Fitting multiple Gaussians using eis_auto_fit_gen

Ver. 1, 6-Nov-2008, Peter Young

The routine eis_auto_fit takes the output structure from eis_getwindata and fits a single Gaussian to an emission line from the structure. Often, however, a wavelength window contains more than one line and so it is necessary to fit multiple Gaussians.

This document describes the routine eis_auto_fit_gen which allows the user to fit multiple Gaussians. Further routines are available for performing the slit tilt and orbital corrections, and browsing the fit results.

The first step in performing a multiple Gaussian fit is to specify a fit template that specifies the number of Gaussians and some initial parameters.

eis_make_fit_template

After prepping the EIS data (creating a level-1 fits file with name l1name), extract a wavelength window using eis_getwindata. We will use as an example the data-set taken at 22:32 on 2007 January 20:

```
IDL> l1name='eis_l1_20070120_223207.fits'
IDL> wd=eis_getwindata(l1name,264.2)
```

The wavelength window to be fitted contains three emission lines: S X 264.2, Fe XIV 264.8 and Fe XVI 265.0.

To create a fit template, do:

```
IDL> eis_make_fit_template, wd
```

You will see a spectrum appear showing the average spectrum for the data window. Instructions as to what to do appear in the IDL prompt window. Do the following:

1. Click once on the plot window at the approximate level of the spectrum background.
2. For each of the three emission lines, click once at the approximate peak of the line
3. When finished, click once in the margin of the plot (i.e., outside of the plot axes). This exits the program.

Now check in your working directory and you will see a file called 'fit_template.txt'. Browse this file and you will see that it contains details of the fit parameters you have chosen. E.g., one portion of the file is:

```
No. of Gaussians:   3
No. of background parameters:   2
Initial guess for background level:     82.2
Gaussian   1
Initial guess for wavelength:        264.253
Initial guess for peak:              2016.5
```

Initial guess for Gaussian width: 0.030
Allowed range for wavelength: 264.103 264.403
Allowed range for width: 0.020 0.060

Some things to note about this file:

1. The initial guess of the line width (note: the Gaussian width, not the full width at half maximum) is set the same for each line.
2. The centroid and width of each line are constrained to lie within certain bounds. This helps make robust fits, and also stops lines being 'swapped' during the fit optimization process (e.g., the Gaussian for 264.8 may actually end up fitting the 265.0 line, and vice versa)
3. The file can be easily hand edited. E.g., you may want to replace the centroid estimate with the rest wavelength of the emission line.

You can now take this template and use it as input for the Gaussian fitting routine.

eis_auto_fit_gen

This routine is based on the routine eis_auto_fit, which already exists in the EIS software tree for fitting single Gaussians. Before using it, we first do:

```
IDL> wd2=eis_bin_windata(wd,xbin=3,ybin=3)
```

This performs spatial binning (3x3) on the data array to improve the signal to noise.

To use eis_auto_fit_gen do:

```
IDL> initdata=eis_read_fit_template('fit_template.txt')
IDL> eis_auto_fit_gen, wd2, initdata, fitdata
```

The first routine reads the fit template into an IDL structure (INITDATA). Eis_auto_fit_gen takes the data (WD2) and the fit template data (INITDATA) to then go through each spatial pixel and fit a 3-Gaussian function. The results are stored in FITDATA.

IMPORTANT NOTE: if you have used the eis_auto_fit routine previously, please note that the format of the FITDATA output structure is different to that from eis_auto_fit.

Tilt and orbit corrections – eis_fit_correction

To perform the slit tilt and orbit corrections, simply do:

```
IDL> eis_fit_correction, wd2, fitdata, line=1
```

By specifying line=1 it means we are using the 2nd line in the fit (Fe XIV 264.8) for performing the orbit correction since this is the strongest line of the three.

Note that fitdata is modified by eis_fit_correction, but it's only the tag 'offset' which is modified – everything else remains the same. You can thus call the routine again (e.g., trying line=0 instead of line=1) without having any adverse effects.

Note that the routines eis_orbit_spline and eis_tilt_correction are used by eis_fit_correction to perform the corrections. Check the Appendix for more information on eis_fit_correction and how it relates to the method used for the eis_auto_fit routine.

Viewing the fits

For the single-Gaussian fitting routine eis_auto_fit, a widget-based routine eis_fit_viewer was available for viewing the quality of the fits. Exactly the same routine can be used for the output of eis_auto_fit_gen. Simply do:

```
IDL> eis_fit_viewer, wd2, fitdata, line=1
```

to study the fits for Fe XIV 264.8. Try using line=0 and line=2 to view the other lines.

Appendix

The eis_orbit_spline and eis_tilt_correction routines were written assuming the input was the structure generated by eis_auto_fit (the single-Gaussian fitting routine). It was decided that the eis_auto_fit structure was not suitable for the multiple-Gaussian case, so the eis_auto_fit_gen output structure has a different format.

Rather than change the eis_orbit_spline and eis_tilt_correction routines to deal with the new format from eis_auto_fit_gen, it was decided to create a routine that converts the eis_auto_fit_gen structure into the eis_auto_fit structure for one of the Gaussians. This routine is called eis_convert_fitdata and is called as:

```
IDL> outdata=eis_convert_fitdata(fitdata,line)
```

The input LINE tells the routine to take the fit data from one of the fitted Gaussians and write it out into the old eis_auto_fit format. E.g., in the example data-set, one would specify LINE=1 to send the 264.8 fit data into OUTDATA.

By doing this we can now re-use the eis_tilt_correction and eis_orbit_spline routines that already exist.