data randomly positioned on a sphere and returns vertices for the one surrounding a specified input point.

curvilinear_to_SCRIP

Writes the description of a curvilinear grid to a SCRIP file.

dsgetp

Gets parameters for Dsgrid routines.

dsgrid2

Interpolates data from an unstructured (randomly-spaced) grid to a rectilinear grid using inverse distance weighted interpolation.

dsgrid3

Interpolates float data from an unstructured (randomly-spaced) grid to a 3D grid using inverse distance weighted interpolation.

dspnt2

Interpolates 2-D data at specified points.

dspnt3

Interpolates 3-D data at specified points.

dssetp

Sets parameters for Dsgrid routines.

ESMF_regrid

Regrids data from one lat/lon grid to another, using ESMF software.

ESMF_regrid_gen_weights

Writes a weight file using the offline ESMF weight generator.

ESMF_regrid_with_weights

Using the provided weight file, regrids data from one lat/lon grid to another.

ftcurv

Calculates an interpolatory spline through a sequence of functional values.

ftcurvd

Calculates the derivatives of an interpolatory spline under tension.

ftcurvi

Calculates integrals of an interpolatory spline under tension between two user-specified limits.

ftcurvp

Calculates an interpolatory spline under tension through a sequence of functional values for a periodic function.

ftcurvpi

Calculates an integral of an interpolatory spline between two specified points.

ftcurvps

Calculates a smoothing spline.

ftcurvs

Calculates a smoothing spline.

ftgetp

Retrieves control parameters for Fitgrid routines.

ftkurv

Calculates an interpolatory spline for parametric curves.

ftkurvd

Calculates an interpolatory spline for parametric curves; it also calculates first and second derivatives of the interpolatory spline.

ftkurvp

Calculates an interpolatory spline under tension through a sequence of points in the plane forming a closed curve.

ftkurvpd

Calculates an interpolatory spline for closed parametric curves; it also calculates first and second derivatives of the interpolatory spline.

ftsetp

Sets control parameters for Fitgrid routines.

idsfft

Interpolates 2D random data.

int2p

Interpolates pressure levels to a different set of pressure levels.

int2p_n

Interpolates pressure levels to a different set of pressure levels on the given dimension.

int2p_n_Wrap

Interpolates pressure levels to a different set of pressure levels and returns meta data.

int2p_Wrap

Interpolates pressure levels to a different set of

pressure levels and returns meta data.

latlon_to_SCRIP
Writes the description of a lat/lon grid to a SCRIP file.

linint1
Interpolates from one series to another using piecewise linear interpolation across the rightmost dimension.

linint1_n
Interpolates from one series to another using piecewise linear interpolation across the given dimension.

linint1_n_Wrap
Interpolates from one series to another using piecewise linear interpolation across the given dimension, and retains metadata.

linint1_Wrap
Interpolates from one series to another using piecewise linear interpolation, and retains metadata.

linint2
Interpolates from a rectilinear grid to another rectilinear grid using bilinear interpolation.

linint2_points
Interpolates from a rectilinear grid to an unstructured grid using bilinear interpolation.

linmsg
Linearly interpolates to fill in missing values.

linmsg_n
Linearly interpolates to fill in missing values, given the dimension to do the interpolation across.

natgrid
Interpolates data from an unstructured (randomly spaced) grid to a rectilinear grid using natural neighbor interpolation.

natgrid_Wrap
Interpolates data from an unstructured (randomly-spaced) grid to a rectilinear grid using natural neighbor interpolation and retains metadata.

nngetaspectd
Retrieves an aspect at a specified coordinate position.

nngetaspects
Retrieves an aspect at a specified coordinate position.

nngetp
Retrieves control parameters for Natgrid routines.

nngetsloped
Retrieves a slope at a specified coordinate position.

nngetslopes
Retrieves a slope at a specified coordinate position.

nngetwts
Retrieves natural neighbors and weights for the function values at those neighbors.

nngetwtsd
Retrieves natural neighbors and weights for the function values at those neighbors.

nnpnt
Interpolates from 2D random data to get values at a specified set of points.

nnpntend
Terminates interpolation from 2D random data to get values at a specified set of points.

nnpntendd
Terminates interpolation from 2D random data to get values at a specified set of points.

nnpntinit
Initializes internal quantities, for given input data, to allow subsequent nnpnt calls to interpolate at specified points.

nnpntinitd
Initializes internal quantities, for given input data, to allow subsequent nnpntd calls to interpolate at specified points.

nnpntinits
Initializes internal quantities, for given input data, to allow subsequent nnpnts calls to interpolate at specified points.

nnsetp
Sets control parameters for Natgrid routines.

obj_anal_ic
Iterative improvement objective analysis.

obj_anal_ic_deprecated
Iterative correction objective analysis (Cressman, Barnes).

obj_anal_ic_Wrap
Iterative improvement objective analysis and returns meta data.

poisson_grid_fill
Replaces all missing (_FillValue) values in a grid with values derived from solving Poisson's equation via relaxation.

rcm2points
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to an unstructured grid.

rcm2rgrid
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to a rectilinear grid.

rectilinear_to_SCRIP
Writes the description of a rectilinear grid to a SCRIP file.

rgrid2rcm
Interpolates data on a rectilinear lat/lon grid to a curvilinear grid like those used by the RCM, WRF and NARR models/datasets.

shgetnp
Finds the nearest point, or points (in a specified set of points), to a given point in 3-space.

shgetp
Retrieves control parameters for Shgrid routines.

shgrid
Interpolates data from an unstructured grid to a grid in 3-space.

shsetp
Sets control parameters for Shgrid routines.

sigma2hybrid
Interpolates from sigma to hybrid coordinates.

trop_wmo
Determines the level of the thermal tropopause.

unstructured_to_ESMF
Writes the description of an unstructured grid to an ESMF file.

wrf_interp_1d
Linearly interpolates a one-dimensional variable in the vertical.

wrf_interp_2d_xy
Extracts a cross section from a given input field.

wrf_interp_3d_z
Interpolates to a specified pressure/height level.

wrf_user_intrp2d
Interpolates ARW WRF 2D model data along a give line.

wrf_user_intrp3d
Interpolates ARW WRF model data vertically or horizontally.

wrf_user_unstagger
Unstaggers an input variable along a specified dimension

Random number generators

generate_2d_array
Generates a "nice" 2D array of pseudo random data, especially for use in 2D graphics.

generate_resample_indices
Generate indices (subscripts) for resampling: with and without replacement.

generate_sample_indices
Generate indices (subscripts) for resampling: with and without replacement.

generate_unique_indices
Generate unique random indices (subscripts).

rand
Generates a pseudo random number.

random_chi
Generates random numbers using a chi-squared distribution.

random_gamma
Generates random numbers using a gamma distribution.

random_normal
Generates random numbers using a normal distribution.

random_setallseed
Sets initial seeds for random number generators.

random_uniform
Generates random numbers using a uniform range distribution.

srand
Establishes a seed for the rand function.

Regridding

area_conserve_remap
Performs areal conservative remapping from one rectilinear grid to another.

area_conserve_remap_Wrap
Performs areal conservative remapping from one rectilinear grid to another and preserves meta data.

area_hi2lores
Interpolates from high resolution rectilinear grids to

low resolution recitlinear grids using local area averaging.

area_hi2lores_Wrap
Interpolates from high resolution rectilinear grids to low resolution rectilinear grids using local area averaging. (retains meta data)

bin_avg
Calculates gridded binned averages and counts onto a rectilinear grid using randomly spaced data.

curvilinear_to_SCRIP
Writes the description of a curvilinear grid to a SCRIP file.

ESMF_regrid
Regrids data from one lat/lon grid to another, using ESMF software.

ESMF_regrid_gen_weights
Writes a weight file using the offline ESMF weight generator.

ESMF_regrid_with_weights
Using the provided weight file, regrids data from one lat/lon grid to another.

f2fosh
Interpolates a scalar quantity from a fixed grid (including pole points) to a fixed-offset grid.

f2fosh_Wrap
Interpolates a scalar quantity from a fixed grid (including pole points) to a fixed-offset grid (retains metadata).

f2foshv
Interpolates a vector quantity on a fixed grid (including pole points) to a fixed-offset grid.

f2foshv_Wrap
Interpolates a vector quantity on a fixed grid (including pole points) to a fixed-offset grid (retains metadata).

f2fsh
Interpolates a scalar quantity from one fixed grid to another.

f2fsh_Wrap
Interpolates a scalar quantity from one fixed grid to another (retains metadata).

f2fshv
Interpolates a vector quantity from one fixed grid to another.

f2fshv_Wrap
Interpolates a vector quantity from one fixed grid to another (retains metadata).

f2gsh
Interpolates a scalar quantity from a fixed grid to a Gaussian grid (with optional truncation).

f2gsh_Wrap
Interpolates a scalar quantity from a fixed grid to a Gaussian grid (with optional truncation) (retains metadata).

f2gshv
Interpolates a vector quantity on a fixed grid to a Gaussian grid (optional truncation).

f2gshv_Wrap
Interpolates a vector quantity on a fixed grid to a Gaussian grid (optional truncation) (retains

metadata).

fo2fsh
Interpolates a scalar quantity from a fixed-offset grid to a fixed grid.

fo2fsh_Wrap
Interpolates a scalar quantity from a fixed-offset grid to a fixed grid (retains metadata).

fo2fshv
Interpolates a vector quantity from a fixed-offset grid to a fixed grid.

fo2fshv_Wrap
Interpolates a vector quantity from a fixed-offset grid to a fixed grid (retains metadata).

ftsurf
Calculates an interpolatory surface passing through a rectangular grid of function values.

g2fsh
Interpolates a scalar quantity from a Gaussian grid to a fixed grid.

g2fsh_Wrap
Interpolates a scalar quantity from a Gaussian grid to a fixed grid (retains metadata).

g2fshv
Interpolates a vector quantity from a Gaussian grid to a fixed grid.

g2fshv_Wrap
Interpolates a vector quantity from a Gaussian grid to a fixed grid (retains metadata).

g2gsh
Interpolates a scalar quantity from one Gaussian grid to another (with optional truncation).

g2gsh_Wrap
Interpolates a scalar quantity from one Gaussian grid to another (with optional truncation) (retains metadata).

g2gshv
Interpolates a vector quantity from one Gaussian to another (optional truncation).

g2gshv_Wrap
Interpolates a vector quantity from one Gaussian to another (optional truncation) (retains metadata).

latlon_to_SCRIP
Writes the description of a lat/lon grid to a SCRIP file.

linint2
Interpolates from a rectilinear grid to another rectilinear grid using bilinear interpolation.

linint2_points
Interpolates from a rectilinear grid to an unstructured grid using bilinear interpolation.

linint2_points_Wrap
Interpolates from a rectilinear grid to an unstructured grid using bilinear interpolation, and retains metadata.

linint2_Wrap
Interpolates from a rectilinear grid to another rectilinear grid using bilinear interpolation, and retains metadata.

obj_anal_ic
Iterative improvement objective analysis.

obj_anal_ic_deprecated
Iterative correction objective analysis (Cressman, Barnes).

obj_anal_ic_Wrap
Iterative improvement objective analysis and returns meta data.

poisson_grid_fill
Replaces all missing (_FillValue) values in a grid with values derived from solving Poisson's equation via relaxation.

pop_remap
Regrids a POP ocean model grid to another grid.

PopLatLon
Regrids a scalar variable on a POP grid to a lat/lon grid or vice-versa.

PopLatLonV
Converts vectors on a POP grid to a lat/lon grid and vice-versa.

rcm2points
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to an unstructured grid.

rcm2points_Wrap
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to an unstructured grid and retains metadata.

rcm2rgrid
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to a rectilinear grid.

rcm2rgrid_Wrap
Interpolates data on a curvilinear grid (i.e. RCM, WRF, NARR) to a rectilinear grid and retains metadata.

rectilinear_to_SCRIP
Writes the description of a rectilinear grid to a SCRIP file.

rgrid2rcm
Interpolates data on a rectilinear lat/lon grid to a curvilinear grid like those used by the RCM, WRF and NARR models/datasets.

rgrid2rcm_Wrap
Interpolates data on a rectilinear lat/lon grid to a curvilinear grid like those used by the RCM, WRF and NARR models/datasets and retains metadata.

triple2grid
Places unstructured (randomly-spaced) data onto the nearest locations of a rectilinear grid.

triple2grid2d
Places unstructured (randomly-spaced) data onto the nearest locations of a curvilinear grid.

triple2grid_Wrap
Places unstructured (randomly-spaced) data onto the nearest locations of a rectilinear grid and retains metadata.

unstructured_to_ESMF
Writes the description of an unstructured grid to an ESMF file.

RIP

wrf_cape_2d
Computes maximum convective available potential

energy (CAPE), maximum convective inhibition (CIN), lifted condensation level (LCL), and level of free convection (LFC).

wrf_cape_3d
Computes convective available potential energy (CAPE) and convective inhibition (CIN).

Singular value decomposition

dgeevx_lapack
Given a square (N,N) real nonsymmetric matrix, compute the eigenvalues and, optionally, the left and/or right eigenvectors via the LAPACK subroutine dgeevx.

svd_lapack
Calculates the singular value decomposition of a general rectangular matrix.

svdcov
Uses singular value decomposition and returns the left and right homogeneous and heterogeneous arrays associated with the two input datasets.

svdcov_sv
Uses singular value decomposition to return the left and right singular vectors associated with the two input datasets.

svdstd
Uses singular value decomposition and returns the left and right homogeneous and heterogeneous arrays associated with the two input datasets.

svdstd_sv
Uses singular value decomposition to return the left and right singular vectors associated with the two input datasets.

Spherical harmonic routines

dv2uvF
Computes the divergent (irrotational) wind components for a fixed grid via spherical harmonics.

dv2uvf
Computes the divergent (irrotational) wind components for a fixed grid via spherical harmonics.

dv2uvF_Wrap
Computes the divergent (irrotational) wind components for a fixed grid via spherical harmonics and retains metadata.

dv2uvG
Computes the divergent (irrotational) wind components for a gaussian grid via spherical harmonics.

dv2uvg
Computes the divergent (irrotational) wind components for a gaussian grid via spherical harmonics.

dv2uvG_Wrap
Computes the divergent (irrotational) wind components for a gaussian grid via spherical harmonics and retains metadata.

exp_tapersh
Performs tapering (filtering) of the spherical harmonic coefficients.

exp_tapersh_wgts
Calculates weights which can be used to perform tapering (filtering) of spherical harmonic coefficients.

exp_tapershC
Performs tapering (filtering) of the spherical harmonic coefficients.

get_sphere_radius
Retrieves the radius of a sphere used in spherical harmonic routines.

gradsf
Computes the gradient of an array that is on a fixed grid using spherical harmonics.

gradsg
Computes the gradient of an array that is on a gaussian grid using spherical harmonics.

igradsf
Computes a scalar array from its gradient components on a fixed grid using spherical harmonics.

igradsF
Computes a scalar array from its gradient components on a fixed grid using spherical harmonics.

igradsg
Computes a scalar array from its gradient components on a gaussian grid using spherical harmonics.

igradsG
Computes a scalar array from its gradient components on a gaussian grid using spherical harmonics.

ilapsf
Inverts the Laplacian (on a fixed grid) using spherical harmonics.

ilapsF
Inverts the Laplacian (on a fixed grid) using spherical harmonics.

ilapsF_Wrap
Inverts the Laplacian (on a fixed grid) using spherical harmonics and retains metadata.

ilapsg
Inverts the Laplacian (on a gaussian grid) using spherical harmonics.

ilapsG
Inverts the Laplacian (on a gaussian grid) using spherical harmonics.

ilapsG_Wrap
Inverts the Laplacian (on a gaussian grid) using spherical harmonics and retains metadata.

ilapvf
Inverts the vector Laplacian (on a fixed grid) using spherical harmonics.

ilapvg
Inverts the vector Laplacian (on a gaussian grid) using spherical harmonics.

lapsF
Computes the Laplacian using spherical harmonics, given a scalar z on a fixed grid.

lapsf
Computes the Laplacian using spherical harmonics,

given a scalar z on a fixed grid.

lapsG
Computes the Laplacian using spherical harmonics, given a scalar z on a gaussian grid.

lapsg
Computes the Laplacian using spherical harmonics, given a scalar z on a gaussian grid.

lapvf
Computes the vector Laplacian using spherical harmonics, given a vector quantity (u,v) on a fixed grid.

lapvg
Computes the vector Laplacian using spherical harmonics, given a vector quantity (u,v) on a gaussian grid.

lderuvf
Computes the latitudinal derivatives using spherical harmonics, given vector components (u,v) on a fixed grid.

lderuvg
Computes the latitudinal derivatives using spherical harmonics, given vector components (u,v) on a gaussian grid.

rhomb_trunc
Performs rhomboidal truncation on spherical harmonic coefficients.

rhomb_trunC
Performs rhomboidal truncation on spherical harmonic coefficients.

set_sphere_radius
Sets the radius of a sphere used in spherical harmonic routines.

sfvp2uvf
Computes the wind components given stream function and velocity potential (on a fixed grid) via spherical harmonics.

sfvp2uvg
Computes the wind components given stream function and velocity potential (on a gaussian grid) via spherical harmonics.

shaeC
Computes spherical harmonic analysis of a scalar field on a fixed grid via spherical harmonics.

shaec
Computes spherical harmonic analysis of a scalar field on a fixed grid via spherical harmonics.

shagC
Computes spherical harmonic analysis of a scalar field on a gaussian grid via spherical harmonics.

shagc
Computes spherical harmonic analysis of a scalar field on a gaussian grid via spherical harmonics.

shsec
Computes spherical harmonic synthesis of a scalar quantity on a fixed grid via spherical harmonics.

shseC
Computes the spherical harmonic synthesis of a scalar quantity on a fixed grid via spherical harmonics.

shsgc
Computes spherical harmonic synthesis of a scalar quantity on a gaussian grid via spherical harmonics.

shsgC
Computes the spherical harmonic synthesis of a scalar quantity on a gaussian grid via spherical harmonics.

shsgc_R42
Computes spherical harmonic synthesis of a scalar quantity via rhomboidally truncated (R42) spherical harmonic coefficients onto a (108x128) gaussian grid.

shsgc_R42_Wrap
Computes spherical harmonic synthesis of a scalar quantity via rhomboidally truncated (R42) spherical harmonic coefficients onto a (108x128) gaussian grid. (creates metadata).

tri_trunC
Performs triangular truncation on spherical harmonic coefficients.

tri_trunc
Performs triangular truncation on spherical harmonic coefficients.

uv2dvf
Computes the divergence using spherical harmonics given the u and v wind components on a fixed grid.

uv2dvF
Computes the divergence using spherical harmonics given the u and v wind components on a fixed grid.

uv2dvF_Wrap
Computes the divergence using spherical harmonics given the u and v wind components on a fixed grid (retains metadata).

uv2dvg
Computes the divergence using spherical harmonics given the u and v wind components on a gaussian grid.

uv2dvG
Computes the divergence using spherical harmonics given the u and v wind components on a gaussian grid.

uv2dvG_Wrap
Computes the divergence using spherical harmonics given the u and v wind components on a gaussian grid (retains metadata).

uv2sfvpF
Computes the stream function and velocity potential via spherical harmonics given u and v on a fixed grid.

uv2sfvpf
Computes the stream function and velocity potential via spherical harmonics given u and v on a fixed grid.

uv2sfvpG
Computes the stream function and velocity potential via spherical harmonics given u and v on a gaussian grid.

uv2sfvpg
Computes the stream function and velocity potential via spherical harmonics given u and v on a gaussian

grid.

uv2vrdvF
Computes the vorticity and divergence via spherical harmonics, given the u and v wind components on a fixed grid.

uv2vrdvf
Computes the vorticity and divergence via spherical harmonics, given the u and v wind components on a fixed grid.

uv2vrdvg
Computes the vorticity and divergence via spherical harmonics, given the u and v wind components on a gaussian grid.

uv2vrdvG
Computes the vorticity and divergence via spherical harmonics, given the u and v wind components on a gaussian grid.

uv2vrF
Computes the vorticity via spherical harmonics, given the u and v wind components on a fixed grid.

uv2vrf
Computes the vorticity via spherical harmonics, given the u and v wind components on a fixed grid.

uv2vrF_Wrap
Computes the vorticity via spherical harmonics, given the u and v wind components on a fixed grid (retains metadata).

uv2vrG
Computes the vorticity via spherical harmonics, given the u and v wind components on a gaussian grid.

uv2vrg
Computes the vorticity via spherical harmonics, given the u and v wind components on a gaussian grid.

uv2vrG_Wrap
Computes the vorticity via spherical harmonics, given the u and v wind components on a gaussian grid (retains metadata).

vhaeC
Computes vector spherical harmonic analysis of vector fields on a fixed grid via spherical harmonics.

vhaec
Computes vector spherical harmonic analysis of vector fields on a fixed grid via spherical harmonics.

vhagC
Computes vector spherical harmonic analysis of vector fields on a gaussian grid via spherical harmonics.

vhagc
Computes vector spherical harmonic analysis of vector fields on a gaussian grid via spherical harmonics.

vhseC
Computes vector spherical harmonic syntheses of vector fields on a fixed grid via spherical harmonics.

vhsec
Computes vector spherical harmonic syntheses of vector fields on a fixed grid via spherical harmonics.

vhsgc
Computes vector spherical harmonic syntheses of vector fields on a gaussian grid via spherical harmonics.

vhsgC
Computes vector spherical harmonic syntheses of vector fields on a gaussian grid via spherical harmonics.

vr2uvf
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a fixed grid.

vr2uvF
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a fixed grid.

vr2uvF_Wrap
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a fixed grid (retains metadata).

vr2uvg
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a gaussian grid.

vr2uvG
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a gaussian grid.

vr2uvG_Wrap
Computes the rotational wind components via spherical harmonics, given an array containing relative vorticity on a gaussian grid (retains metadata).

vrdv2uvf
Computes the wind components via spherical harmonics, given vorticity and divergence on a fixed grid.

vrdv2uvF
Computes the wind components via spherical harmonics, given vorticity and divergence on a fixed grid.

vrdv2uvg
Computes the wind components via spherical harmonics, given vorticity and divergence on a gaussian grid.

vrdv2uvG
Computes the wind components via spherical harmonics, given vorticity and divergence on a gaussian grid.

Lat/lon functions
add90LatX
Adds two fake pole points (90S and 90N) to the rightmost dimension of the given data.

add90LatY
Adds two fake pole points (90S and 90N) to the leftmost dimension of the given data.

area_poly_sphere
Calculates the area enclosed by an arbitrary polygon on the sphere.

bin_avg

Calculates gridded binned averages and counts onto a rectilinear grid using randomly spaced data.

bin_sum

Calculates binned sums and counts over multiple invocations of the procedure onto a ectilinear grid.

gaus

Computes gaussian latitudes and weights.

gaus_lobat

Computes gaussian latitudes and weights using Gauss-Lobatto quadrature.

gaus_lobat_wgt

Computes Gauss-Lobatto weights given a one-dimensional array of Gauss-Lobatto latitudes.

gc_aangle

Finds the acute angle between two great circles on the globe.

gc_clkwise

Tests clockwise/counterclockwise ordering of points on spherical polygon.

gc_dangle

Finds the directed angle between two great circles having a specified intersection point.

gc_inout

Determines if a list of lat/lon specified points are inside or outside of spherical lat/lon polygon(s).

gc_latlon

Finds the great circle distance (true surface distance) between two points on the globe and interpolates points along the great circle.

gc_onarc

Determines if a point on the globe lies on a specified great circle arc.

gc_pnt2gc

Finds the angular distance from a point to a great circle.

gc_qarea

Finds the area of a quadrilateral patch on the unit sphere.

gc_tarea

Finds the area of a triangular patch on the unit sphere.

getind_latlon2d

Finds the indices (subscripts) of two-dimensional latitude/longitude arrays closest to a user specified latitude/longitude coordinate pair.

landsea_mask

Returns a grid that contains a land sea mask given any latitude and longitude array.

latGau

Generates gaussian latitudes and associated metadata.

latGauWgt

Generates gaussian weights and associated metadata.

latGlobeF

Generates latitudes and associated metadata for a global fixed grid.

latGlobeFo

Generates latitudes and associated metadata for a

global fixed offset grid.

latlon2utm

Converts from lat/lon to UTM using a specified datum.

latRegWgt

Generates [sin(lat+dlat/2)-sin(lat-dlat/2)] weights for equally spaced (regular) global grids that will sum to 2.0.

lonFlip

Reorders an array about the central longitude coordinate variable (rectilinear grids only).

lonGlobeF

Generates longitudes and associated metadata for a global fixed grid.

lonGlobeFo

Generates longitudes and associated metadata for a global fixed offset grid.

lonPivot

Pivots about a user specified longitude (rectilinear grids only).

nggcog

Calculates the latitudes and longitudes of a set of points approximating a circle at a given point on the surface of the globe.

niceLatLon2D

Check two dimensional map coordinates to see if they have a "nice" structure.

NormCosWgtGlobe

Create normalized cosine weights that sum to 2.0.

plt_pdfxy

Creates a nice plot of the joint probability array created by the pdfxy function.

region_ind

Returns the indices (subscripts) of two-dimensional latitude/longitude arrays that span user specified latitude/longitude boundaries.

utm2latlon

Converts from UTM to lat/lon using a specified datum.

wrf_ij_to_ll

Finds the nearest longitude, latitude locations to the specified model grid indices (i,j).

wrf_ll_to_ij

Finds the nearest model grid indices (i,j) to the specified location(s) in longitude and latitude.

wrf_user_ij_to_ll

Finds the nearest longitude, latitude locations to the specified model grid indices (i,j).

wrf_user_ll_to_ij

Finds the nearest model grid indices (i,j) to the specified location(s) in longitude and latitude.

Metadata routines

assignFillValue

Transfers the _FillValue attribute from one variable to another.

copy_VarAtts

Copies all of a variable's attributes from one variable to another.

copy_VarCoords

Copies all named dimensions and coordinate variables from one variable to another.

copy_VarCoords_1

Copies all named dimensions and coordinate variables from one variable to another except for the rightmost dimension.

copy_VarCoords_2

Copies all named dimensions and coordinate variables from one variable to another except for the rightmost two dimensions.

copy_VarMeta

Copies all attributes, named dimensions and coordinate variables from one variable to another.

default_fillvalue

Returns the default missing value for the given variable type.

delete_VarAtts

Deletes one or more attributes associated with a variable.

getFillValue

Retrieves the _FillValue of a variable, if present.

getvaratts

Returns a list of attribute names for the given variable or file pointer.

getvardims

Returns a list of dimension names for the given variable.

getVarFillValue

Retrieves the missing value of a variable, otherwise, it returns the default _FillValue.

isatt

Returns logical values indicating whether the given attributes are attached to the given variable.

iscoord

Returns True for every input string that is a coordinate variables of the given variable.

isdim

Returns True if variable dimensions are defined in the given variable.

isdimnamed

Returns True if variable dimensions have names in given variable.

ismissing

Returns True for every element of the input that contains a missing value.

isunlimited

Returns True if the given dimension name is defined as unlimited on the given file.

nameDim

Assigns given named dimensions, long_name, and units to the input variable.

set_default_fillvalue

Sets the default missing value for the given variable type.

Meteorology

angmom_atm

Calculates the atmosphere's relative angular momentum.

dewtemp_trh
>Calculates the dew point temperature given temperature and relative humidity.

fluxEddy
>Calculates time averaged eddy flux quantities.

hydro
>Computes geopotential height using the hydrostatic equation.

hyi2hyo
>Interpolates from data on one set of hybrid levels to another set of hybrid levels.

hyi2hyo_Wrap
>Interpolates from data on one set of hybrid levels to another set of hybrid levels and preserves metadata.

kf_filter
>Extract equatorial waves by filtering in the Wheeler-Kiladis wavenumber-frequency domain.

lclvl
>Calculates the pressure of the lifting condensation level.

mixhum_ptd
>Calculates the mixing ratio or specific humidity given pressure and dew point temperature.

mixhum_ptrh
>Calculates the mixing ratio or specific humidity given pressure, temperature and relative humidity.

omega_to_w
>Convert omega vertical velocity (Pa/s) to (m/s).

prcwater_dp
>Computes column precipitable water of the rightmost dimension.

pres2hybrid
>Interpolates data on constant pressure levels to hybrid levels.

pres2hybrid_Wrap
>Interpolates data on constant pressure levels to hybrid levels and retains metadata.

pres_sigma
>Calculates the pressure at specified sigma levels.

pslec
>Computes sea level pressure from CCM/CAM hybrid model variables using the ECMWF formulation.

pslhor
>Computes sea level pressure from CCM/CAM hybrid model variables using the ECMWF formulation and Trenberth's horizontal correction.

pslhyp
>Computes sea level pressure using the hypsometric equation.

rdsstoi
>Reads weekly/monthly compocp site and climatology grids.

relhum
>Calculates relative humidity given temperature, mixing ratio, and pressure.

relhum_ttd
>Calculates relative humidity given temperature and dew point temperature.

stdatmus_p2tdz
>Calculates the corresponding temperature, density,

and height based on the 1976 U.S. standard atmosphere, given the pressure.

stdatmus_z2tdp
>Calculates the corresponding temperature, density, and pressure based on the 1976 U.S. standard atmosphere, given the height.

uv2dv_cfd
>Computes divergence using centered finite differences.

uv2vr_cfd
>Computes relative vorticity using centered finite differences.

vibeta
>Performs vertical integration using beta factors.

w_to_omega
>Convert vertical velocity with units (m/s) to Pa/s.

wind_component
>Calculate zonal and meridional wind components from wind speed and wind direction.

wind_direction
>Calculate meteorological wind direction from zonal and meridional wind components.

z2geouv
>Computes the geostrophic zonal and meridional wind components using geopotential height on isobaric levels (rectilinear grid only).

zonal_mpsi
>Computes a zonal mean meridional stream function.

zonal_mpsi_Wrap
>Computes a zonal mean meridional stream function and retains metadata.

Climatology

calcDayAnomTLL
>Calculates daily anomalies from a daily data climatology.

calcMonAnomLLLT
>Calculates monthly anomalies by subtracting the long term mean from each point (lev,lat,lon,time version)

calcMonAnomLLT
>Calculates monthly anomalies by subtracting the long term mean from each point (lat,lon,time version)

calcMonAnomTLL
>Calculates monthly anomalies by subtracting the long term mean from each point (time,lat,lon version)

calcMonAnomTLLL
>Calculates monthly anomalies by subtracting the long term mean from each point: (time,lev,lat,lon) version.

clmDayTLL
>Calculates long term daily means (daily climatology) from daily data.

clmDayTLLL
>Calculates long term daily means (daily climatology) from daily data.

clmMon2clmDay
>Create a daily climatology from a monthly climatology.

clmMonLLLT
>Calculates long term monthly means (monthly climatology) from monthly data: (lev,lat,lon,time) version.

clmMonLLT
>Calculates long term monthly means (monthly climatology) from monthly data (lat,lon,time version)

clmMonTLL
>Calculates long term monthly means (monthly climatology) from monthly data: (time,lat,lon) version

clmMonTLLL
>Calculates long term monthly means (monthly climatology) from monthly data: (time,lev,lat,lon) version

month_to_season
>Computes a user-specified three-month seasonal mean (DJF, JFM, FMA, MAM, AMJ, MJJ, JJA, JAS, ASO, SON, OND, NDJ).

month_to_season12
>Computes three-month seasonal means (DJF, JFM, FMA, MAM, AMJ, MJJ, JJA, JAS, ASO, SON, OND, NDJ).

month_to_seasonN
>Computes a user-specified list of three-month seasonal means (DJF, JFM, FMA, MAM, AMJ, MJJ, JJA, JAS, ASO, SON, OND, NDJ).

rmAnnCycle1D
>Removes annual cycle from a one-dimensional time series.

rmMonAnnCycLLLT
>Removes the annual cycle from "monthly" data.

rmMonAnnCycLLT
>Removes the annual cycle from "monthly" data.

rmMonAnnCycTLL
>Removes the annual cycle from "monthly" data.

smthClmDayTLL
>Calculates a smooth mean daily annual cycle for an array nominally dimensioned (Time,Lat,Lon).

smthClmDayTLLL
>Calculates a smooth mean daily annual cycle for an array nominally dimensioned (Time,Level,Lat,Lon).

stdMonLLLT
>Calculates standard deviations of monthly means.

stdMonLLT
>Calculates standard deviations of monthly means.

stdMonTLL
>Calculates standard deviations of monthly means.

stdMonTLLL
>Calculates standard deviations of monthly means.

Oceanography

depth_to_pres
>Convert ocean depth to pressure.

mixed_layer_depth
>Calculates a quantity derived from the Potential Density of significance. (Documentation under construction.)

pop_remap
>Regrids a POP ocean model grid to another grid.

PopLatLon

Regrids a scalar variable on a POP grid to a lat/lon grid or vice-versa.

PopLatLonV

Converts vectors on a POP grid to a lat/lon grid and vice-versa.

potmp_insitu_ocn

Calculate seawater potential temperature at an arbitrary reference pressure given insitu temperature, salinity and depth.

rho_mwjf

Computes ocean water density given a specified range for potential temperature (deg Celisus) and salinity (psu).

CESM

angmom_atm

Calculates the atmosphere's relative angular momentum.

band_pass_area_time

Create a time series of area averages; band pass filter the resulting area averaged time series and calculate other statistics.

band_pass_area_time_plot

Generate a plot using the output from band_pass_area_time.

band_pass_hovmueller

Create a band-pass filtered time series suitable for a time vs longitude (Hovmueller) plot.

band_pass_hovmueller_plot

Generate a plot using the output from band_pass_hovmueller.

band_pass_latlon_time

Create band-pass filtered series at each lat/lon grid point.

band_pass_latlon_time_plot

Generate a plot using the output from band_pass_latlon_time.

cz2ccm

Computes geopotential height in hybrid coordinates.

decomposeSymAsym

Decompose a variable which is symmetric about the equator into symmetric and asymmetric parts.

depth_to_pres

Convert ocean depth to pressure.

dpres_hybrid_ccm

Calculates the pressure differences of a hybrid coordinate system.

dpres_plevel

Calculates the pressure layer thicknesses of a constant pressure level coordinate system.

dpres_plevel_Wrap

Calculates the pressure layer thicknesses of a constant pressure level coordinate system.

dz_height

Calculates the height layer thicknesses at each grid point over varying surface terrain.

kf_filter

Extract equatorial waves by filtering in the Wheeler-Kiladis wavenumber-frequency domain.

mixed_layer_depth

Calculates a quantity derived from the Potential Density of significance. (Documentation under construction.)

mjo_cross

Calculate space-time cross spectrum over multiple segments.

mjo_cross_coh2pha

Calculate space-time coherence-squared and phase using the array returned by mjo_cross_segment .

mjo_cross_plot

Plot coherence-squared and phase spectra as returned by mjo_cross.

mjo_cross_segment

Calculate space-time cross spectrum for a single time segment.

mjo_phase_background

Plot background for MJO phase-space diagram.

mjo_space_time_cross

Calculate space-time cross spectrum over multiple segments.

mjo_spectra

Driver to calculate and plot seasonal spectra via segment averaging as defined by the US-CLIVAR MJO diagnostics website .

mjo_spectra_season

Calculates seasonal spectra via segment averaging as defined by the US-CLIVAR MJO diagnostics website .

mjo_wavenum_freq_season

Calculates wavenumber-frequency spectar via seasonal averaging as defined by the US-CLIVAR MJO diagnostics website .

mjo_wavenum_freq_season_plot

Plot wavenumber-frequency spectra as returned by mjo_wavenum_freq_season.

mjo_xcor_lag_ovly

Plot lagged cross-correlations as returned by mjo_xcor_lag_season.

mjo_xcor_lag_ovly_panel

Plot lagged cross-correlations as returned by mjo_xcor_lag_season as panels.

mjo_xcor_lag_season

Calculates lagged correlations between a reference series and gridded data as specified by the US-CLIVAR MJO diagnostics website .

moc_globe_atl

Facilitates calculating the meridional overturning circulation for the globe and Atlantic.

omega_ccm

Calculates omega [vertical pressure velocity] using the model diagnostic method.

omega_ccm_driver

Computes vertical pressure velocity [omega] via model diagnostic code.

pop_remap

Regrids a POP ocean model grid to another grid.

PopLatLon

Regrids a scalar variable on a POP grid to a lat/lon grid or vice-versa.

PopLatLonV

Converts vectors on a POP grid to a lat/lon grid and vice-versa.

potmp_insitu_ocn

Calculate seawater potential temperature at an arbitrary reference pressure given insitu temperature, salinity and depth.

pres_hybrid_ccm

Calculates pressure at the hybrid levels.

resolveWavesHayashi

Reorder the complex coefficients returned by cfftf to resolve the progressive and retrogressive waves.

vinth2p

Interpolates CAM (Community Atmosphere Model) hybrid coordinates to pressure coordinates.

vinth2p_ecmwf

Interpolates CESM hybrid coordinates to pressure coordinates but uses an ECMWF formulation to extrapolate values below ground.

vinth2p_ecmwf_nodes

Interpolates CESM hybrid coordinates to pressure coordinates but uses an ECMWF formulation to extrapolate values below ground.

vinth2p_nodes

Interpolates CESM hybrid coordinates to pressure coordinates on an unstructured grid.

vintp2p_ecmwf

Interpolates data at multidimensional pressure levels to constant pressure coordinates and uses an ECMWF formulation to extrapolate values below ground.

wgt_vert_avg_beta

Computes weighted vertical average or sum using pressure thickness and beta factors.

wgt_volave_ccm

Calculates the volume average of a quantity from the CCM using weights.

wgt_volrmse_ccm

Calculates a weighted volume root-mean-square-difference between two variables from the CCM.

wkSpaceTime

Calculates Wheeler-Kiladis space-time spectra.

wkSpaceTime_cam

Calculates Wheeler-Kiladis space-time spectra using a generic CAM interface.

ESMF

curvilinear_to_SCRIP

Writes the description of a curvilinear grid to a SCRIP file.

ESMF_regrid

Regrids data from one lat/lon grid to another, using ESMF software.

ESMF_regrid_gen_weights

Writes a weight file using the offline ESMF weight generator.

ESMF_regrid_with_weights

Using the provided weight file, regrids data from one lat/lon grid to another.

latlon_to_SCRIP
> Writes the description of a lat/lon grid to a SCRIP file.

rectilinear_to_SCRIP
> Writes the description of a rectilinear grid to a SCRIP file.

unstructured_to_ESMF
> Writes the description of an unstructured grid to an ESMF file.

WRF

wrf_avo
> Calculates absolute vorticity from WRF model output.

wrf_cape_2d
> Computes maximum convective available potential energy (CAPE), maximum convective inhibition (CIN), lifted condensation level (LCL), and level of free convection (LFC).

wrf_cape_3d
> Computes convective available potential energy (CAPE) and convective inhibition (CIN).

wrf_contour
> Creates a contour plot from ARW WRF model output.

wrf_dbz
> Calculates simulated equivalent radar reflectivity factor [dBZ] from WRF model output.

wrf_eth
> Calculates equivalent potential temperature from WRF model output.

wrf_helicity
> Calculates storm relative helicity from WRF model output.

wrf_ij_to_ll
> Finds the nearest longitude, latitude locations to the specified model grid indices (i,j).

wrf_interp_1d
> Linearly interpolates a one-dimensional variable in the vertical.

wrf_interp_2d_xy
> Extracts a cross section from a given input field.

wrf_interp_3d_z
> Interpolates to a specified pressure/height level.

wrf_ll_to_ij
> Finds the nearest model grid indices (i,j) to the specified location(s) in longitude and latitude.

wrf_map
> Creates a map background for ARW WRF model data.

wrf_map_overlays
> Overlays different contour and vector plots over a WRF-ARW map background.

wrf_map_resources
> Sets map plotting resources based on an input WRF-ARW file.

wrf_omega
> Calculates approximate omega in C, given vertical velocity, water vapor mixing ratio, temperature, and pressure from WRF model output.

wrf_overlays
> Overlays multiple plots, created from other ARW WRF plot functions.

wrf_pvo
> Calculates potential vorticity from WRF model output.

wrf_rh
> Calculates relative humidity from ARW WRF model output.

wrf_slp
> Calculates sea level pressure from ARW WRF model output.

wrf_smooth_2d
> Smooths a given field.

wrf_td
> Calculates dewpoint temperature in [C] from ARW WRF model output.

wrf_times_c
> Converts WRF variable "Times" which is of type character to user specified numeric units.

wrf_tk
> Calculates temperature in [K] from ARW WRF model output.

wrf_updraft_helicity
> Calculates updraft helicity from WRF model output.

wrf_user_getvar
> Extracts data from ARW WRF model output, and does basic diagnostics calculations.

wrf_user_ij_to_ll
> Finds the nearest longitude, latitude locations to the specified model grid indices (i,j).

wrf_user_intrp2d
> Interpolates ARW WRF 2D model data along a give line.

wrf_user_intrp3d
> Interpolates ARW WRF model data vertically or horizontally.

wrf_user_list_times
> Extracts the list of available times in the ARW WRF model output.

wrf_user_ll_to_ij
> Finds the nearest model grid indices (i,j) to the specified location(s) in longitude and latitude.

wrf_user_unstagger
> Unstaggers an input variable along a specified dimension.

wrf_uvmet
> Rotates u,v components of the wind to earth coordinates.

wrf_vector
> Creates a vector plot from ARW WRF model output.

wrf_virtual_temp
> Calculates virtual temperature, given temperature and mixing ratio from WRF model output.

wrf_wetbulb
> Calculates wet bulb temperature in C, given pressure in temperature in K and mixing ratio in kg/kg from WRF model output.

Unclassified routines

dim_spei_n
> Calculate the standardized precipitation evapotransporation index (SPEI).

dim_spi_n
> Calculate the standardized precipitation index (SPI).

dim_thornthwaite_n
> Estimate the potential evapotranspiration (PET) via the Thornthwaite method.

dim_ttwpet_n
> Estimate the potential evapotranspiration (PET) via the Thornthwaite method.

rtest
> Determines the statistical significance of a linear correlation coefficient.