Contents

Understanding Horace data structures

```
%This section is aimed at showing the different kinds of data that are
%stored in memory. This is especially important if you are using a machine
%without a huge RAM, but it is also worth taking a moment to appreciate
%some of the subtlties for use later
```

sqw data

display a summary of an sqw object

```
head(alignment_slice1)
```

```
2-dimensional object:
-------------------------
Original datafile: C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_real_file.sqw
            Title: <none>

Lattice parameters (Angstroms and degrees):
        a=2.844          b=2.844           c=2.844
    alpha=90          beta=90          gamma=90

Size of 2-dimensional dataset: [434x434]
    Plot axes:
        \zeta = -5.015:0.03:8.005 in [\zeta, 0, 0]
        \xi = -5.015:0.03:8.005 in [0, \xi, 0]

    Integration axes:
        -0.05 =< \eta =< 0.05 in [0, 0, \eta]
```

```
          -10 =< E =< 10
```

Number of contributing pixels

```
size(alignment_slice1.data.pix)
```

```
ans =

           9      275523
```

Size of signal array

```
size(alignment_slice1.data.s)
```

```
ans =

   434   434
```

Size of axes (nb they are bin boundaries, c.f. size above):

```
size(alignment_slice1.data.p{1})
```

```
ans =

   435     1
```

Notice number of pix is same as number of pix!!

```
sum(sum(sum(alignment_slice1.data.npix)))
```

```
ans =

      275523
```

For more detailed investigation, convert into a structure array:

```
gg=get(alignment_slice1);

%Information about how file was created
nfiles=numel(gg.header)
```

```
nfiles =

    46

```

Sample orientation

```
cu=gg.header{1}.cu
cv=gg.header{1}.cv

%to see size in memory, in Matlab command window slice on the workspace
%panel, then the "view" menu, the select "choose columns", and ensure that
%"bytes" is clicked. You can then see the memory taken by each object in
%the workspace. Two equivalent dnd and sqw objects take up vastly different
%memory

my_d2d=d2d(alignment_slice1);
%See how this is much smaller in memory
```

```
cu =

    0.9908   -0.0002   -0.0002


cv =

    0.0002    0.9906   -0.0159

```

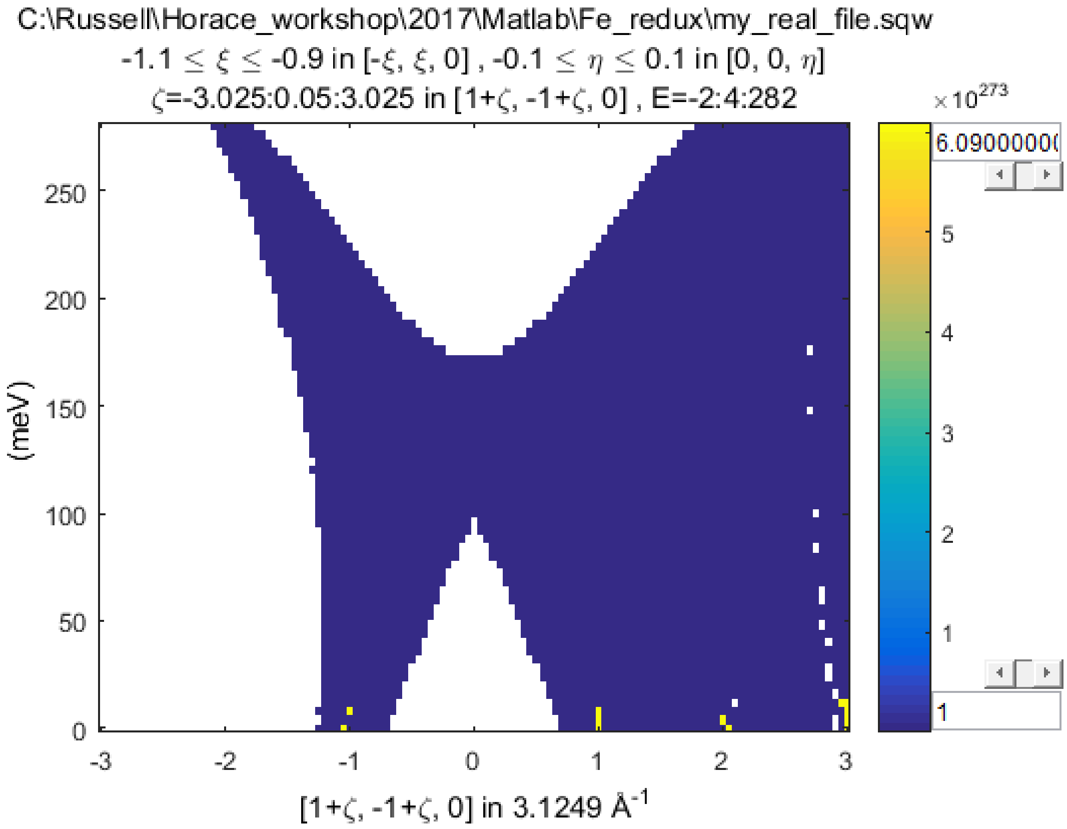
dnd data

```
%repeat the above (with data field) for my_d3d, to confirm that it is the
%same. But metadata has been thrown away.
```

Binary and unary operations

```
%Unary operations:
w1=cut_sqw(sqw_file,proj,[-3,0.05,3],[-1.1,-0.9],[-0.1,0.1],[0,4,280]);
w1_d2d=d2d(w1);

w1_exp=exp(w1); w1_d2d_exp=exp(w1_d2d);

plot(w1_d2d_exp)%NB: discovered an error with log, in particular how it handles zeros!
```

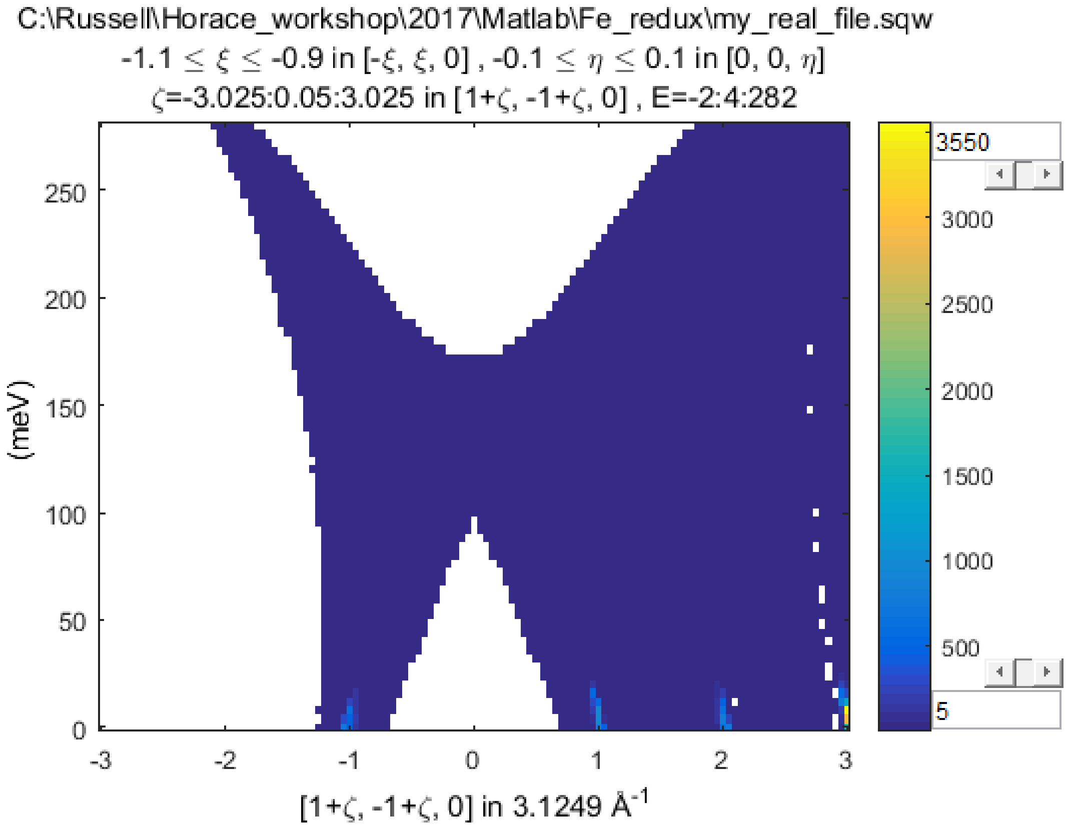
```
--------------------------------------------------------------------------------
Taking cut from data in file C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_real_file.sqw...
Step   1 of    1; Have read data for 2647039 pixels -- now processing data... ----->  retained  913103 pixels
-----------------------------
Inside cut_data:
  Timings for reading:
        Elapsed time is 0.55999 seconds
            CPU time is 0.5772 seconds

  Timings in accumulate_cut:
        Elapsed time is 0.066002 seconds
            CPU time is 0.1872 seconds

  Timings for handling pixel information
        Elapsed time is 0.103 seconds
            CPU time is 0.2496 seconds
-----------------------------

Number of points in input file: 139573798
         Fraction of file read: 1.8965 %   (=2647039 points)
     Fraction of file retained: 0.6542 %   (=913103 points)

Total time in cut_sqw:
Elapsed time is 1.592 seconds
    CPU time is 1.9188 seconds
--------------------------------------------------------------------------------
```

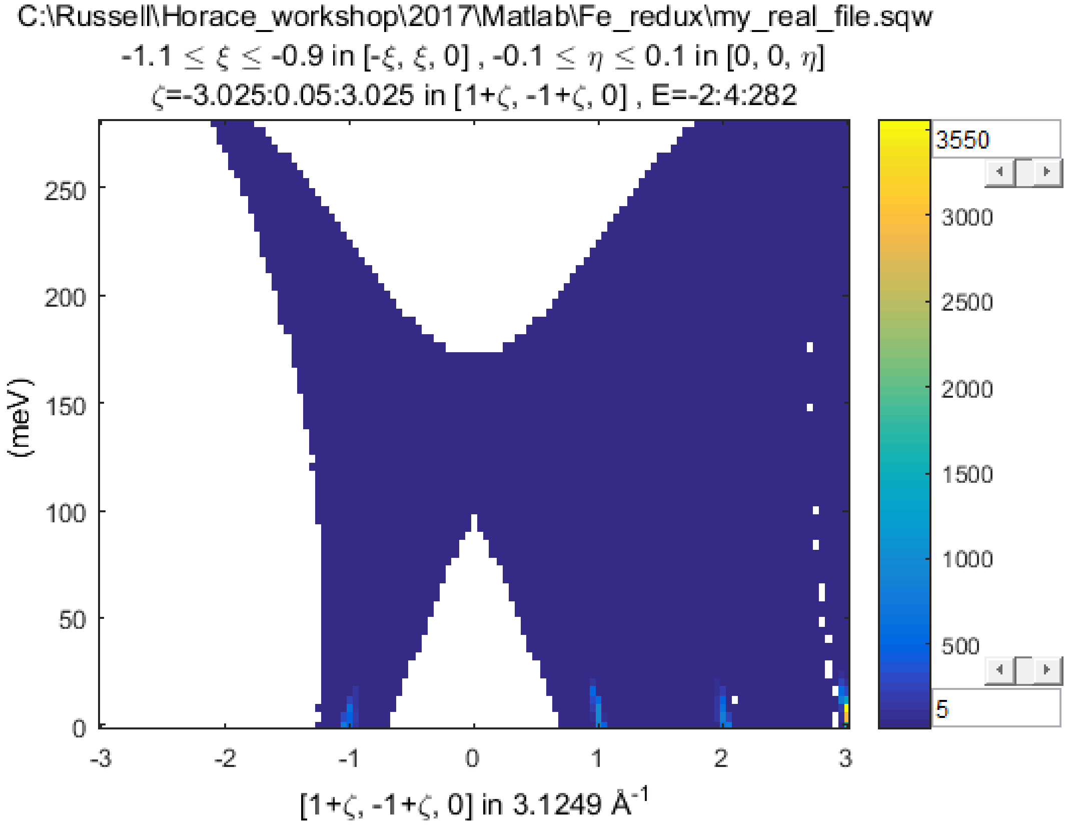


Binary operations:

Add a number:

```
plot(w1+5);
keep_figure
plot(w1_d2d+5);
keep_figure
```





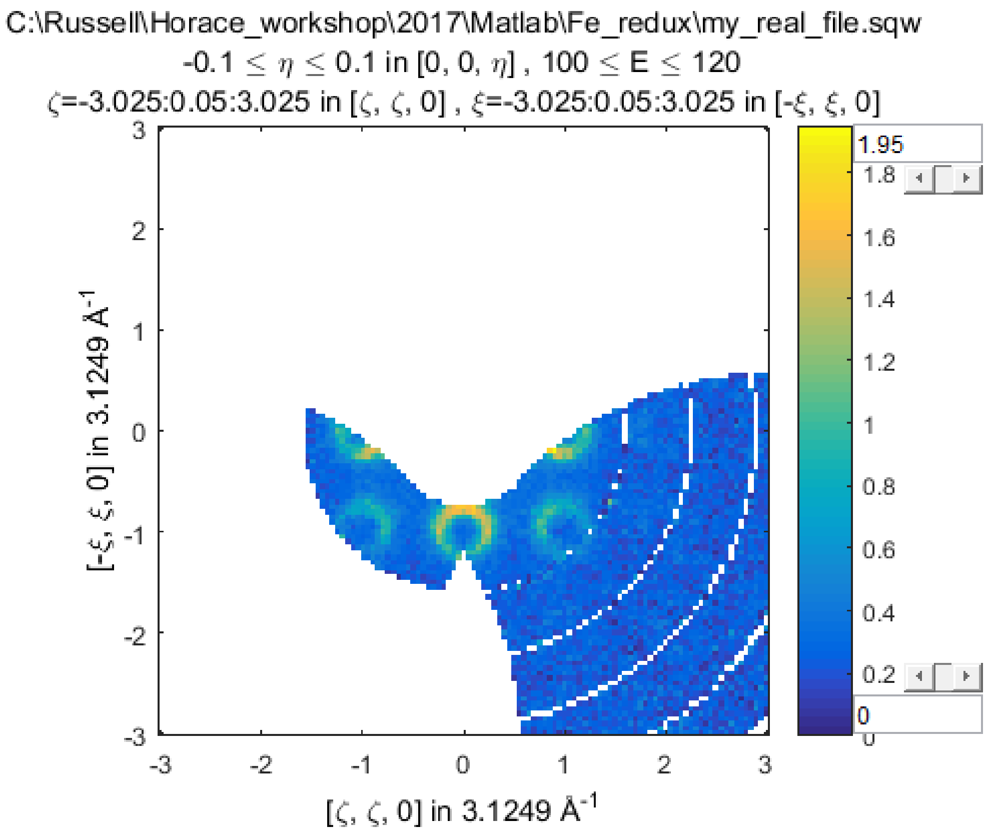
Add two objects:

```
w2=cut_sqw(sqw_file,proj,[-3,0.05,3],[-3,0.05,3],[-0.1,0.1],[100,120]);
w3=cut_sqw(sqw_file,proj,[-3,0.05,3],[-3,0.05,3],[-0.1,0.1],[120,140]);

w2_d2d=d2d(w2);
```

```
w3_d2d=d2d(w3);

plot(w2_d2d+w3_d2d);
keep_figure;

%plot(w2+w3);%this does not work because of different size of pix arrays
%(if pix arrays exist (sqw object) they are added. If not (dnd) then signal
%arrays added. Generally the pix arrays will not be commensurate so adding
%sqw objects to one another is essentially impossible [deliberately so])
%keep_figure;
```

```
--------------------------------------------------------------------------------
Taking cut from data in file C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_real_file.sqw...
Step   1 of    1; Have read data for 1021806 pixels -- now processing data... -----> retained  678225 pixels
-----------------------------
Inside cut_data:
  Timings for reading:
        Elapsed time is 0.092003 seconds
            CPU time is 0.156 seconds

  Timings in accumulate_cut:
        Elapsed time is 0.045998 seconds
            CPU time is 0.156 seconds

  Timings for handling pixel information
        Elapsed time is 0.057014 seconds
            CPU time is 0.1872 seconds
-----------------------------

Number of points in input file: 139573798
         Fraction of file read: 0.7321 %   (=1021806 points)
     Fraction of file retained: 0.4859 %   (=678225 points)

Total time in cut_sqw:
Elapsed time is 0.96199 seconds
    CPU time is 1.3416 seconds
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Taking cut from data in file C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_real_file.sqw...
Step   1 of    1; Have read data for 1494275 pixels -- now processing data... -----> retained  706731 pixels
-----------------------------
Inside cut_data:
  Timings for reading:
        Elapsed time is 0.12901 seconds
            CPU time is 0.1248 seconds

  Timings in accumulate_cut:
        Elapsed time is 0.036995 seconds
            CPU time is 0.093601 seconds

  Timings for handling pixel information
        Elapsed time is 0.070999 seconds
            CPU time is 0.1872 seconds
-----------------------------

Number of points in input file: 139573798
         Fraction of file read: 1.0706 %   (=1494275 points)
     Fraction of file retained: 0.5063 %   (=706731 points)

Total time in cut_sqw:
```

```
Elapsed time is 0.982 seconds
    CPU time is 1.2324 seconds
--------------------------------------------------------------------------------
```



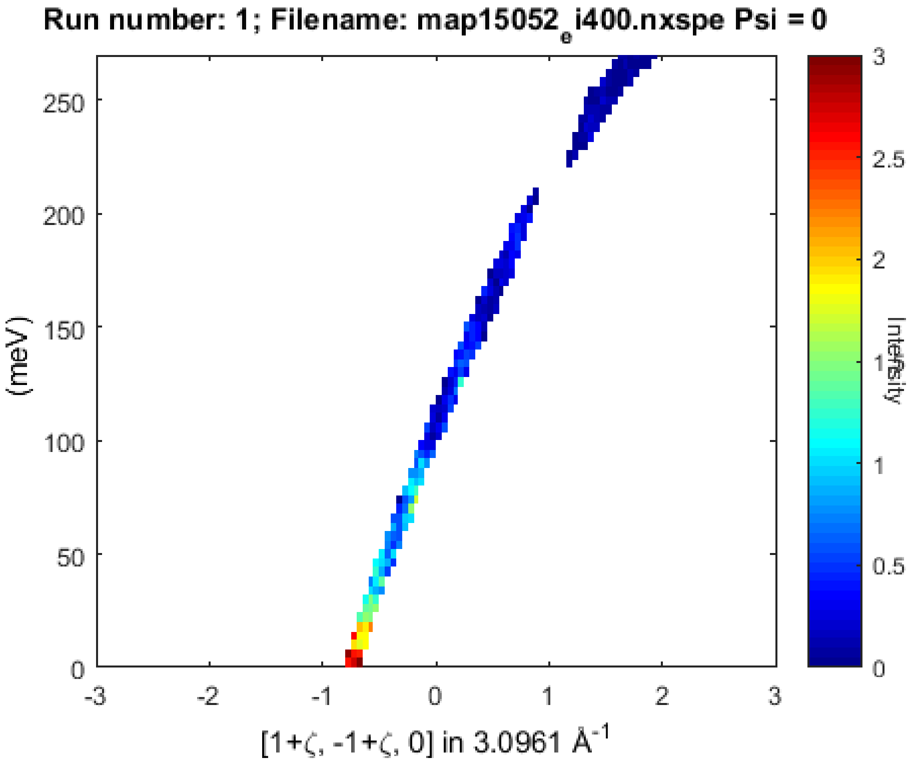
Output data to file in various formats:

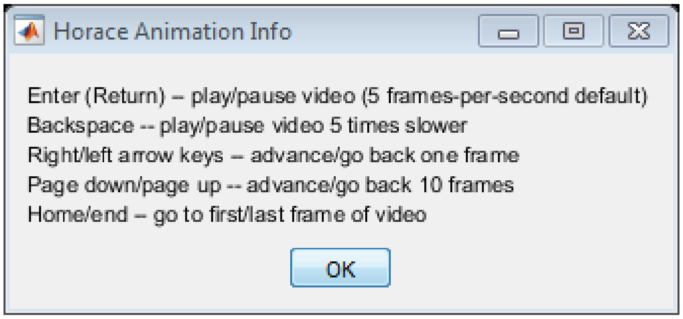
```
%Save to sqw file:
save(w2,[data_path,'my_w2.sqw']);

%Read it back in again
w2a=read_sqw([data_path,'my_w2.sqw']);

%Save dnd:
save(w2_d2d,[data_path,'my_w2.d2d']);

%Read it back in again
w2b=read_dnd([data_path,'my_w2.d2d']);

%Save data to ascii file, for use in some other program:
save_xye(w2_d2d,[data_path,'my_w2_ascii.dat']);
```

```
*** Writing to C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_w2.sqw...
*** Writing to C:\Russell\Horace_workshop\2017\Matlab\Fe_redux\my_w2.d2d...
```

Examine sqw data with run_inspector - this allows you to look at data on a run-by-run basis. Useful for spurion hunting

```
run_inspector(my_slice,'ax',[-3,3,0,270],'col',[0,3]);
```





Enter (Return) -- play/pause video (5 frames-per-second default)
Backspace -- play/pause video 5 times slower
Right/left arrow keys -- advance/go back one frame
Page down/page up -- advance/go back 10 frames
Home/end -- go to first/last frame of video

Published with MATLAB® R2015b