1 Program 3dmaskave

1.1 Purpose

Computes average of all voxels in the input dataset which satisfy the criterion in the options list. If no options are given, then all voxels are included.

1.2 Usage

3dmaskave [options] dataset

1.3 Options

-mask mset Means to use the dataset 'mset' as a mask. Only voxels with nonzero values in 'mset' will be averaged from 'dataset'. Note that the mask dataset and the input dataset must have the same number of voxels.

-mindex miv Means to use sub-brick #'miv' from the mask dataset. If not given, then miv=0.

-mrange a b Means to further restrict the voxels from 'mset' so that only those mask values between 'a' and 'b' (inclusive) will be used. If this option is not given, all nonzero values from 'mset' are used. Note that if a voxel is zero in 'mset', then it won't be included, even if $a < 0 < b$.

-dindex div Means to use sub-brick #'div' from the dataset. If not given, all sub-bricks will be processed.

-drange a b Means to only include voxels from the dataset whose values fall in the range 'a' to 'b' (inclusive). Otherwise, all voxel values are included.

-sigma Means to compute the standard deviation as well as the mean.

-dump Means to print out all the voxel values that go into the average. This option cannot be used unless the -mask option is also used.

-udump Means to print out all the voxel values that go into the average, UNSCALED by any internal factors. Also requires -mask. N.B.: the scale factors for a sub-brick can be found using program 3dinfo.

-indump Means to print out the voxel indexes (i,j,k) for each dumped voxel. Has no effect if -dump or -udump is not also used. N.B.: if nx,ny,nz are the number of voxels in each direction, then the array offset in the brick corresponding to (i,j,k) is i+j*nx+k*nx*ny.

The output is printed to stdout (the terminal), and can be saved to a file using the usual redirection operation '>'.