Superfit 3.5

Author: Andy Howell

Introduction

Superfit is a tool designed to find the best match to an observed spectrum from a library of hundreds of previously observed spectra of supernovae and other objects. Superfit is written in IDL and requires IDL to run. It is meant as a tool to help human classifiers. While there are some features that attempt to automate typing, such as epoch determination, these have not been extensively tested and could be vastly improved.

It can be launched via the command line or by using the graphical user interface (gui). The output is stored in a file which lists the best matches in order of best fit to worst fit, along with the corresponding derived parameters. A companion program for viewing the results is called supergraph. It is a graphical user interface which allows the user to view the output file, select one of the matches, change the parameters, view a graph of the results.

The reference for superfit is: Howell et al. 2005, ApJ, 634, 1190

Version

Superfit version 3.5 was developed in August 2012, and released widely (via github) in August 2013. The underlying code is largely the same, though the library has been updated and there have been several bug fixes. A list of changes can be found at the end of this document.

IDL requirements: It was tested under IDL 6.4.1. It should work under IDL 8.1 as well.

To check the version, at the IDL prompt, type:

```
IDL> sfgui,/version
```

This works for sf.pro and sggui.pro too.

What's new in v. 3.5

- Updated library (thanks largely to Sagi Ben-Ami, and some to Christopher White). Sagi has called his version with this library SuperSuperfit, but I'd like not to use that name to avoid confusion. I have altered the updated libraries they provided, cropping some spectra and fixing epochs and redshifts of others.

- New in this library: SLSNe, 2002cx-likes, SN 2007bi, SN 2002bj, etc.

- SNe Ib/c now split into SN Ib and SN Ic. New categories: Others, 2002cx.

- Fixed a bug where at high redshift some fits were giving underflow (divide by zero) errors.
- Fixed a bug where postscript plots had incorrect colors.
- Redshift is now given to 4 decimal places
- You can now import templates with up to 20,000 lines (this is easy to increase if necessary)
- New feature: You can start sfgui with a switch: /nogal . This unchecks all but one host galaxy and sets the maximum galaxy contribution to zero.
- New feature: You can start sfgui with a keyword, 'library' to preselect the SN template library so you don't have to select it from the dropdown menu. The choices are: 'Ia','Ib','Ic',II','Others','allsne', '2002cx' and 'snelt10d' for supernovae less than or equal to 10 days after max. e.g. sfgui,library='Ia'
- Should now work with IDL 8.1

Installation and setup

If you already have superfit running, but are upgrading to version 3.5, I recommend renaming your old superfit directory to something like “old_superfit” and installing the new version into your superfit directory. That way you don’t have to change any of the paths set in any startup files.

You need to install astrolib, the astronomical user's library for IDL before running superfit. Instructions can be found at http://idlastro.gsfc.nasa.gov.

You can download superfit from github at: https://github.com/dahowell/superfit. You should install it into its own directory. I call mine “superfit.”

Alternatively, if you have git installed, from the command line you can type:

```
git clone https://github.com/dahowell/superfit.git PATH/TO/
INSTALLATION/superfit
```

or by simply executing

```
git clone https://github.com/dahowell/superfit.git
```

while in the directory where you'd like to insert the superfit directory. Then updates can be done by going to the superfit directory and running

```
git pull
```

Next set this path as the SF_INSTALLDIR environment variable by putting a line like the following in your .cshrc or .bashrc file (or whichever file you use to set environment variables).

bash: `export SF_INSTALLDIR=/home/howell/idl/superfit`

csh: `setenv SF_INSTALLDIR /home/howell/idl/superfit`

Of course you should put the path to your own installation.

IDL startup files

At the end of my .bashrc, I also have the following:
`source ~/.idlenv`

which sets up certain things for IDL. My .idlenv file has commands like this:

```
export IDL_STARTUP='/home/howell/.idlstartup'
```

Again the syntax "export" is because I'm using bash. This tells IDL to run the commands in my .idlstartup file when IDL first runs.

In your .idlstartup file you should have the following commands:

```
! PATH = expand_path('/home/howell/idl/') + ':' + !PATH
! PATH = expand_path('+/home/howell/idl/superfit') + ':' + !
PATH
astrolib
defsysv,'!top_color',0b
defsysv,'!top_colorx',0b
;defsysv,'!top_colorps',0b
defsysv,'!top_colorps',255b
defsysv,'!colorstr',''
defsysv,'!colorstr_ps',''
defsysv,'!colorstr_x',''
deepcolor
! p.background=colordex('white')
! p.color=colordex('black')
```

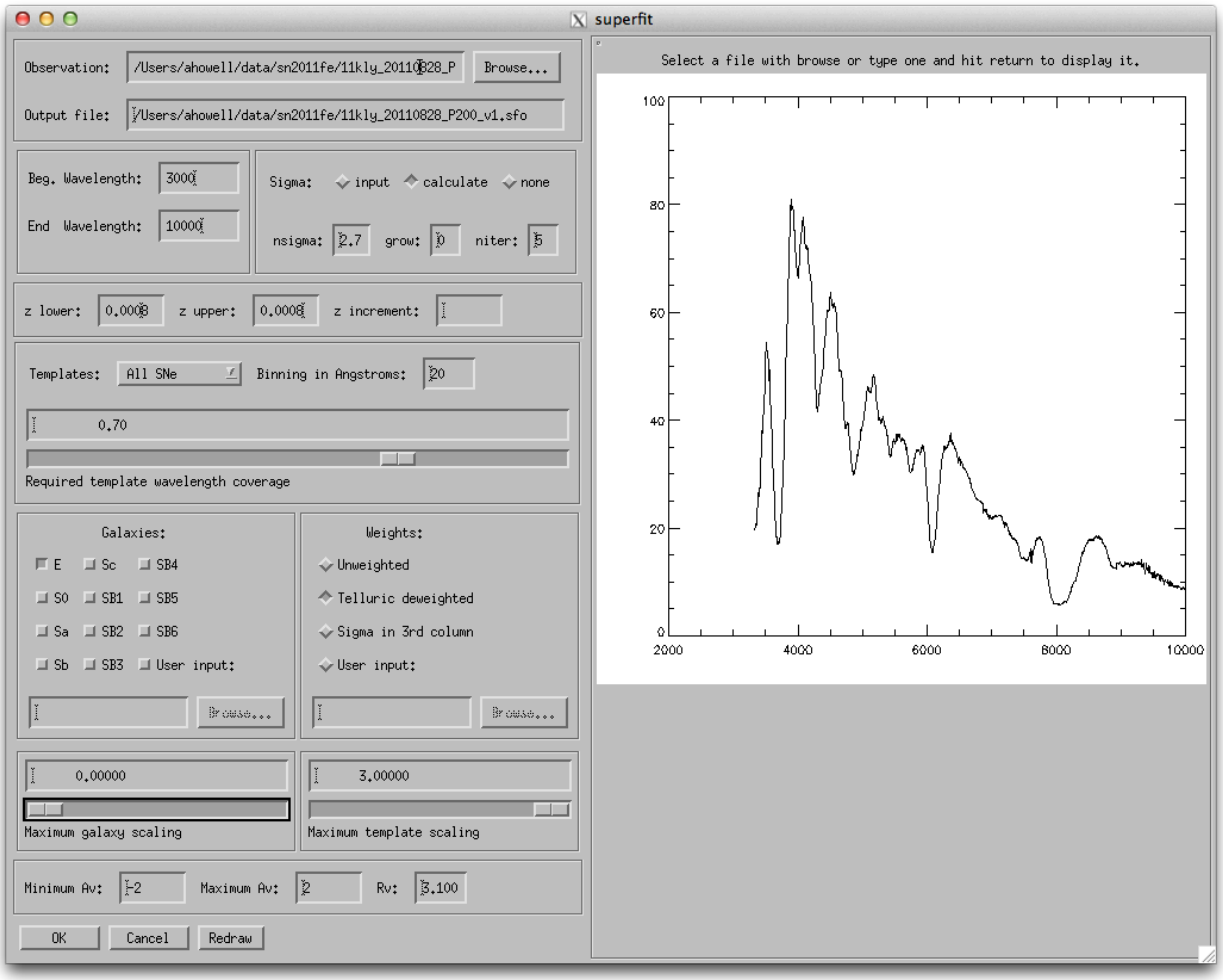
Running superfit with graphical user interface (gui)

To start the superfit with the graphical user interface (gui), go to the directory with the spectra you want to match and start IDL. Then type sfgui at the IDL prompt. The program should start with the default parameters. It is best (though not necessary) to complete each element in the gui form, working down the page from the top, because responses to certain key elements can provide useful feedback to other elements farther down in the form. The gui is designed to

need minimal input from the user. The only required input is a selection of the input file. Once an input file is selected the user can hit the OK button to start superfit. Even if the other elements of the form are not filled in, superfit will be launched with its default values.

If you have previously run superfit on an object and want to load those values into the gui, just give superfit the previously generated output file like this:

```
IDL> sfgui,file='example.sfo'
```



We will now discuss each of the elements of the gui in detail:

Input file selector

Input spectra can be in either fits format or ascii format. If the spectrum is in ascii format the first column should be wavelength in Angstroms and the second should be flux in F-lambda, although the scaling does not matter. Optionally, there may be a third column containing the flux error. To choose an input spectrum either hit the "Browse..." button to select it using a dialog box or type the filename into the selection box. The default filter setup is set to first look

for files with the '.dat' extension, though it is not necessary that input files end with this extension. Once the file is loaded a graph is displayed. If the filename ends in '.dat', '.asc', or '.fits' an output filename can automatically be generated (see below).

Output file selector
If the input file ends in '.dat', '.asc', or '.fits', when the user either completes the selection dialog or hits return, the program will automatically generate an output file name by stripping off the extension and replacing it with '.sfo' for **s**uper**f**it **o**utput file. This name is printed in the output file selector box. It is then possible to edit the name of the output file. If any text is already in the output file selector text box then no attempt will be made to automatically generate an output file name.

w1 and w2 (lower and upper wavelength cutoffs)
The ends wavelengths of spectra often have low signal to noise or other edge effects so sometimes it is necessary to crop them. For this purpose, the user can enter a lower (*w1*) and upper (*w2*) wavelength cutoff in Angstroms.

Sigma clipping
The section of the form to the right of the beginning and ending wavelength boxes controls sigma clipping. This can be used to eliminate cosmic rays or other deviant data points from the spectrum. Points greater than *nsigma* are clipped. If *grow* is set to a nonzero value then *grow* pixels on either side of the clipped pixel are also clipped. The error spectrum used for sigma clipping can either be provided by the user (in the "weights" section below) or it can be calculated from the input spectrum. If it is calculated from the input spectrum, you can perform several iterations of sigma clipping by setting *iter* to a number greater then one.

zl, zu, zi (lower redshift, upper redshift, and redshift increment)
Superfit will begin at the lower redshift limit (*zl*), and try to fit the input spectrum with a library template spectrum shifted to that redshift. It then increases the redshift of the template by the redshift increment *zi* and tries again. It keeps increasing the redshift of the template by *zi* until the upper redshift limit, *zu*, is reached. Thus the spectra are kept in the observers frame. If none of *zl, zu*, or *zi* text boxes are filled in, then zero redshift is assumed. If *zl=zu*, then *zi* is ignored.

Binning in wavelength
All spectra are rebinned to the bin size (in Angstroms) listed here. The running time of the program is inversely proportional to the bin size, so a run with a bin size of 20A will execute roughly 20 times faster than a bin size of 1A. The default value for *disp* is 20, though the matching algorithm still works quite well at 50A.

Template library selector
The user can select a template library via a droplist. In the following table 'button name' is the text printed on the droplist, 'number' is the number of supernovae in that library, and 'rootname' is the root filename of that library. The IDL save file for a given library is of the form rootname.idlsave. The list of files that are in a given library is is of the form rootname.list.

Once a template library is selected a graph of the number of available templates for that selection appears in the display window. The behavior of this graph depends on which values are already entered into the gui. If *w1* and *w2* have values, but none of the *z* fields are populated, then each spectrum in the library is represented by a horizontal line spanning that spectrum's wavelength coverage. Blue vertical lines are drawn at *w1* and *w2*. If a template spectrum has full wavelength coverage over the range [*w1,w2*], then it appears green. If it has only partial wavelength coverage then it is red. To check for wavelength coverage at a particular redshift, set *zl=zu*, and reselect the template library. To check for wavelength coverage over a range of wavelengths, fill in *zl, zu*, and *zi* and reselect the template library. In this case the graph appearance changes to redshift along the x-axis and number of supernovae with full wavelength coverage at each redshift along the y-axis.

Underneath the template library droplist is a slider that says: "*Required template wavelength coverage.*" If this slider is set to 1.0 then the template must have wavelength coverage from *w1* to *w2*. Any other fractional wavelength coverage requirement can be set by either moving the slider or typing the fraction into the text box. If a template does not have the required wavelength coverage, it is skipped and its goodness-of-fit value (S) is listed as 999.99 in the output file. Requiring full wavelength coverage ensures that each template is compared in a consistent way (each over the same wavelength range). However, this approach has the disadvantage that you are cutting down the total number of templates in the comparison. At high redshift this can be a problem, since you are comparing the input spectrum to the UV portion of local templates, and not many have UV coverage. In the output file, the rightmost column lists the fractional wavelength coverage of the template, *sfrac*.

Galaxy list

Superfit can attempt to fit and subtract host galaxy light from the observed SN spectrum. The user can select from 11 host galaxy templates: E,S0,Sa,Sb,Sc, and SB1-6 (starburst galaxies), or specify his or her own galaxy template. To use a user template click the "User template:" checkbox and either type in the filename or select it with the "Browse..." button. Note that user-provided galaxies are assumed to be at the same redshift as the SN (i.e. not deredshifted), while library galaxy spectra are at z=0.

More than one host galaxy template can be selected at a time, but note that the program running time is proportional to the number of galaxies selected. All galaxies are smoothed versions of those in Kinney et al. except the Sc galaxy, which is from the pegase models. Other galaxy modes and more information about the galaxies can be found in the "gal" directory of the superfit installation.

Weighting function list

The user can select a weighting function or supply his or her own. The default is "Telluric deweighted". which has a weight of unity across the entire spectrum, but higer errors (and thus lower weight) around telluric features. Another choice is "Sigma in 3rd column," If this is checked, the input file observation file should be in ascii format and have errors listed in the third column. The fourth choice is "User input". User supplied weights should be in two columns -- the first is wavelength in Angstroms and the second is the error value at that wavelength.

Error spectra/weighting functions are rebinned along with the input, template, and galaxy spectra. Spectra are weighted by the inverse of the variance. This weighting is used to weight the goodness-of-fit criteria and it is also used to rebin the spectrum.

Maximum galaxy and template scale sliders
Input spectra and host galaxy spectra are initially scaled so that their median value is 1. The fitting algorithm is then allowed to multiply both the template and the host galaxy by a scale factor of between 0 and *maxscale* to achieve the best fit. The *maxscale* can be set to a value of between 0 and 3 by either adjusting the slider or typing text into the text box above the slider. Set the galaxy slider to zero to force the galaxy contribution to be zero in the final fit. A setting of 3 allows any galaxy scaling, up to a scale factor of 3 (this upper limit was chosen empirically). Alternatively, setting the template *maxscale* slider to zero forces the fit to have no contribution from the supernova template.

Reddening parameters
Superfit will estimate the reddening of the input spectrum. The user can choose a reddening law, *Rv* (the default is 3.1), and specify the maximum allowable reddening in magnitudes of visual extinction, Av. The default value for *max Av* is 3.0. The minimum reddening (actually bluening if negative), *min Av*, can also be set. The default value is -3.0. It is useful to allow bluening of the template spectrum because some may not have been dereddened. To allow no reddening correction, set *min Av* and *max Av* to zero. Note that the "reddening" determined in this way may not correspond to physical reddening since there is often a degeneracy between this value and the type of host galaxy subtracted, and not all library spectra have been dereddened.

The superfit output file

Superfit generates an output file (.sfo) listing templates in order of goodness of fit (S). The columns are:

abbrev.	cols	description
tempfile	0-29	The name of the template in the form sne/type/name.xxx.dat, e.g. sne/Ia/sn1981b.p21.dat. Here type is either Ia, Ib, Ic, II, 2002cx, or Others. If xxx begins with p then it indicates days past B maximum. An m indicates days before maximum. SNe at maximum are labeled 'max'. A 'u' indicated that the date of maximum is unknown. If there are more than one spectra of a supernova with an unknown maximum, then the first is called 'u00' and each after that is called 'uxx' where the xx is the number of days after the first spectrum.
S	31-38	The goodness of fit parameter: S=weight*(observed-cc*10^(-Av*redlaw/2.5)-ff*galaxy)/sfrac
z	40-46	The best-fit redshift for a given template.

abbrev.	cols	description
galaxy	48-50	Type of best-fit host galaxy. If a user input galaxy is the best fit, then 'inp' is listed here and the name of the file is listed below in common parameters section.
Av	52-58	Magnitudes of visual extinction, Av
cc	60-66	Scale factor derived for template. A value between 0 and 3.
ff	68-74	Scale factor derived for host galaxy. A value between 0 and 3.
galfrac	76-81	Ratio of galaxy to total light in the V band.
sfrac	83-86	Fractional wavelength coverage: sfrac=(template end wavelength-template begin wavelength)/(w2-w1), where template end wavelength <= w2, template begin wavelength >= w1.

After the ordered list of templates, the output file contains a list of the parameters that were set for that run in the format ';;parameter=value'. A listing of these parameters can be found in the section which describes the command line version of superfit. The only additional parameters listed here in the output file that are not listed there (because they are not input parameters) are *beginw* and *endw*. *Beginw* is the beginning wavelength in the rebinned spectrum. *Endw* is the ending wavelength. Note that these are not necessarily the same as *w1* and *w2*. For example, if w1 (which is entered by the user) is less than the value of the first wavelength, then *beginw* > *w1*.

Running supergraph

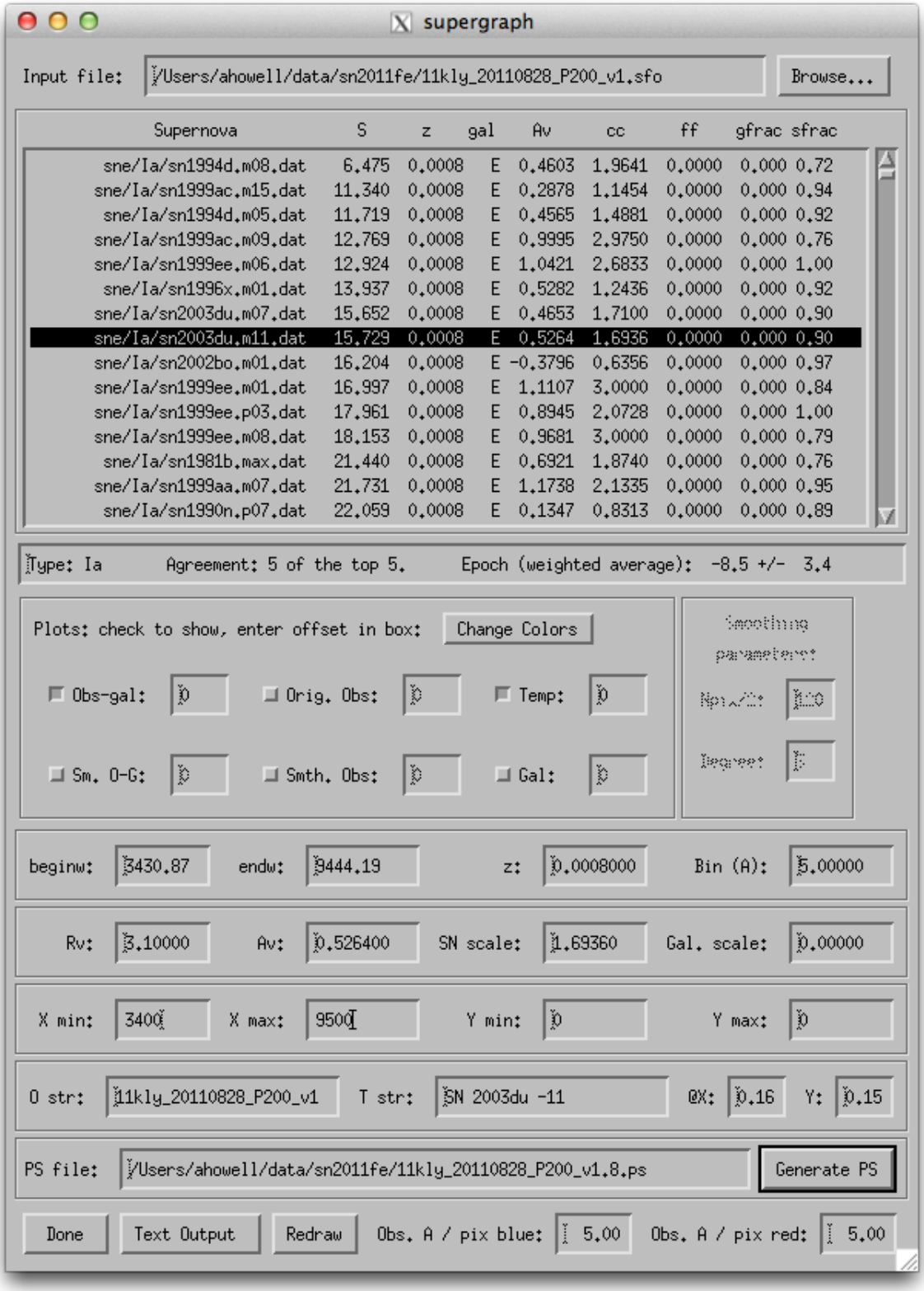
To start supergraph with the gui interface, type sggui at the IDL prompt. You may start sggui with an optional parameter, *file*, the name of the superfit output file, e.g.:

```
IDL> sggui,file='example.sfo'
```

If *file* is not specified at startup, then you may pick the superfit output file to load by either typing it into the file selection text box or selecting it with the browse button. When an sfo file is loaded, supergraph reads in the observed spectrum indicated in the sfo file. When a file is loaded its contents are read into the supergraph display. You can then select one of the lines with the mouse as shown below.

When a template is selected, a graph of it pops up into a new graphing window and its key parameters are read into text boxes in the supergraph gui. You may change the values in these boxes and hit the redraw button to see the graph with the new parameters. Most of the parameters are the same as those listed above in the superfit output file section. Additionally there are the graphing parameters *xmin, xmax, ymin*, and *ymax*. These determine the x- and y-

axis limits on the plot window. By default they are set to zero which lets IDL choose its own limits.



You may select which lines to plot by checking the appropriate checkbox:

- *Obs-gal:* observations minus the galaxy spectrum.
- *Sm. O-G:* smoothed version of the observations minus the galaxy, with smoothing parameters set by the panel to the right of the plot selection panel.
- *Orig. Obs:* the original, uncropped, unbinned un-galaxy-subtracted observations.
- *Smth. Obs.:* the original observations smoothed.
- *Temp:* the template spectrum.
- *Gal*: the template galaxy that was subtracted. Beside each plot type is a box listing the offset. These can be changed to control the relative position of the plots on the screen.

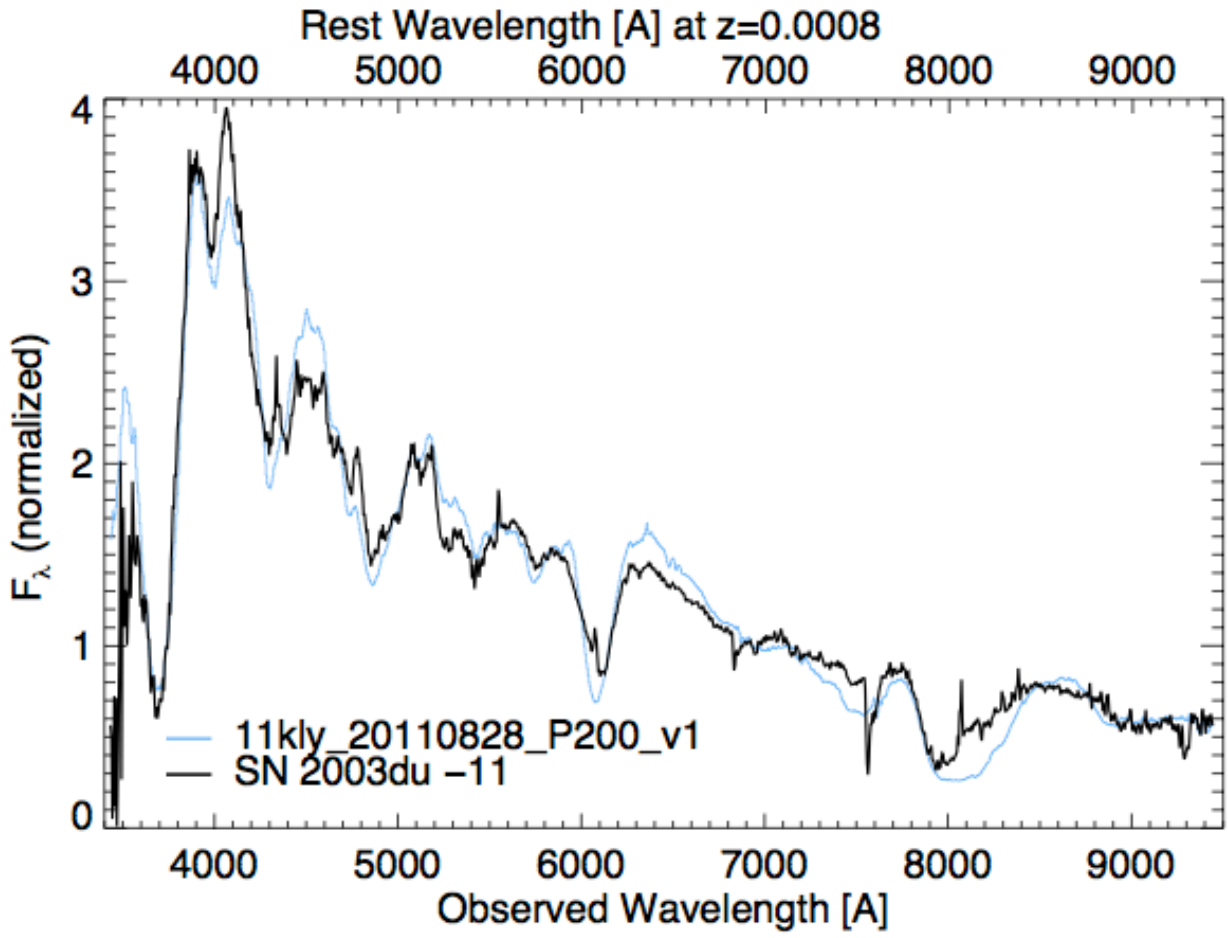
If a smoothed plot is selected, the *Smoothing parameters* panel becomes active. These are the parameters used for Savitsky-Golay smoothing the data. *Npix/2* is the half-width of the smoothing box. *Degree* is the degree of the polynomial used to locally fit the data. See *Numerical Recipes* for a more complete description of Savitsky-Golay smoothing.

The legend text is controlled by the fields *O str* (for the observation string) and *T str* for the template. The location of the legend is specified (as a fraction of the data window) by the *@X* and *@Y* fields.

To generate ascii text output of any of the graphs, hit the *Text Output* button. This will pop up another gui with suggested filenames, ending in '.sft' for **s**uper**f**it **t**ext. These may be changed by the user. Smoothed versions of the graphs are not written as text files unless the appropriate smoothed graph has been selected. In the text output files, the relative scalings are meaningful. For example, if you subtract the template from the original observation you should get the obs-gal.

One final thing to notice about sggui is the text box underneath the listing of the templates. This shows the type of the best fit SN and the number of the SNe in the top 5 matches that agrees with this type. Next to that is the epoch, which is the weighted average of the epoch of the top 5 fits. The error on the epoch is a weighted standard deviation of the epochs of the top 5 fits.

An example graph is shown below.



The blue line is the observed spectrum minus the host galaxy spectrum. The black line is the template corrected for reddening. You can change the colors of the lines by clicking the appropriate button on sggui. When a template is selected or the redraw button is pressed, if a window does not automatically pop up make sure that you do not already have an idl graph window minimized. In this case you may have to open the window and select the template again to get the window to redraw.

The final section of the supergraph gui is the postscript file selector. Supergraph automatically fills in this section with a suggestion for the postscript file name which is the rootname of the sfo file, stripped of the '.sfo' extension, appended with the rank of the selected template, and ending with '.ps'. For example, sn1981b.max.3.ps would be the third best fitting template to sn1981b.max. The user may change the name of the postscript file. Press the "Generate PS" button to create the postscript file. If a postscript file already exists with the given name it is overwritten.

Running superfit from the command line

You never have to run superfit from the command line, but it can be useful to do this for e.g. batch processing. To run superfit from the command line, at the IDL prompt type

```
IDL> sf,o='example.dat'
```

where *example.dat* is the filename of the input supernova observation. This is the only required parameter. The full list of parameters is:

keyword	description
o	Observed supernova spectrum (input file). This filename must be provided in single quotes. This should be an ascii file. The first column should be wavelength in Angstroms, and the second should be flux in f-lambda. If the input file ends in '.dat' or '.asc', when the user either completes the selection dialog or hits return, the program will automatically generate an output file name by stripping off the '.dat' or '.asc' and replacing it with '.sfo' for **s**uper**f**it **o**utput file.
disp	All spectra are rebinned to a bin size of *disp* Angstroms. The running time of the program is inversely proportional to the bin size, so a run with a bin size of 20A will execute roughly 20 times faster than a bin size of 1A. The default value for *disp* is 20, though the matching algorithm still works quite well at 50A.
zl	Lower redshift limit.
zu	Upper redshift limit
zi	Redshift increment. Superfit will begin at the lower redshift limit (*zl*), and try to fit the input spectrum with a library template spectrum shifted to that redshift. It then increases the redshift of the template by the redshift increment *zi* and tries again. It keeps increasing the redshift of the template by *zi* until the upper redshift limit, *zu*, is reached. Thus the spectra are kept in the observers frame. If none of the keywords *zl, zu*, or *zi* are present, then zero redshift is assumed. If*zl=zu*, then *zi* is ignored.
w1	Lower wavelength cutoff in Angstroms. Useful cropping spectra with low S/N at edges.
w2	Lower wavelength cutoff in Angstroms.

keyword	description
weight	Name of weighting function file, in single quotes. Available weighting functions are 'one.weight', which is unity everywhere and thus no weights (the default), 'no77.weight', which has a weight of unity across the entire spectrum, but a lower weights (effectively higher errors) around the telluric features. Alternatively, the user can supply his own weighting function, listing the filename in single quotes. User supplied weights should be in two columns -- the first is wavelength in angstroms and the second is the weight value at that wavelength. If the weighting function name is the same as input spectrum name, then it is assumed that the weighting function (error spectrum) is given in the third column. In any case weighting functions are rebinned along with the input, template, and galaxy spectra. If it is available, the sigma spectrum should be used as the weighting function. The program converts this to a variance and uses inverse variance weighting.
Rv	The reddening law: Av/E(B-V). The default is 3.1.
avmin	The minimum reddening (actually bluening if negative), in magnitudes of visual extinction, Av. The default value is -3.0. It is useful to allow bluening of the template spectrum because some may not have been dereddened. To allow no reddening correction, set *avmin* and *avmax* to zero.
avmax	Maximum allowable reddening in magnitudes of visual extinction, Av. The default value is 3.0.
tempscale	Maximum template scaling. Input spectra and template spectra are initially scaled so that their median value is 1. The fitting algorithm is then allowed to multiply the template spectrum by a scale factor of between 0 and *tempscale* to achieve the best fit. A setting of 3 (the default) allows any galaxy scaling up to a scale factor of 3 (this upper limit was chosen empirically). If *tempscale* is set to zero then the best spectrum will have no supernova light -- only host galaxy and reddening.
galscale	Maximum galaxy scaling. Input spectra and galaxy spectra are initially scaled so that their median value is 1. The fitting algorithm is then allowed to multiply the galaxy spectrum by a scale factor of between 0 and *galscale* to achieve the best fit. A setting of 3 (the default) allows any galaxy scaling up to a scale factor of 3 (this upper limit was chosen empirically). If *galscale* is set to zero then the best spectrum will have no galaxy contribution.

keyword	description
results	The name of the superfit output file. If omitted, and the input file ends in '.dat' or '.asc', the program will automatically generate an output file name by stripping off the '.dat' or '.asc' and replacing it with '.sfo' for **s**uper**f**it **o**utput file.
galnames	Array specifying which galaxy templates to use, with each galaxy in single quotes, and the entire array enclosed in brackets. To try all galaxies, use: *galname*=['E','S0,'Sa','Sb','Sc','SB1','SB2','SB3','SB4','SB5','SB6']. Any subset may be provided instead. If none are provided, galname is set to ['E'] and galscale is set to zero. Thus even though the output file will say that an E galaxy was used, the scaling will be set to zero. To provide a user-input galaxy spectrum, use *gspec*.
gspec	User-input galaxy spectrum. Must be in single quotes.
tempfile	The name of the template library idlsave file. Possible values include: allsne.idlsave, II.idlsave, Ib.idlsave, Ic.idlsave, Ia.idlsave, 2002cx.idlsave, snelt10d.idlsave, and Others.idlsave. These are described in the gui template library selector section.
sfracrequire	Flag that can be set by typing /requirefullwav in the call to sf. Superfit will only allow matches if the template covers a fraction more than sfracrequire of the wavelength range of the input spectrum. If a template does not have the required wavelength coverage, it is skipped and its goodness-of-fit value (S) is listed as 999.99 in the output file. Requiring full wavelength coverage ensures that each template is compared in a consistent way (each over the same wavelength range). However, this approach has the disadvantage that you are cutting down the total number of templates in the comparison. At high redshift this can be a problem, where you are comparing the input spectrum to the UV portion of local templates, since only a few templates have UV coverage.
sigmasource	The source for sigma if sigma clipping. Must be in single quotes. Options are 'calculate', which calculates sigma from the input spectrum, 'input', which uses a user input sigma spectrum, and 'none'.
nsigma	If sigma clipping, nsigma is the number of sigma at which clipping begins.
niter	The number of iterations of sigma clipping.
grow	Expand number of pixels clipped on either side of the main clipped pixel by this number.

An example call of superfit from the IDL command prompt, with every parameter (note that any except 'o' can be omitted to get the default behavior):

```
IDL> sf, o='/home/howell/data/gemini/jun04/04D3kr.dat',
w1=4708.64, w2=8900.00, disp=20, zl=0.337, zu=0.337, zi=0,
weight='/home/howell/data/gemini/jun04/04D3kr.dat', Rv=3.1,
avmin=-2, avmax=2, galscale=3, tempscale=3, gspec='',
galnames=['Sa','Sb','Sc'], results='/home/howell/data/gemini/
jun04/04D3kr.sfo', tempfile='savefiles/allsne.idlsave',
sfracrequire=0.8, sigmasource='calculate', niter=5, nsigma=2.3,
grow=0
```

Every time you run sfgui, the call to sf.pro (similar to the above) is saved in the file sf_command.txt. So if you want to rerun the same command, changing only one or two things you can edit the file sf_command.txt and then from the linux prompt do:

```
IDL < sf_command.txt
```

Batch files can also be created and executed in this way.

Changing template spectra

To change the template spectra in a given library, put spectra in the appropriate directory under the 'sne' directory, and edit the list of spectra in that library by editing the savefiles/ rootname.list file for the corresponding library. A list of template library rootnames can be found in the template library selector section. Important: do not leave any blank lines at the end of the file.

Note that some all SNe appear in at least two lists -- the one for their subtype (e.g. *Ia.list*) and *allsne.list*. The supernovae less than 10 days after max also appear in *snelt10d.list*.

After you have changed the list of files you must run the sfsetup program by calling it from the IDL command prompt, e.g.:

```
IDL> sfsetup
```

This will take several minutes while the libraries rebuild themselves. Sfsetup does not need to be run again until the next time you want to change libraries.

Troubleshooting

What to do when any of the programs crash

If either superfit or supergraph crashes, type:

```
IDL> close,/all

IDL> retall
```

and then restart the program. If things get really screwed up you may have to exit and reenter IDL.