

## Miscellaneous AFNI Utilities

- ❖ Reminder: There are almost 300 AFNI programs, plugins, and scripts, most of which come with help menus that provide a reminder about their usage; for most programs, the output of **-help** is the most up-to-date documentation.

---

- **3dinfo** -- Print out information from a dataset's .HEAD file

- ★ Example from command line: **3dinfo airstrip+orig**

- ★ Example from AFNI GUI: **Define Datamode → Misc → Anat Info**

```
Dataset File:      airstrip+orig
Identifier Code:   XYZ_8qmBAapL9YwE3I Creation Date: Wed Jun 9 11:54:12 2004
Dataset Type:     Spoiled GRASS (-spgr)
Byte Order:       MSB_FIRST [this CPU native = LSB_FIRST]
Data Axes Orientation:
  first (x) = Anterior-to-Posterior
  second (y) = Superior-to-Inferior
  third (z) = Left-to-Right [-orient ASL]
R-to-L extent:   -74.000 [R] -to- 73.600 [L] -step- 1.200 mm[124 voxels]
A-to-P extent:  -119.531 [A] -to- 119.531 [P] -step- 0.938 mm[256 voxels]
I-to-S extent:  -119.531 [I] -to- 119.531 [S] -step- 0.938 mm[256 voxels]
R-to-L center:   -0.200 [R]
A-to-P center:   0.000 [P]
I-to-S center:   -0.000 [I]
Number of values stored at each pixel = 1
  -- At sub-brick #0 '#0' datum type is short:          0 to          733

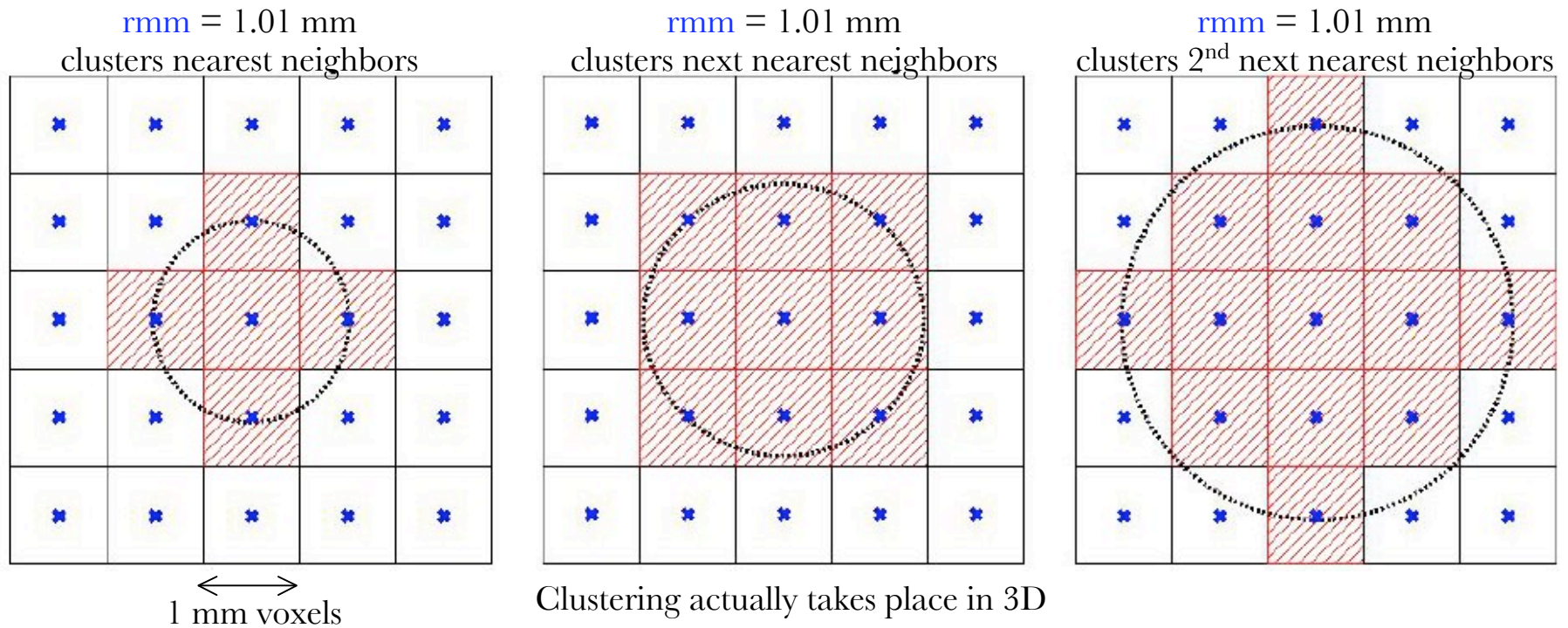
---HISTORY---
[cox@elrond: Mon Jun 14 16:04:31 2004] 3dIntracranial -min_val 30
-anat fred+orig -prefix airstrip
```

- **3dnewid** -- Change the ID code in a dataset's .HEAD file
  - ★ ID codes used internally to identify datasets (e.g., in parent-child relationships)
  - ★ Type **3dnewid -fun** to see a sample ID code (supposed to be globally unique)
- **3dcopy** -- Make a copy of a dataset (.HEAD and .BRIK files)
  - ★ Is equivalent to using Unix **cp** command on the .HEAD and .BRIK files, then using **3dnewid** on the copy
    - ↳ E.g., **3dcopy fred+orig ethel+orig**
      - ◇ Result: makes a duplicate of fred+orig and renames the new file ethel+orig
- **3drename** -- Rename a dataset (.HEAD and .BRIK files)
  - ★ Is equivalent to using Unix **mv** command on the .HEAD and .BRIK files
    - ↳ E.g., **3drename fred+orig ethel+orig**
      - ◇ Changes the dataset's prefix name from **fred** to **ethel**
- **3dNotes** -- Attach notes to a dataset's .HEAD file that will be printed by **3dinfo**
  - ↳ E.g., **3dNotes -a "Subject smells like old socks" fred+orig**
  - ★ This note will now be added to **fred+orig**
  - ★ **Dataset NOTES** plugin provides an interactive way to do the same thing

- **3dbucket** - Assemble various sub-bricks into a single “bucket” dataset
  - ★ Lets you put diverse results into one place for easy viewing
    - ↳ E.g., **3dbucket fred+orig ethel+orig -output big\_dset**
      - ◇ Result is **big\_dset+orig**, which contains fred+orig (as sub-brick #0) and ethel+orig (as sub-brick #1)
  - ★ Can also be used to slim down a big bucket dataset
    - ↳ E.g., **3dbucket 'func[5,17-21]'+orig -output func\_slim+orig**
      - ◇ The result is dataset **func\_slim+orig**, which contains only sub-bricks 5, 9, and 17 through 21
- **3ddup** - Make a warp-on-demand (.HEAD file *only*) copy of a dataset
  - ★ Allows AFNI to resample a dataset to a new grid without destroying an existing data .BRIK
    - ↳ E.g., **3ddup fred+orig ethel+orig**
  - ★ Then use **Define Datamode → Resam** and **→ Write** to write new dataset to disk
- **3drefit** - Modify parameters in a dataset’s .HEAD file (does not change the .BRIK file at all)
  - ★ Lets you “patch” mistakes made in **to3d** (not by *you*, by someone else, right?)
    - ↳ E.g., **3drefit -orient LPI -zorigin 30 fred+orig**
      - ◇ The result is **fred+orig** dataset with xyz-orientation of LPI and center of the first slice at 30 mm Inferior

# Spatial Utilities

- **3dclust** -- Find clusters of “active” voxels and print out a report about them
  - ★ “Active” means nonzero (survives thresholding operation)
  - ★ Clusters are defined by a connectivity radius parameter **rmm**:



- ★ Clustering starts by finding some nonzero voxel
- ★ All nonzero voxels closer than **rmm** millimeters (center-to-center distance) to the given voxel are included in the cluster
- ★ Cluster then grows outwards from all newly included voxels, using rmm again

- ★ Clustering actually takes place in 3D:
  - ↳ Assume cubical voxels with grid size L mm
  - ↳  $L < rmm < \sqrt{2} L \Rightarrow$  connect voxels that share a common face
  - ↳  $\sqrt{2} L < rmm < \sqrt{3} L \Rightarrow$  connect voxels that share a common edge
  - ↳  $\sqrt{3} L < rmm < 2L \Rightarrow$  connect voxels that share a corner
  - ↳ Larger values of **rmm** will jump over zero voxels
- ★ You can override actual voxel size (which may not be cubical) by using the **-dxyz=1** command line switch, which then pretends that voxel size L=1
- ★ Sample report: **3dclust -lthresh 0.47 7 600 fred\_epi+orig**

```
Cluster report for file fred_epi+orig
[Connectivity radius = 7.00 mm Volume threshold = 600.00 ]
[Single voxel volume = 98.4 (microliters) ]
[Voxel datum type   = short ]
[Voxel dimensions   = 3.750 mm x 3.750 mm x 7.000 mm ]
Mean and SEM based on Absolute Value of voxel intensities:
```

Volume	CM RL	CM AP	CM IS	minRL	maxRL	minAP	maxAP	minIS	maxIS	Mean	SEM	Max Int	MI RL	MI AP	MI IS
3839	2.3	-15.3	4.4	-11.0	10.0	-28.1	-5.6	-9.4	20.6	0.0069	4.8e-04	0.0176	3.0	-13.1	5.6
2067	16.0	56.8	9.4	3.0	24.0	39.4	65.6	1.9	16.9	0.0059	4.3e-04	0.0107	17.0	61.9	13.1
1772	38.4	-5.3	0.2	24.0	52.0	-13.1	-1.9	-9.4	5.6	0.006	5.1e-04	0.0111	31.0	-1.9	-1.9
1575	-18.4	-36.7	4.5	-25.0	-18.0	-43.1	-28.1	-9.4	13.1	0.0072	0.001	-0.0181	-18.0	-43.1	5.6
1477	-1.4	-65.8	-31.7	-4.0	10.0	-69.4	-58.1	-39.4	-28.1	0.0109	0.001	-0.0201	-4.0	-65.6	-31.9
1280	24.5	-30.0	1.4	24.0	31.0	-35.6	-24.4	-9.4	9.4	0.0053	4.9e-04	-0.0089	24.0	-35.6	5.6
1181	0.7	-50.9	0.6	-4.0	10.0	-54.4	-46.9	-13.1	5.6	0.0071	0.0011	-0.0154	-4.0	-50.6	5.6
886	42.9	-28.8	-10.9	38.0	52.0	-31.9	-20.6	-13.1	-5.6	0.0059	7.7e-04	0.0096	38.0	-31.9	-9.4
689	33.2	-4.7	17.8	31.0	38.0	-9.4	-1.9	13.1	24.4	0.0074	0.0011	0.0133	31.0	-1.9	20.6
14766	9.8	-20.1	-1.8							0.0069	2.7e-04				

- ★ **-lthresh 0.47**=threshold to apply to dataset; **7 = rmm**; **600** = volume of smallest cluster to report (in mm<sup>3</sup> = microliters)

- **3dmerge** -- Spatially edit and/or combine datasets (the very first “3d”program!)
  - ★ “Editing” options (applied to each input dataset) include:
    - ↳ Clipping (setting small values to zero)
    - ↳ Thresholding (setting voxels to zero based on some other sub-brick)
    - ↳ Spatial filtering (e.g., Gaussian blur)
    - ↳ Spatial clustering
  - ★ “Combining” options include (all are voxel-wise across multiple input datasets, after the editing operations are carried out):
    - ↳ Mean; Mean of nonzero inputs
    - ↳ Maximum; Maximum absolute value
    - ↳ Count of nonzero input voxels
- **3dZeropad** -- add planes of zeros around a dataset
  - ★ Can also cut planes off edges of dataset
    - ↳ E.g., **3dZeropad -RL 4 -AP 2 -IS 3 -prefix fred\_pad fred+orig**
- **3dZcutup** and **3dZcat** -- cut slices out of dataset; glue datasets together
  - ★ In the slice (z) direction
  - ★ Used when processing a big 3D+time dataset is too much
    - ↳ E.g., **3dZcutup -prefix fred\_zcut -keep 2 15 fred\_epi+orig**
      - ◇ Result: **fred\_zcut+orig**, containing only slices 2 through 15 of fred\_epi+orig

- [3dZregrid](#) -- Resample datasets in the slice (z) direction
  - ★ Used when you want to compare datasets acquired with different slice thickness
  - ★ But you should really be more careful when acquired your datasets!
- [3dcalc](#) -- Voxel-by-voxel general purpose calculator
  - ★ Useful for combining ROI masks in various ways
  - ★ Useful for forming 'conjunction tests', and many other voxel-wise operations

examples:

```
3dcalc -prefix mask_17.2 -a stats+orig'[2]' -expr 'ispositive(a-17.2)'  
3dcalc -prefix stat_mask -a stats+orig'[2]' -b mask+orig -expr 'a*ispositive(b)'  
3dcalc -prefix stats_9 -a stats+orig'[2]' -b mask+orig -expr 'a>equals(b,9)'  
  
3dcalc -prefix data_9_to_4 -a stats+orig -expr '9*iszero(a-4)+a*notzero(a-4)'
```

- [3dMean](#) -- Voxel-by-voxel mean of input datasets
  - ★ Like [3dmerge -gmean](#), but simpler to used, and much faster than [3dcalc](#)
    - ↳ E.g., [3dMean -prefix mean\\_dset ziad+orig bob+orig rick+orig](#)
- [3dClipLevel](#) -- Estimate the voxel value at which to threshold an EPI dataset so as to remove most non-brain tissue (essentially creates a mask)
  - ★ Same algorithm is used as starting point in [3dAutomask](#)

- **3dIntracranial** -- Strip the scalp and other non-brain tissue from a high-resolution T1-weighted dataset
  - ↳ E.g., `3d Intracranial -anat fred+orig -prefix fred_strip`
  - ★ Useful for volume rendering fun
  - ★ Combined with `3dfractionize`, is another way to make a brain-only mask
- **new!** **3dSkullStrip** -- Perform spatial normalization on MRI T1-weighted volume, and then extract the brain from surrounding tissue
  - ↳ E.g., `3dSkullStrip -input fred_anat+orig -prefix fred_strip \  
-no_avoid_eyes -niter 750 -ld 50`
- **3daxialize** -- Rewrite a dataset in a new slice orientation
  - ↳ E.g., `3daxialize -orient LPI -prefix fred_LPI fred_RAS+orig`
- **3dresample** -- Rewrite a dataset in a new orientation and interpolates to a new voxel size
  - ↳ E.g., `3dresample -dxyz 1.0 1.0 0.9 -prefix fred_1.1.09 \  
-inset fred+orig`
- **3drotate** -- Does rigid body 3D rotation of a dataset
  - ↳ E.g., `3drotate -prefix fred_rotate -bshift 10S 0 0 \  
-rotate 30R 0 0 fred+orig`
    - ◇ Result is a new dataset that has been shifted by 10mm in the Superior direction, followed by a 30 degree rotation about the Right-to-Left axis.



- **3dWarp** -- Does nonrigid 3D transformation of a dataset
  - ↳ E.g., **3dWarp -tta2mni -prefix fred\_mni fred+tlrc**
    - ◇ Example shows Talairach-Tourneaux dataset that has been transformed to MNI-152 coordinates (but still keeps 'tlrc' suffix, e.g., fred\_mni+tlrc).
- **3dAnatNudge** -- Automatically tries to align EPI and structural datasets.
  - ★ Note: The anat dataset is moved around to align properly with the EPI dataset.
  - ★ Skull from anat dataset should first be stripped before running 3dAnatNudge.
    - ↳ E.g., **3dAnatNudge -anat spgr+orig -epi run1+orig \**  
**-prefix spgr\_nudge**
  - ★ **Dataset Nudge** plugin lets you do the same manually
- **3dTagalign** -- Align 2 datasets based on manually set tags
  - ↳ E.g., **3dTagalign -master fred+orig -rotate \**  
**-prefix ethel\_tag ethel+orig**
    - ◇ Result: ethel\_tag+orig should be aligned with master dataset fred+orig
  - ★ Tags are set with the **Edit Tagset** plugin
  - ★ Alignment can be rigid body (6 parameters) or affine (12 parameters)
  - ★ This can be very useful when **3dAnatNudge** doesn't work well

## Masking and ROI Utilities

- **3dAutomask** -- Create a brain-only mask from an EPI dataset
  - ↳ E.g., `3dAutomask -prefix fred_mask -dilate 2 fred+orig`
    - ◇ Result: A mask of fred+orig, which has been dilated outwards 2 times.
- **3dfractionize** -- Resample a high-resolution dataset to lower resolution
  - ★ i.e., takes high-resolution mask (ROI) datasets to lower EPI resolution
  - ★ Can compute fraction of each output voxel that is occupied by nonzero input voxels (default operation)
  - ★ Can let input voxels “vote” on the value of output voxels (since multiple input voxels can overlay a larger output voxel, must have a scheme to decide which input value “wins”)
    - ↳ E.g., `3dfractionize -template func+orig -clip 0.5 -preserve\  
-input ROI_highres+orig -prefix ROI_lores`
- **3dmaskdump** -- Print out all the voxel values indicated by a mask dataset
  - ↳ E.g., `3dmaskdump -noijk -mask ROI_fred+tlrc -mrange 1 3 \  
'func_fred+tlrc[5]'`
- **3dUndump** -- Take a text file and puts its values into a dataset
  - ★ The inverse of **3dmaskdump**

- **3dmaskave** -- Print out the average of voxels over an ROI
  - ↳ E.g., `3dmaskave -mask ROI+tlrc -mrange 1 1 -q 'func_fred+tlrc[5]'`
  - ★ Can compute a 1D time series averaged from a 3D+time dataset over an ROI
  - ★ Can also use **ROI Average** plugin.
- **3dROIstats** -- Print out statistics of voxels from multiple ROIs
  - ↳ E.g., `3dROIstats -mask ROI+tlrc func_fred+tlrc`
- **3dOverlap** -- Count the number of voxels that are nonzero in all input sub-bricks

## New! Diffusion Tensor Imaging (DTI) Programs

- **3dDWItoDT** -- For diffusion weighted image (DWI) data, calculate the diffusion tensor image (DTI) data.
  - ★ Uses traditional linear or an iterative non-linear method to compute diffusion tensor. Computes eigen values, eigen vectors, fractional anisotropy, mean diffusivity
- **3dTeig** -- From DTI data compute eigen values, eigen vectors and fractional anisotropy.
- **3dDTtoDWI** -- Compute diffusion weighted volumes based on the diffusion tensor and an ideal B0 volume with no gradient.
  - ★ Useful for testing purposes only.
- **DTIStudioFibertoSegments** -- Takes output of fiber tracking from DTIStudio.
  - ★ Popular DTI program from Johns Hopkins. Output can be displayed in SUMA.
- **3dAFNItoRaw** -- Converts a multi-sub-brick AFNI dset into a “raw” file format with each corresponding sub-brick voxel value placed into consecutive n-tuplets of data.
  - ★ Developed to convert AFNI eigen vector data for use in DTIStudio

## 3D+time Data Utilities

- [3dTsmooth](#) -- Smooth a 3D+time dataset along the time axis
- [3dFourier](#) -- Filter time series in the Fourier domain
  - ★ Allows more general filtering than 3dTsmooth
- [3dTcat](#) -- Catenate 3D+time datasets together
  - ★ For use in [3dDeconvolve](#), for example
- [3dTstat](#) -- Perform basic statistics on 3D+time datasets
  - ★ Voxel-wise means, standard deviation, median, etc. (mean is default)
    - ↳ E.g., `3dTstat -prefix mean_run1 fred_run1+orig`
- [3dTqual](#) and [3dToutcount](#) -- Check 3D+time datasets for 'outliers'
  - ★ Now also included automatically in to3d
  - ★ [3dDespike](#) --Remove outliers (spikes) from voxel time series
- [3dDetrend](#) -- Subtract least squares fits of 'trends' from voxel time series
  - ★ Usually better to do this at the same time as activation analysis
- [3dTshift](#) -- Align dataset slices to the same time origin
  - ↳ E.g. `3dTshift -prefix fred_run1_ts fred_run1+orig`
  - ★ Can also be done in [3dvolreg](#), before registration (using the `-tshift` option)

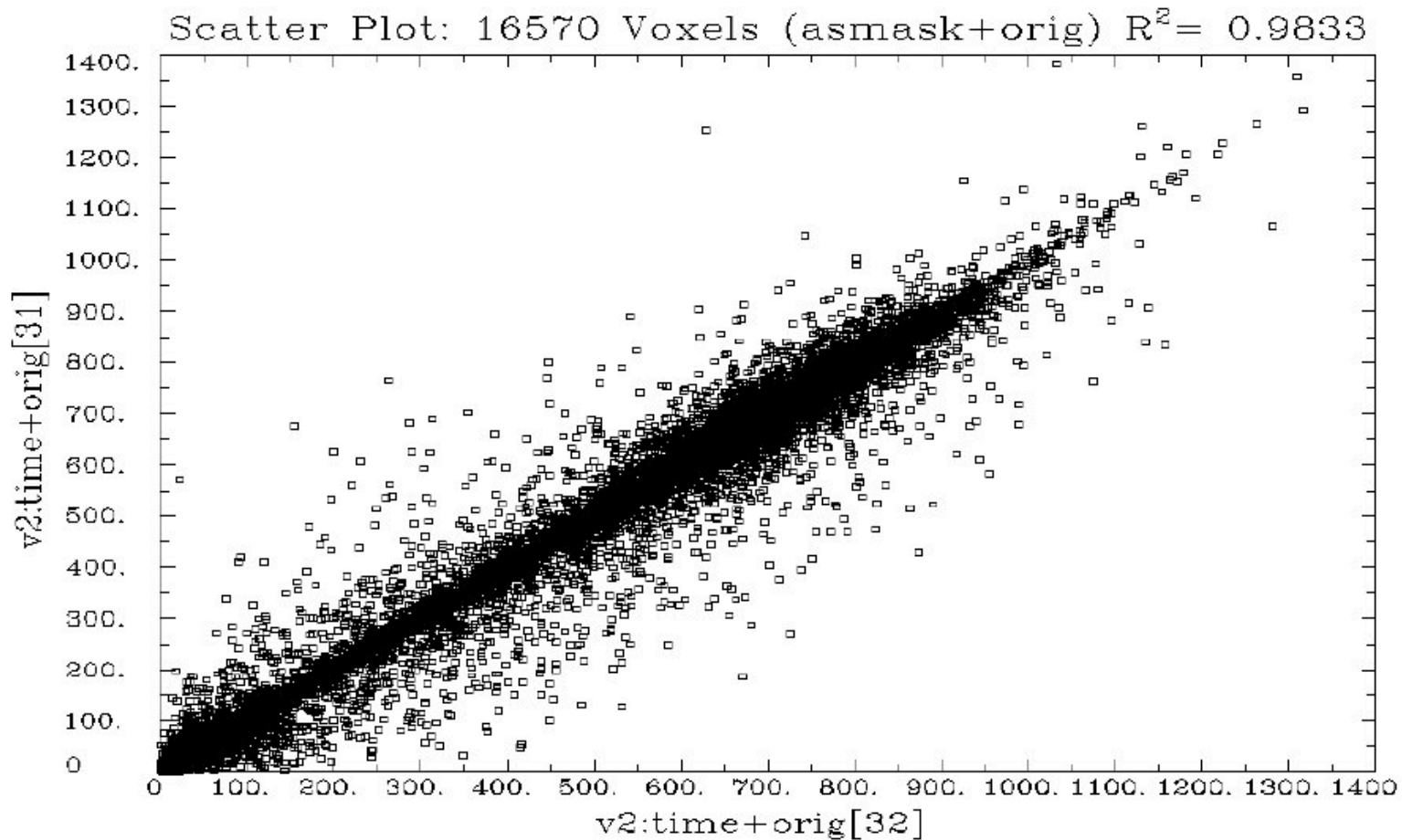
# Miscellaneous Useful Plugins

- Define Datamode → Plugins →

```
-- Cancel -- Dataset#2 maskcalc
2D Registration Dataset#N Maxima
3D Cluster Deconvolution NLfit & NLerr
3D Correlation Draw Dataset Nudge Dataset
3D Dump98 Dset Zeropad Permutation Test
3D Edit Edit Tagset Power Spectrum
3D Registration Expr 0D Render [new]
3D+t Extract Fourier Stuff Render Dataset
3D+t Statistic Gyrus Finder Reorder
4D Dump Hemi-subtract ROI Average
BRIC Compressor Hilbert Delay98 ROI Plot
Coord Order Histogram ScatterPlot
Dataset Copy Histogram: BFit SingleTrial Avg
Dataset Dup Histogram: CC Threshold
Dataset NOTES L1_Fit & Dtr TS Generate
Dataset Rename LSqFit & Dtr Wavelets
```

- Coord Order -- Set the order in which coordinates are displayed
  - ★ AFNI default order is RAI:
    - x = Right                    +x = Left
    - y = Anterior                +y = Posterior
    - z = Inferior                +z = Superior
  - ★ Can choose another order with this plugin: most common is LPI or “flipped”
- Histogram -- Graph a histogram of a sub-brick
  - ★ Or of an ROI , or a spherical region about the AFNI focus point

- **Dataset#N** -- Allows you to graph extra dataset time series in an AFNI graph viewer (overlaid in color on the current underlay dataset time series)
  - ★ In conjunction with the **Double Plot** graphing function
- **ScatterPlot** -- Plot values from 1 sub-brick vs. values from another:



## 1D Time Series

- [waver](#) -- Generate 1D time series which are convolution of input stimulus timing with model hemodynamic response functions
- [1dplot](#) -- Graph 1D time series files
  - ★ Not very fancy graphs: mostly useful for quick look at data, not for publications
- [1dcat](#) -- Catenate columns of 1D time series files
- [1deval](#) -- Similar to [3dcalc](#), but for 1D time series
- [1dtranspose](#) -- Transpose a 1D file (i.e., exchange rows and columns)
  - ★ If you want to input a 1D file as an AFNI dataset, then the columns correspond to sub-bricks

## Data Simulators

- [3dTSgen](#) -- Generate a dataset from a time series model and noise
- [3dConvolve](#) -- Generate a dataset by convolution (opposite of 3dDeconvolve)
- [3dcalc](#) -- Can be used to generate datasets with noise added (if you are clever)



## Environment Variables and .afnirc

- Operation of AFNI is affected by many Unix environment variables
  - ★ Full documentation is in file README.environment (in AFNI distributions)
  - ★ Environment variables can be set in your shell startup file (e.g., .cshrc) or in AFNI's startup file (.afnirc), in your home directory
  - ★ Some environment variables can be set from the pseudo-plugin Define Datamode → Misc → Edit Environment
- *Some* useful environment variables (there are many more)
  - ★ **AFNI\_PLUGINPATH** gives the directory where AFNI will look for plugins when it starts up
  - ★ **AFNI\_SESSTRAIL** gives the number of directory levels to show in the Switch Session chooser
  - ★ **AFNI\_HINTS** can be used to turn off the popup hints (tooltips)
  - ★ **AFNI\_COMPRESSOR** can be used to tell AFNI programs to compress .BRIK files when they are written out
  - ★ **AFNI\_AUTOGZIP** can be used to tell AFNI programs to gzip compress .BRIK files if they appear like “good” candidates for compression (e.g., ROI datasets)

- ★ **AFNI\_LEFT\_IS\_LEFT** can be used to have axial and coronal images displayed with the subject's left on the display left (default is subject's left on the display right: radiological order)
- ★ **AFNI\_ALWAYS\_LOCK** can be used to turn on inter-controller Lock at startup
- ★ **AFNI\_NOSPLASH** can be used to hide the AFNI splash window (but why?!)
- ★ **AFNI\_ENFORCE\_ASPECT** can be used to make defective window managers (KDE, Gnome) keep the image window aspect ratios when resizing (I then also recommend setting the window manager so that it doesn't redraw the windows during resizing operations)

- Sample .afnirc file:

```
***ENVIRONMENT
AFNI_LEFT_IS_LEFT      = YES      // images show subject's left on screen left
AFNI_graph_width      = 512      // in pixels
AFNI_graph_height     = 384
AFNI_graph_ggap       = 6        // gap between sub-graphs
AFNI_graph_data_thick = 1        // use thick lines for data graphs
AFNI_SPLASHTIME       = 1.0     // shorten the splash screen display
AFNI_ALWAYS_LOCK      = YES      // locking windows together
AFNI_ENFORCE_ASPECT   = YES
AFNI_AUTOGZIP         = YES      // 02 Mar 2001
```

- See [README.environment](#) and [README.setup](#) for details on all environment variables and other setup issues

## AFNI Command Line Switches

- Command line switches to the interactive AFNI program itself are not often needed, but are sometimes useful:
  - ★ **-R** = recursively search directories for datasets (may take a long time)
  - ★ **-noqual** = skip the “quality” check for marker transformations
  - ★ **-noplugins** = don’t load AFNI plugins
  - ★ **-skip\_afnirc** = don’t read the .afnirc file
  - ★ **-ncolors nnn** = use ‘nnn’ gray levels for image display (default = 80)
  - ★ **-nomall** = disable use of AFNI’s internal malloc() (if it causes problems)

## Final AFNI Fun

- Try clicking each mouse button in the blank area to the right of the [Done](#) button
- Try clicking or typing into the splash screen
- Try using [Bhelp](#) in a blank area of the AFNI controller
- To exit AFNI quickly, press the Shift key down, then click on [Done](#)

## Roundup of Useful AFNI Programs and Plugins

- Dataset Creation and Conversion

<b>to3d</b>	Read image files, write AFNI format datasets
<b>3dAFNIto3D</b>	Convert AFNI format dataset to .3D format (ASCII lists)
<b>3dAFNItoANALYZE</b>	Convert AFNI format dataset to ANALYZE format
<b>3dAFNItoMINC</b>	Convert AFNI format dataset to MINC format
<b>3dANALYZEtoAFNI</b>	Convert ANALYZE format dataset to AFNI format
<b>3dMINCtoAFNI</b>	Convert MINC format dataset to AFNI format
<b>3dThreetoRGB</b>	Convert 3 scalar datasets to 1 RGB AFNI format dataset

- Auxiliary Programs for Dataset Creation from Images

<b>Ifile</b>	Read GE realtime EPI files and runs to3d
<b>Imon</b>	Read GE realtime EPI files as they are created
<b>Dimon</b>	Read DICOM files as they are created
<b>rtfeedme</b>	Dissect one dataset, sends images to AFNi realtime plugin
<b>plugin: RT Options</b>	Control options for AFNI realtime image input
<b>from3d</b>	Write dataset slices into image files
<b>abut</b>	Create zero-filled slices to put into dataset gaps

- Quality Checks for 3D+time Datasets

<b>3dToutcount</b>	Check voxel time series for quality (temporal outliers)
<b>3dTqual</b>	Check dataset sub-bricks for quality (spatial outliers)

- 3D+time Pre-Processing Programs

- 3DTshift** Shift slices to a common time origin (temporal interpolation)
- 3dDespike** Remove spikes from voxel time series
- 3dDetrend** Remove trends from voxel time series
- 3DFourier** FFT-based lowpass and highpass filtering
- 3dTsmooth** Smooth time series in the time domain

- 3D+time Analysis Programs

- 3dDeconvolve** Multiple linear regression and deconvolution
- plugin: Deconvolution** Interactive deconvolution
- 3dfim** Single regression linear analysis
- 3dfim+** Extended version of 3dfim
- 3ddelay** Single regressor linear analysis with time shifting
- 3dNLfim** Nonlinear regression
- plugin: Nlfit & Nlerr** Interactive nonlinear regression
- 3dTcorrelate** Correlate two input datasets, voxel-by-voxel
- 3dAutoTcorrelate** Correlate each voxel with every other voxel
- 3dpc** Principal component analysis

- Model 1D Time Series Generators

- sqwave** Generate a square wave (a very old program)
- waver** Generate hemodynamic responses to stimulus time series

- Dataset Histogram Programs

<b>3dAnhist</b>	Create and plot histogram of dataset, print peaks
<b>3dhistog</b>	Create histogram of dataset to a file
plugin: <b>Histogram</b>	Interactively graphs histogram of a dataset (or ROI)
plugin: <b>ScatterPlot</b>	Interactively graphs 1 sub-brick vs. another (or ROI)
<b>3dClipLevel</b>	Find value to threshold off outside-the-brain voxels
<b>3dUniformize</b>	Correct T1-weighted dataset for non-uniform histogram
<b>3dIntracranial</b>	Strip off outside-the-brain voxels

- Group Dataset Statistical Analysis Programs

<b>3dtttest</b>	Paired and unpaired t-tests
<b>3dANOVA</b>	1-way ANOVA (fixed effects)
<b>3dANOVA2</b>	2-way ANOVA (fixed, random, mixed effects)
<b>3dANOVA3</b>	3-way ANOVA (fixed, random, mixed effects)
<b>3dANOVA4</b>	4-way ANOVA (currently a MatLab script, March 2005)
<b>3dANOVA5</b>	5-way ANOVA (in the works, March 2005)
<b>3dFriedman</b>	Nonparametric Friedman test
<b>3dKruskalWallis</b>	Nonparametric Kruskal-Wallis test
<b>3dWilcoxon</b>	Nonparametric Wilcoxon test
<b>3dMannWhitney</b>	Nonparametric 3dMannWhitney test
<b>3dRegAna</b>	Voxel-wise linear regression analyses
<b>3dFDR</b>	False Discovery Rate analysis

- Programs for Manipulating Information in the Dataset Header

- 3dinfo** Print out information from the header
- 3dAttribute** Print out a single header attribute
- 3dnewid** Assign a new ID code to a dataset
- 3drefit** Lets you change attributes in a dataset header
- 3dNotes** Lets you put text notes into a dataset header
- plugin: **Dataset NOTES** Interactive header notes editor
- nifti\_tool** Displays, modifies, copies nifti structures in datasets

- Programs for Changing Dataset Spatial Structure

- 3daxialize** Rewrite dataset with slices in different direction
- 3dresample** Rewrite dataset in new orientation, with new voxel size
- 3dLRflip** Flip dataset Left ↔ Right

- Programs for Assembling Sub-bricks into 4D Datasets

- 3dTcat** Assemble a 3D+time dataset from multiple input sub-bricks
- 3dbucket** Assemble a bucket dataset from multiple input sub-bricks

- Programs for Changing Slice Structure

- 3dZcat** Glue multiple sub-bricks together along the z-axis
- 3dZcutup** Cut slices out of a dataset to make a 'thinner' dataset
- 3dZeropad** Add zero slices around the edges of a dataset
- 3dZregrid** Interpolate a dataset to a different slice thickness

- Spatial Transformations of Dataset Geometry

- 3drotate** Rigid body rotation of dataset in 3D
- 3dWarp** Non-rigid transformation of 3D coordinates
- 3dAnatNudge** Try to align EPI and structural volumes automatically
- plugin: **Nudge Dataset** Align EPI and structural volumes manually
- 3dTagalign** Align datasets by matching manually placed 'tags'
- plugin: **Edit Tagset** Place 'tags' in a dataset interactively
- adwarp** Transform dataset using warp from dataset header
- Vecwarp** Transform 3-vectors using warp from dataset header

- Dataset File Manipulation

- 3dcopy** Copy a dataset to make new files
- 3drename** Rename dataset files
- 3ddup** Make an 'empty' duplicate (warp-on-demand) of a dataset

- ROI Generation and Usage Programs

- plugin: **Draw Dataset** Manually draw ROI mask datasets
- 3dAutomask** Generate a brain-only mask from an EPI dataset
- 3dmaskave** Calculate dataset values averaged over a ROI
- 3dmaskdump** Output all dataset values in a ROI
- 3dROIstats** Calculate dataset values from multiple ROIs
- 3dUndump** Input text values into a dataset (inverse of 3dmaskdump)
- 3dOverlap** Create mask that is overlap of nonzero voxels from multiple datasets
- 3dfractionize** Resample a mask dataset to a different resolution



- Simple Calculations on Datasets, Producing New Datasets

- 3dcalc**                      Voxel-by-voxel general purpose calculator
- 3dmerge**                    Various spatial filters, thresholds, and averaging
- 3dTstat**                    Various statistics of multi-brick datasets, voxel-by-voxel
- 3dMean**                    Average datasets together, voxel-by-voxel, for each timept
- 3dWinsor**                  Nonlinear order statistics filter for spatial smoothing

- Computation of Various Numbers from Datasets

- 3ddot**                      Dot product (correlation coefficient) of 2 sub-bricks
- 3dclust**                    Find connected clusters of nonzero voxels
- 3dExtrema**                Find local maxima (or minima) of datasets
- 3dFWHM**                  Estimate Full Width Half Max of dataset spatial correlation
- 3dMax**                    Find range of voxel values (max, min, count) in a dataset

- Simulated Dataset Generators

- 3dTSgen**                    Generate 3D+time dataset from 1D model and noise
- AlphaSim**                  Simulate datasets and estimate statistical power
- 3dConvolve**                Simulate datasets via convolution

- Programs for Dealing with 1D Time Series

- 1dcat**                      Catenate them horizontally
- 1deval**                    1D calculator (like 3dcalc for 1D files)
- 1dplot**                    Graph them
- 1dtranspose**                Transpose them (interchange rows and columns)

- Image Registration Programs

**3dvolreg** Volumetric registration (rigid body in 3D)  
**3dImReg** Slice-by-slice registration (rigid body in 2D)

- Miscellaneous File Manipulations

**2swap** Byte pair swap: ab ba  
**4swap** Byte quad swap: abc dcba  
**24swap** Mixed 2 and 4 byte swaps in same file  
**strblast** Find a string in a file and replace it with junk

- Miscellaneous Utilities

**byteorder** Report the byteorder of the current CPU  
**ccalc** A command line calculator (like 3dcalc)  
**cdf** Compute probabilities, thresholds for standard distributions  
**count** Generate numbered strings for command line scripts

- Image File Header Printouts

**dicom\_hdr** Print information from a DICOM file  
**ge\_header** Print information from a GE I. file  
**mayo\_analyze** Print information from an ANALYZE .hdr file  
**siemens\_vision** Print information from a Siemens Vision .ima file

- Miscellaneous Visualization Tools

**aiv** AFNI Image Viewer program  
plugin: **Render[new]** Interactive volume rendering  
plugin: **Dataset#N** Graph extra dataset time series in AFNI graph viewer