ALLENMINER users guide
v2.01 (2012 Nov 27)

Fred P. Davis
`http://fredpdavis.com`

August 25, 2015

Introduction

ALLENMINER searches gene expression in the adult and developing mouse brain and spinal cord atlases published by the Allen Institute for Brain Science (AIBS). It can identify genes that are (1) expressed, (2) expressed specifically, (3) expressed non-uniformly, (4) expressed in a graded fashion, or (5) expressed in a pattern similar to a query gene, in a user-specified region of interest of the mouse brain – specified either as a set of brain region names or as x,y,z bounds of a cuboid.

The code includes several routines for processing Allen Brain Atlas (ABA) data files, including the 3D expression data and the adult and developing reference atlases. Besides accessing the raw data, these routines are useful for visualizing ABA expression files on platforms where the ABA BrainExplorer application is not available, such as GNU/Linux. For example, all ABA expression files can be converted to PDB (Protein DataBank) format for visualization in protein structure viewers, such as PyMOL (`http://pymol.org`).

I developed ALLENMINER as a postdoc at the HHMI Janelia Research Campus.

What's new in v2.01?

Updated to the lastest adult mouse brain atlas (v3).

What was new in v2.0?

- Easier to install. No longer requires Python, any CPAN packages, or the (now outdated) ABA API package.
- Searches the latest ABA adult (v2.0; Nov 2011 release) and developing mouse brain atlases (v3.0; Nov 2011 release), as well as the adult and developing spinal cord atlases.
- Experimental mode (`make_report`) to format search results together with original 2D ISH images for viewing in a web browser.

Contents

1 Installing ALLENMINER

1.1 Prerequisites

1. Perl (`http://www.perl.org`)

   Usually pre-installed on Mac OS X and GNU/Linux. Other visit (`http://www.perl.org/get.html`).
2. wget (`http://www.gnu.org/software/wget`) - to download ABA data.
3. unzip - to parse expression data.
4. gzip - to compress output files.

1.2 Download the program

Download the program from `http://fredpdavis.com/allenminer` and unpack it in your desired installation directory:

```
cd /usr/local/software
tar xvfz allenminer_v2.01.tar.gz
```

1.3 Place the perl library in your PERL5LIB path

Place the directory containing alnmnr.pm (src/perl_api) in your PERL5LIB environment variable:

For example, if you run a csh or tcsh shell, add this to your .cshrc file:

```
setenv PERL5LIB {$PERL5LIB}:/usr/local/software/allenminer_v2.0/src/
   perl_api
```

For a bash shell, add this to your .bashrc:

```
PERL5LIB=/usr/local/software/allenminer_v2.0/src/perl_api:$PERL5LIB
export PERL5LIB
```

1.4 Specify local configuration details

Edit the alnmnr.pm specs section (line 65) to specify the:

- Directory where ABA files will be downloaded.

  ```
  $specs->{allen_data_dir} =
      '/groups/eddy/home/davisf/work/databases/ABA_data' ;
  ```

- If you want to run parallel searches on an SGE cluster, edit the following lines (67-70):

  ```
  $specs->{cluster}->{OPTION} = VALUE ;
  ```

  1. `cluster_mode` - if 1, uses a cluster by default. if 0, will not.
  2. `head_node` - this is the name of the machine from which jobs can be submitted; make sure you have passwordless ssh setup between your local machine and the cluster head node. If you can submit directly from your local machine, set this to blank: '' .
  3. `numjobs` - maximum number of compute nodes to use.
  4. `qstat_sleep` - how often to qstat the jobs (sec)

1.5 Download ABA atlases

A local repository of adult (∼ 10GB) and developing (∼ 8GB) three-dimensional expression data is useful for custom ROI, gradient/patterning searches, or similarity searches. However, for simple queries of the expression level in a reference atlas region, or relative enrichment between two atlas regions, you can use the `fastsearch` run mode, which doesn't require a full dataset, and runs much faster. It does this by searching through summary files generated by ALLENMINER that describe expression levels for all genes, in all regions, at all developmental timepoints.

For fastsearch runs, you'll need to download these summary files (total ∼ 800 MB) by running this command:

```
perl allenminer.pl -mode download -type fastsearch
```

To get the full datasets for the adult brain, developing brain or spinal cord data, these commands will download the appropriate files from the ABA webserver into the directory specified above (`$specs->{allen_data_dir}`):

```
perl allenminer.pl -mode download -type adultbrain
perl allenminer.pl -mode download -type develbrain
perl allenminer.pl -mode download -type spinalcord
```

2 Running ALLENMINER

ALLENMINER has six run modes for the brain (search, fastsearch, simsearch, roiop, convert, make_report) and one for the spinal cord (spinesearch).

2.1 search: Search for genes expressed/enriched/patterned in an ROI

Processes expression files and returns the absolute levels and enrichment within one or more user-specified region of interest.

```
perl allenminer.pl -mode search -roidef_fn ROI_DEFINITION_FILE [-age AGE]
   [-cluster_fl 1] [-xpz_list_fn XPZ_LIST_FILE]
```

- `-roidef_fn ROI_DEFINITION_FILE`

  file containing ROI definition (see *Input file formats* section)

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

- `-xpz_list_fn XPZ_LIST_FILE` file containing a list of XPZ files to process; if not specified, will parse all XPZ files located in the $specs->{allen_xpz_dir}/AGE directory specified in alnmnr.pm

- `-patterning` <0|1> (optional)

  if 1, computes gradient and patterning scores for any ROI that have been partitioned (`with -mode roiop -type partition`). If set, the output contains 2 more record types: 'S' - specificity and 'P' - patterning.

- `-lrcheck` <0|1> (optional)

  if 1, creates L/R copies of all defined ROI, and separately computes expression values for each side. It also sets the `-patterning 1` option to compute the enrichment in either side vs the full ROI. High 'enrichment' scores indicate imageseries that are asymmetric.

- `-cluster_fl` <0|1> (optional)

  if 1, runs the query in parallel using an SGE computing cluster, as specified in SGE.pm

- `-xyz_details 1` (optional)

  if 1, creates text format tab-delimited files for each processed expression file that contains detailed expression data per ROI coordinate.

  NOTE: this will create a file FOR EACH expression file processed. If you are running a query over the full expression data, this means ∼ 25,000 files for adult and ∼ 2,500 files for each developing atlas. In this case, it is probably wise to specify the `-xyz_details_out_dir` and `-xyz_details_out_subdir` options.

- `-xyz_details_compress 1` (optional; for use with -xyz_details 1)

  Compress xyz_details output files

- `-xyz_details_out_suffix SUFFIX` (optional)

  Append xyz_details output files with the suffix "SUFFIX"

- `-xyz_details_out_dir OUTDIR` (optional)

  Deposit xyz_details output files into the OUTDIR directory; defaults to the current directory

- `-xyz_details_out_subdir <0|1>` (optional)

  If specified, will deposit xyz_details output files into a sub-directory using the first character of the filename. For example, the output for Egfr_coronal_XXXX.xpz will be placed in the E subdirectory instead of directly into the output directory – which is either the current directory or a directory specified by `-xyz_details_out_dir`

2.2 `fastsearch`: Index-accelerated search of ABA brain regions

Provides a faster alternative to the `search` mode for expression queries involving named ABA brain regions. This mode is faster because instead of parsing individual XPZ files, it uses a single file that contains pre-computed `search` results for all genes in all named ABA brain regions.

```
perl allenminer.pl -mode fastsearch -query_specs_fn QUERY_SPECS_FILENAME
   [-out_fn OUTPUT_FILE_NAME] [-outfile_prefix OUTPUT_FILE_PREFIX]
```

- `-query_specs_fn <QUERY SPECIFICATIONS FILE>`

  this file describes the details of the query to be performed (see *Input file formats* section).

- `-outfile_prefix <OUTPUT FILE PREFIX>` (optional)

  specify this if you would like the results of each query to be displayed in a separate file, with this prefix. The suffix is "_QUERYNAME.allenminer.out"

- `-out_fn <OUTPUT FILE NAME>` (optional)

  specify this if you want the results of all queries to be displayed in the same output file

If nothing is specified about an output file, it displays the results in:
"aba_fast_query.QUERYNAME.PID.allenminer.out" where PID is the process id.

2.3 `simsearch`: similarity search within an optionally defined ROI

Computes the similarity of pairs of expression patterns across the whole brain or in a specified region of interest. The similarity score between two expression files is calculated as the Pearson's correlation of the expression energy values across all cubes shared between the two datasets, or across the cubes in an optionally defined ROI. This is the same scoring function as used by the AIBS's NeuroBLAST tool. The expression files contain two measures of expression for each 3D-registered voxel: intensity and density. The energy score combines these measures into a single (*ad hoc*) measure (energy = intensity · density) for use in calculating the Pearson's correlation.

```
perl allenminer.pl -mode simsearch -roidef_fn ROI_DEFINITION_FILE -
   xpz_query_list_fn QUERYLIST -xpz_query_fn QUERY_XPZ_FILE -
   xpz_target_list_fn TARGETLIST [-out_fn OUTPUT_FILE_NAME] [-cluster_fl
   1] [-age AGE]
```

- -roidef_fn ROI_DEFINITION_FILE

  file containing ROI definition (see *Input file formats* section). If no ROI is specified the similarity is computed across all voxels shared between query and target expression patterns.

- -xpz_query_list_fn FILENAME

  file containing a list of XPZ files to use as query patterns; either xpz_query_fn or xpz_query_list_fn must be specified.

- -xpz_query_fn FILENAME

  XPZ file to use as query patterns; either xpz_query_fn or xpz_query_list_fn must be specified.

- -xpz_query_target_fn FILENAME (optional)

  file containing a list of XPZ files to search against. if not specified will search against all XPZ files.

- -lrcheck <0|1> (optional)

  if 1, creates L/R copies of all defined ROI, and separately computes expression similarity values within each side.

- -cluster_fl <0|1> (optional)

  if 1 will run the query in parallel using an SGE computing cluster, as specified in SGE.pm

- -out_fn <OUTPUT FILE NAME> (optional)

  specifies the output file name; if not specified will display to a generic filename: 'results_simsearch.XXXXX.out'

- -age AGE

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

- -plane PLANE (optional)

  Specify a plane to restrict queries to those dataseries, one of: coronal or sagittal. If not specified, no restriction on plane.

2.4 (EXPERIMENTAL) make_report: Format search results for easy viewing

NOTE: Requires a local mirror of the atlas and expression files.

This run mode formats search results into an HTML file that optionally retrieves 2D ISH images from the region(s) of interest.

```
perl allenminer.pl -mode make_report -roidef_fn ROI_DEFINITION_FILE -
    search_results_fn SEARCH\_RESULTS_FILE -age AGE -out_prefix
    OUTPUT_FILE_PREFIX [-get_images IMAGESTORETRIEVE] [-num_images
    NUMIMAGES] [-sortby SORTFIELD] [-query_plane PLANE] [-hit_plane PLANE]
    [-plane PLANE]
```

- -roidef_fn ROI_DEFINITION_FILE

  file containing ROI definition (see *Input file formats* section).

- -search_results_fn SEARCH_RESULTS_FILE

  name of file containing search results to be formatted.

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

- `-query_plane <coronal|sagittal>`

  If simsearch, only format results where query data series is from the specified sectioning plane.

- `-hit_plane <coronal|sagittal>`

  If simsearch, only format results where hit data series is from the specified sectioning plane.

- `-plane <coronal|sagittal>`

  If not simsearch, only format results where hit data series is from the specified sectioning plane.

- `-sortby SORTFIELD`

  Optionally specify field on which to sort results. This field must match one of the column names listed in the header line of the results file. If not specified, uses reasonable defaults.

2.5 `roiop`: ROI operations

This run mode performs several operations on one or more ROI.

`peel`: Peel the edges of an ROI

Removes the specified number of edge layers from an ROI, like peeling a layer from an onion. It can be useful for reducing the effects of edge mis-registration artifacts.

```
perl allenminer.pl -mode roiop -type peel -roidef_fn ROI_DEFINITION_FILE -
   layers NUMBER_OF_LAYERS -out_fn NEWROI_DEFINITION_FILE [-age AGE]
```

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

`flip`: Flip an ROI about the mid-sagittal plane

Flips an ROI about the mid-sagittal plane. This is useful for analyzing expression within named brain regions of the right hemisphere, since the ABA region definitions refer only to the left hemisphere.

```
perl allenminer.pl -mode roiop -type flip -roidef_fn ROI_DEFINITION_FILE -
   out_fn NEWROI_DEFINITION_FILE [-age AGE]
```

The name of the flipped ROI in the output ROI_DEFINITION_FILE is the old name prepended with 'flip_'.

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

separate_adjacent: Remove ROI edges that border neighboring ROIs

Removes the specified number of edge layers at the border of adjacent ROIs. It can be useful in reducing the effects of edge mis-registration artifacts. This mode is similar in spirit to the roi_peel run mode, but only removes edge voxels that border other ROIs specified in the ROI definition file.

```
perl allenminer.pl -mode roiop -type separate_adjacent  -roidef_fn
   ROI_DEFINITION_FILE -layers NUMBER_OF_LAYERS -out_fn
   NEWROI_DEFINITION_FILE [-age AGE]
```

- -age AGE

  specifies the developmental stage - both for loading the correct reference atlas and for parsing the correct expression files. Must be one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult).

  if not specified, defaults to adult.

subtract: Compute the difference between two ROI

Creates a new roi definition file describing the difference between roi1 and roi2.

```
perl allenminer.pl -mode roiop -type subtract -roidef_fn
   ROI_DEFINITION_FILE -roi1 ROI1NAME -roi2 ROI2NAME -new_name
   NAMEOFNEWROI [-out_fn NEW_ROI_DEFINITION_FILE] [-age AGE]
```

- -roidef_fn ROI_DEFINITION_FILE

  file containing ROI definition (see *Input file formats* section)

- -roi1 ROI1NAME

  name of first ROI

- -roi2 ROI2NAME

  name of second ROI

- -new_name NEW_ROI_NAME

  name of difference ROI

- -out_fn NEW_ROI_DEFINITION_FILE (optional)

  name of file to display new ROI to. otherwise displays to STDOUT

- -age AGE

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

rename: Rename ROI

partition: Generate new ROIs by partitioning an ROI along RC, ML, or DV axes

```
perl allenminer.pl -mode roiop -type partition -roidef_fn
   ROI_DEFINITION_FILE -roi_basename PARTITIONED_ROI_BASENAME_ -numbins
   NUMBER_OF_BINS -axes <AP|ML|DV> [-fitted_cuts 1] [-age AGE]
```

- `-roidef_fn ROI_DEFINITION_FILE`

  file containing ROI definition (see *Input file formats* section)

- `-roi_basename BASENAME`

  specifies the prefix of the name to give to the newly created ROI partitions.

  The new ROIs will be named: BASENAME_partN where N runs from from 0 to $(numbins - 1)$.

- `-numbins NUMBER_OF_BINS`

  specifies the number of ROI partitions to create

- `-axes <AP|ML|DV>`

  specifies the axis along which to partition the ROI; AP = anteroposterior (or RC = rostro-caudal), ML = mediolateral, DV = dorsoventral.

- `-fitted_cuts 1`

  optional flag that specifies that ROI partitions should be generated with the space across the partition axis equally split among the partitions. This results in partition edges that adapt to the shape of the ROI. If this option is not specified, the partitions edges are parallel to the axis.

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

2.6 `convert`: Convert between file formats

`roi2pdb`: Convert an ROI definition file to PDB format

This run mode is useful for visualizing region of interests; *eg*, after manipulation.

```
perl allenminer.pl -mode convert -type roi2pdb -roidef_fn
    ROI_DEFINITION_FILE -pdb_fn OUTPUT_PDB_FILENAME [-age AGE]
```

- `-roidef_fn ROI_DEFINITION_FILE`

  file containing ROI definition (see *Input file formats* section)

- `-pdb_fn OUTPUT_PDB_FILENAME`

  name of PDB output file

- `-age AGE`

  Specifies the developmental stage, one of: E11.5, E13.5, E15.5, E18.5, P4, P14, P28, P56 (or adult). If not specified, defaults to adult.

`xpz2pdb`: Convert an XPZ file to PDB format

This run mode is useful for visualizing XPZ expression files in platforms without ABA BrainExplorer 2.

```
perl allenminer.pl -mode convert -type xpz2pdb -xpz_fn XPZ_filename -
    pdb_fn OUTPUT_PDB_FILENAME
```

- `-xpz_fn XPZ_FILE_NAME`

  name of ABA XPZ file to convert

- `-pdb_fn OUTPUT_PDB_FILENAME`

  name of PDB output file

xpz2txt: Convert an XPZ file to text format

```
perl allenminer.pl -mode convert -type xpz2txt -xpz_fn XPZ_filename -
   out_fn OUTPUT_FILENAME
```

- `-xpz_fn XPZ_FILE_NAME`

  name of ABA XPZ file to convert

- `-out_fn OUTPUT_FILENAME`

  name of output file

2.7 spinesearch: Search spinal cord expression patterns

Searches adult (P56) and developmental (P4) spinal cord expression patterns using flexible query patterns. Results in HTML format, view in a web browser.

```
perl allenminer.pl -mode spinesearch [-rcbits RCPATTERN] [-dvbits
   DVPATTERN] [-min_rc INTEGER] [-maxrc INTEGER] [-min_dv INTEGER] [-
   max_dv INTEGER] [-ignore_missing 1] [-color 0|1] [-out_fn
   OUTPUT_FILENAME]
```

- `-rcbits RCPATTERN`

  16-character long string of 0,1,or .(wildcard) representing ‘no expression’, ’expression’, or either, respectively in rostrocaudal sections of the spinal cord.

  Each bit represents one of 4 sections in each of cervical, thoracic, lumbar, sacral/coccygeal regions of the spinal cord.

- `-dvbits DVPATTERN`

  11-character long string of 0,1,or .(wildcard) representing expression in dorso-ventral regions of the spinal cord.

  The bits represent:

  1. Laminae 1-3
  2. Laminae 4-6
  3. Laminae 7-8
  4. Laminae 9
  5. Intermediolateral Column
  6. Gray Matter
  7. White Matter
  8. Central Canal
  9. Ventral-dorsal Midline in Gray Matter
  10. Radially Arrayed in White Matter
  11. Vascular-like in Gray and White Matter

- `-min_rc (1-16).` Minimum number of RC bits with expression

- `-min_dv (1-11).` Minimum number of DV bits with expression

- `-max_rc (1-16).` Maximum number of RC bits with expression
- `-max_dv (1-11).` Maximum number of DV bits with expression
- `-ignore_missing (0-3)`

  0: will treat missing bits as mis-matches (default) 1: will allow missing bits for '0' and '1' query bits 2: will allow missing bits for '0' query bits 3: will allow missing bits for '1' query bits

  Bit reminder: 0=no expression, 1=expression, .=wildcard, x=missing data
- `-color 0|1.` 0 = don't use color in output; defaults to 1 = use color in output.
- `-out_fn OUTPUT_FILENAME` name of output file.

2.8 Runs described in the manuscript

Several enrichment, patterning, and gradient queries are illustrated in the manuscript describing ALLENMINER (Davis and Eddy, 2009). The associated input and output files are available here: https://zenodo.org/record/29427/files/allenminer_manuscript_data.tar.gz.

- Measure gene expression in the ROI defined in neocortex.roi, using the compute cluster.

```
perl allenminer.pl -mode search -roidef_fn neocortex.roi -cluster_fl 1
    -out_fn neocortex.search_results.txt
```

- Create 5 fitted mediolateral bins from the ROI defined in cp.roi; name the new ROIs using a prefix of 'CP_5MLbins_'

```
perl allenminer.pl -mode roiop -type partition -roidef_fn cp.roi -
   roi_basename CP_5MLbins_ -numbins 5 -axes ML -fitted_cuts 1 >
   cp_MLbins.roi
```

- Calculate the patterning/gradient of expression entropy across the 5 CP ROI

```
perl allenminer.pl -mode search -patterning 1 -roidef_fn CP_MLbins.roi
    -out_fn CP_5MLbins.search_results.txt
```

- Convert ROI defined in neocortex.roi into PDB format neocortex.roi.pdb

```
perl allenminer.pl -mode convert -type roi2pdb -roidef_fn neocortex.
   roi -pdb_fn neocortex.roi.pdb
```

3 Input file formats

3.1 ROI definition file

This file specifies one or multiple ROI. The `example` directory has an example file.

It is a tab-delimited file with the following fields:

1. ROI name - required
2. ROI subname - optional, leave blank if don't want to use it
3. type of definition - one of: point, brainstrx, boxbounds

4. if point: x,y,z in Allen BrainExplorer (and .XPR) coordinates

   if brainstrx: either a single abbreviation from Allen Brain Atlas (eg CTX) or if want to specify a subset, as follows: CTX=on,OLF=off,HPF=off

   if boxbounds: one of xmin, xmax, ymin, ymax, zmin, or zmax - if an axes bound is not defined, extends to edge of axes.

5. (if boxbounds): number value or equation defining the axis boundary

   - eg, '10' to specify a hard upper axis limit of 10
   - eg, 'y * 0.5' to specify a y-dependent axis limit of half the y value

If multiple lines are specified with the same ROI name, they get appended onto the ROI definition.

3.2 Query specs file for `fastsearch` run mode

The query_specs file is a tab-delimited file with the following fields:

1. query name: arbitrary label for labeling the output. Ideally don't include spaces.
2. query type: either 'enrichment' or 'level' depending on whether you would like a ROI enrichment score or expression level
3. roi1: ABA atlas region name (*eg*, CP for caudoputamen)
4. roi2: specify a second ROI - only required if this is an enrichment query: will compute the relative enrichment of each gene in roi1 vs roi2.

4 Output file formats

These are also provided as commented header lines at the start of the files.

4.1 search: main results

Three possible entry types (E = expression levels, P = patterning information, S = ROI partition-specific expression levels)

4.1.1 Expression level entries

1. 'E' - record type
2. gene name
3. plane: coronal or sagittal
4. XPZ file name
5. ROI name; 'internal_background' refers to the whole brain
6. number of voxels in the ROI definition
7. number of voxels reported in the XPZ file within the ROI (in newest XPZ data format, all voxels have data, even if not registered as expressing)
8. total intensity over all ROI voxels
9. average intensity (total / number of ROI voxels)

10. total density
11. average density
12. total energy
13. average energy
14. ROI specificity of intensity (relative to the whole brain)
15. ROI specificity of normalized intensity (relative to the whole brain)
16. ROI specificity of density (relative to the whole brain)
17. ROI specificity of normalized density (relative to the whole brain)
18. ROI specificity of energy (relative to the whole brain)
19. ROI specificity of normalized energy (relative to the whole brain)
20. ROI specificity of number of registered voxels (relative to the whole brain)

4.1.2 Patterning entries

1. 'P' - record type
2. gene name
3. plane
4. XPZ file name
5. ROI basename
6. total ROI energy level
7. number of point information content - "number of point" or "number of voxel" measures are likely no longer useful, because the new XPZ expression files have a point for all voxels in the atlas space, even when no expression is observed.
8. normalized number of point information content
9. number of points gradient numerator
10. number of points gradient denominator
11. number of points gradient score
12. density information content
13. normalized density information content
14. density gradient numerator
15. density gradient denominator
16. density gradient score
17. intensity information content
18. normalized intensity information content

19. intensity gradient numerator
20. intensity gradient denominator
21. intensity gradient score
22. energy information content
23. normalized energy information content
24. energy gradient numerator
25. energy gradient denominator
26. energy gradient score

4.1.3 Partition-specific expression level entries

1. 'S' - record type
2. gene name
3. plane
4. XPZ file name
5. ROI name
6. numpoints bin specificity - again, number of point measures unlikely to be useful.
7. numpoints bin enrichment
8. numpoints bin value
9. numpoints total value
10. density bin specificity
11. density bin enrichment
12. density bin value
13. density total value
14. intensity bin specificity
15. intensity bin enrichment
16. intensity bin value
17. intensity total value
18. energy bin specificity
19. energy bin enrichment
20. energy bin value
21. energy total value

4.2 search: xyz_details files

Expression measures listed in the original XPZ file at any voxels in the ROI.

Tab-delimited:

1. ROI
2. X coordinate
3. Y coordinate
4. Z coordinate
5. Intensity level
6. Density of expressors
7. Energy (intensiy * density)

4.3 simsearch

Tab-delimited:

1. 'SIM' - denotes entry type
2. Query file
3. Query gene
4. Query plane (coronal or sagittal)
5. Hit file
6. Hit gene
7. Hit plane
8. ROI name
9. Similarity score
10. p-value (IGNORE)
11. Number of 'shared' cubes (voxels) with expression registered in both query and hit files.
12. Mean query expression level in shared cubes.
13. Mean hit expression level in shared cubes.

4.4 fastsearch

Two types of records: "F" for single ROI levels, "FE" for ROI1-vs-ROI2 enrichment query.

4.4.1 Single ROI results

1. 'F' - record type
2. XPZ file name
3. plane
4. query name
5. query type
6. ROI name
7. ROI density
8. ROI intensity
9. ROI energy

4.4.2 ROI1-vs-ROI2 results

1. 'FE' - record type
2. XPZ file name
3. plane
4. query name
5. query type
6. ROI1 name
7. ROI1 density
8. ROI1 intensity
9. ROI1 energy
10. ROI2 name
11. ROI2 density
12. ROI2 intensity
13. ROI2 energy
14. density enrichment
15. intensity enrichment
16. energy enrichment

4.5 spinesearch

HTML file viewable in any web browser:

1. hit number
2. Gene name (linked to ABA server)
3. Age
4. rostro-caudal pattern
5. dorsoventral pattern
6. ABA Imageseries identifer (linked to ABA server)

5 Citing ALLENMINER

A tool for identification of genes expressed in patterns of interest using the Allen Brain Atlas.
Fred P. Davis and Sean R. Eddy. *Bioinformatics* (2009). 25(13):1647-54. doi:10.1093/bioinformatics/btp288

6 Contact Information

Fred P. Davis
web: http://fredpdavis.com
email: fredpdavis@gmail.com