

The Complete Manual on Spectral Analysis

BY: BRANDON MCQUILKIN AND DR. MICHAEL SITKO

Setting Up IDL	4
Workspace and Data Organization	6
Mike's SpexTool Manual 2.2	7
SpexTools Tools Overview	7
XSPEXSTOOL	8
Setup:	8
Calibration:	9
LXD With Cloudy Skies:	16
XCOMBSPEC	18
XTELLCOR	22
XCONKERN	24
XSCALELINES	28
Estimate Tool	30
Fix Tool	31
Splines Tool	33
More on Fix Tool	34
XFINDSHIFT	36
XTELLCOR (LXD)	41
XMERGEORDERS	46
Merging Noisy LXD Data	51
Reality Check	53
XCLEANSPEC	54
XMERGEVD	57
Handling Prism Data	61
XSPEXTOOL	61
XCOMBSPEC	64
XTELLCOR	66
XMERGEORDERS	69
XCLEANSPEC	70
XMERGEVD	72
Brandon's IDL Manual 1.0	73
Fitting with Mathematica	74
Exporting Data for Mathematica	74
Fitting	77
Plotting in IDL	84
Model Creator	85

Prism Data	87
Band Flux	88
Auto Lines	89
Accretion Rates	91
Plot(star name).pro	92
Line Comparison	93
Plot(star name).pro	95
Line Comparison Charts	96
Plot(star name).pro	96
Accretion Rate Charts	98
Appendix	100
SpexTool Cursor Commands	100
Line Identifications	101

Setting Up IDL

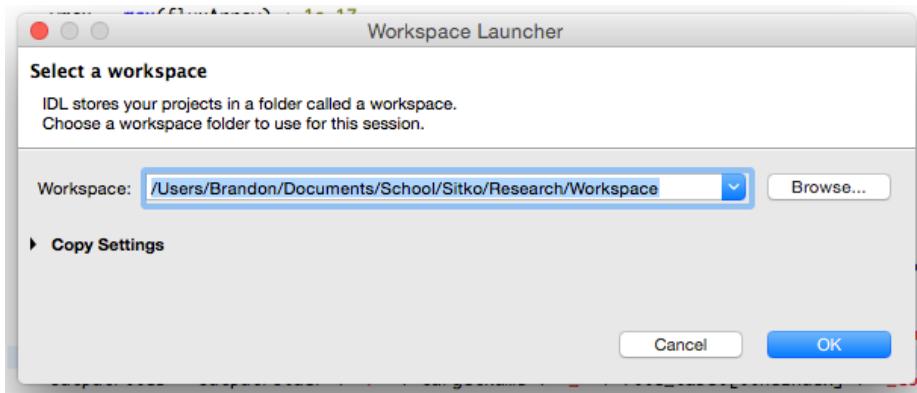
This section will go through the steps necessary to setup IDL for analysis.

1. Install IDL

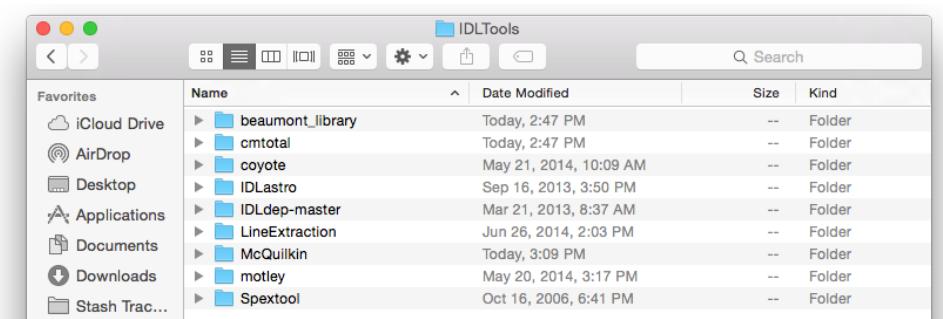
- A. Download and Install IDL through Exelisvis. Follow the instructions for your system.
 - a. **License Installation Instructions:** <http://www.exelisvis.com/Support/HelpArticlesDetail/TabId/219/ArtMID/900/ArticleID/4660/4660.aspx>
 - b. **Obtaining the Host Name and ID:** <http://www.exelisvis.com/Support/HelpArticlesDetail/TabId/219/ArtMID/900/ArticleID/5321/5321.aspx>

2. Setup your workspace.

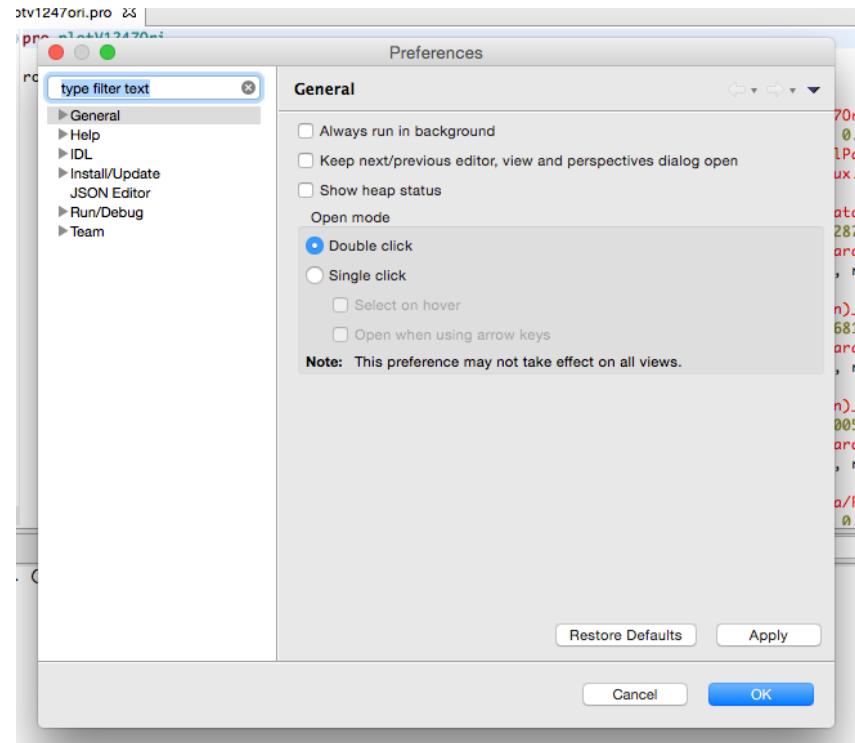
- A. Move your default workspace folder to a more convenient location. The default location is usually not one easily accessible.
 - a. Create a folder in a convenient location (Usually your documents folder.) for your research stuff. Inside that folder create a folder called “Workspace”.
 - b. Open IDL
 - c. Select File → Switch Workspace → Other, and a new window should open.



- d. Browse to the new workspace folder that you created and hit “OK”. This will move IDL’s workspace to the new location. You can go ahead and delete the old location if you wish.
3. Add the necessary tools to IDL. To run all the code the following libraries will need to be installed: SpexTool, IDLastro Library, Coyote Library (This one should be included in the IDLastro Library), and the McQuilkin Astro Library.
 - A. Setup the libraries
 - a. Create a folder next to the “Workspace” folder in the previous step. Call it “IDLTools”.
 - b. Copy all the libraries into this folder. The top level structure should now look something like this.
(There are more libraries installed on my system than what you will have.)
 - B. Add the necessary paths to IDL so that it can find the installed libraries.

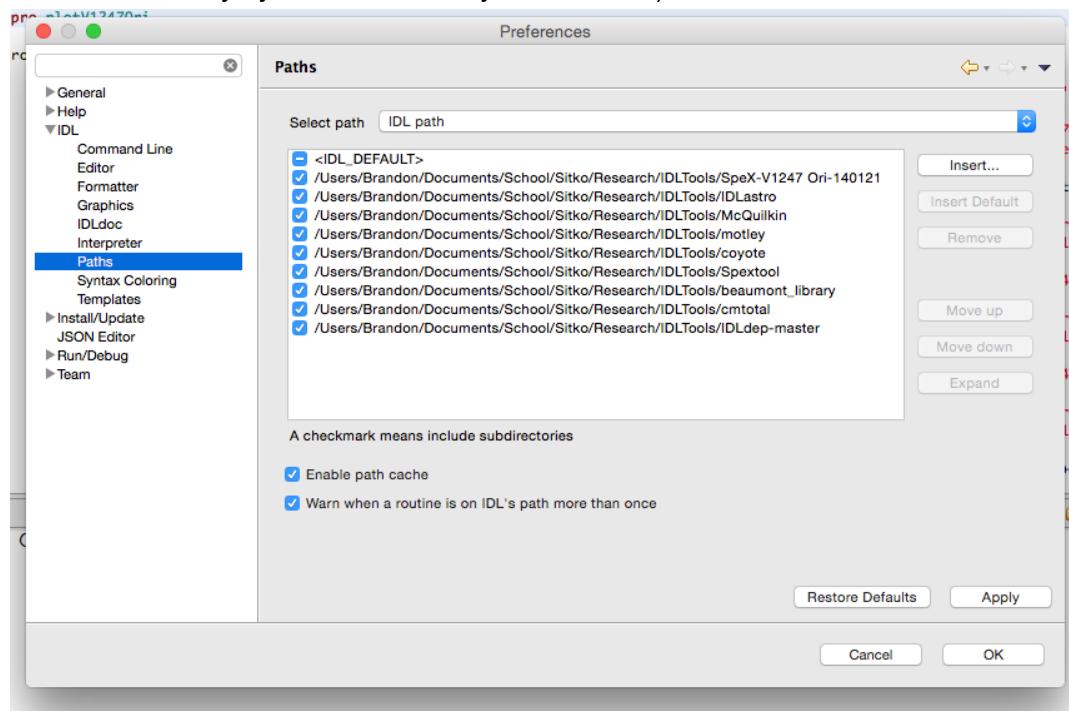


- a. Go to IDL and open its preferences window. You should see something like this:



- b. Then select IDL → Paths

- c. Press the “Insert” button, and browse to each folder in the “IDLTools” folder and add it individually. Then click the check box to the left of each path in the table to activate the library. Your screen should end up looking something like this: (Again note that I have more libraries installed on my system than what you will have.)



- d. Click “Apply” and close the preferences window.

Your IDL setup is now complete, you should now be able to process spectra.

Workspace and Data Organization

It is very important that you keep the data very organized. It is very easy to loose data, and mix data up without a good naming system. It took me much trial and error to figure out this system, and this system will specifically work with the data extraction programs.

Folder Organization Structure:

- *Workspace* : The workspace you setup earlier.
 - *Star A* : Create a folder for each star. Put any files on the star (like papers) in this folder.
 - *SpexTool* : A folder containing all the raw data that SpexTool needs, and the data for the intermediate steps.
 - *RawData* : The folder containing the data directly from the telescope. All the arc###, flat###, and wavecal### files should go here. They are usually files with some prefix and a three or four digit index number.
 - *xSpexToolOut* : The folder containing the files output by xspextool.
 - *Data* : A folder containing the unaltered data. Usually fits files from spextool, and possibly sav files. This is the final data output by SpexTool.
 - *Data Files* : The files in this folder should be named in the following fashion:
starname_date(_prism).fits (date is the format of: YYMMDD)
Example: V1247Ori_100228.fits or V1247Ori_100228_prism.fits
 - *Processed* : The folder containing the processed data, sorted by data file.
 - *Data Folders*: A folder for each file set in the “Data” folder. The folder should be named with the same name as the file. (A file set is the SXD, LXD, and Prism data for a specific star collected on the same collection run.)
 - *Lines* : A folder for all the line extraction data and plots to export to.
 - *Model* : A folder for all the model data and plots to export to.
 - *StarFitter.nb* : The fitting file for that particular data file. (To be explained later.)
 - *.fits.csv : The CSV version of the fits file for Mathematica to import. (Since Mathematica cannot read .fits files.)
 - *Comparison* : A folder for all the line comparison plots and data.
 - *Parameters.txt* : A place to store all the parameters obtained by all the Mathematica fits for easy access.
 - *plot(starname).pro* : A file that runs all the necessary commands, in order, to create all the plots and extracted data in one go. It makes it much easier to add and remove data in the future.
 - *Star B* : ...

Generally you will copy data into one of two places. If you have data that needs to be run through SpexTool, place it in the “RawData” folder. If you have data that has already been run through SpexTool, and needs to be plotted, place it in the “Data” folder.

Mike's SpexTool Manual 2.2

SpexTools Tools Overview

First of all, this is NOT meant to replace the Spextool help files that you can access using the Menu button at the bottom of each module's IDL window. (You can also read these text files directly: look in Spextool/helpfiles, they are plain text). My intention was to provide an annotated visual record of what a Spextool session is like, and also describe some things I have come to learn about the details of the process. Here I summarize each module and what it does, in the order that they are to be done.

Xspextool – Here you construct your calibration frames (master flat and wavelength calibration), determine extraction apertures, extract the data (after background subtraction and flat-fielding), and merge the A and B beam observations. The result is a number of spectra representing each original observation file, properly calibrated. SXD and LXD are done separately.

Xcombspec – Here you merge the various observations together, SXD and LXD dome separately.

Xtellcor - (and **xtellcor_finish**) Generally the most time-consuming part of the process, at least for SXD (again SXD and LXD are done separately). Here you use your observations of an A0V star and telluric model absorption spectrum to determine the telluric corrections for your target star data. It is based on doing a flux calibration of your A0V star based on Vega. This requires determining how to alter the Vega spectrum to match the H line shapes and strengths of your A0V star, and apply it to the entire spectrum. This is then used to “remove” the H lines from your A0V star data so that it should resemble the telluric data. Since no two A0V stars are exactly identical (and Vega is peculiar, being a rapidly rotating star viewed pole-on) you need to go in and “tweak” them into submission. You can then construct the telluric correction data and apply it to your science target spectrum. You then adjust the wavelength scales of the telluric spectrum and your science target data by aligning their telluric features. The divine the two to get a (hopefully) telluric-corrected spectrum. If you have an A0V calibration star that is a good match in airmass & telescope position for more than one science target, **xtellcor_finish** allows you to skip most of the first part, which would simply be repeating what you have done, and go right to the spectra alignment.

Xmergeorders – This allows you to merge the various orders together within SXD and LXD separately.

Xcleanspec – For trimming out data that is noisy, or where you want to remove wavelength regions with systematic distortions due to poor matching of airmass between the A0V star and the science targets.

Xmergexd – Here you can merge SXD and LXD together.

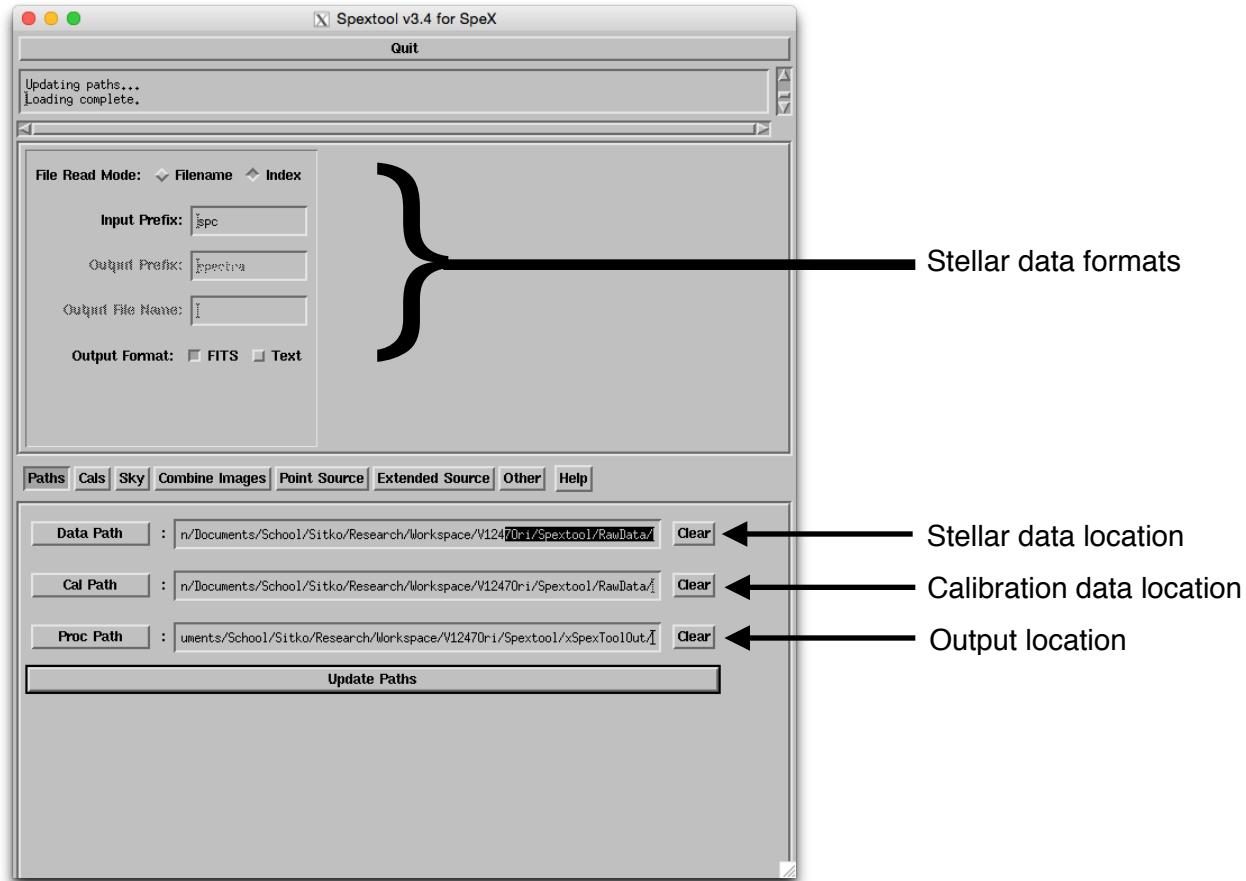
At any stage of the process you can examining the spectra using **xvspec**. Note that at each stage of file writing, information of what you just did is added onto the FITS file header. I find it useful to look at these using a suitable FITS viewer, like the SAOIMAGE DS9 utility. Note that **Xcleanspec** can be used after **Xmergexd** if you need to go back and do some more cleaning.

For more info, see: Vacca et al. 2003, PASP, 115,389 – Telluric Corrections and Cushing et al. 2004, PASP, 116, 362 – Xspextool Package

XSPECSTOOL

Setup:

Open IDL, and in the console at the bottom of the screen type: “xspecstool” (without the quotes.) And you should see the following window:



Make sure the “Paths” tab is selected. Then input the folder paths for the stellar data, calibration data, and data output by clicking the button to the left of the field. Once that is done make sure you click “Update Paths” so that *xspecstool* will remember these paths.

For the stellar data formats, we will generally load indexed files, (so select index.) Then type the prefix on all stellar data files. (The prefix is the part before the file index number. *prefix001.fits*, *prefix002.fits*...) The prefix is usually the star name.

Note: There will usually be two sets of data for a star: SXD and LXD. It is important to keep these straight, and separate. The SXD is shorter wavelength data, and LXD is longer wavelength data. The two sets will be merged later on, but for now must remain separate.

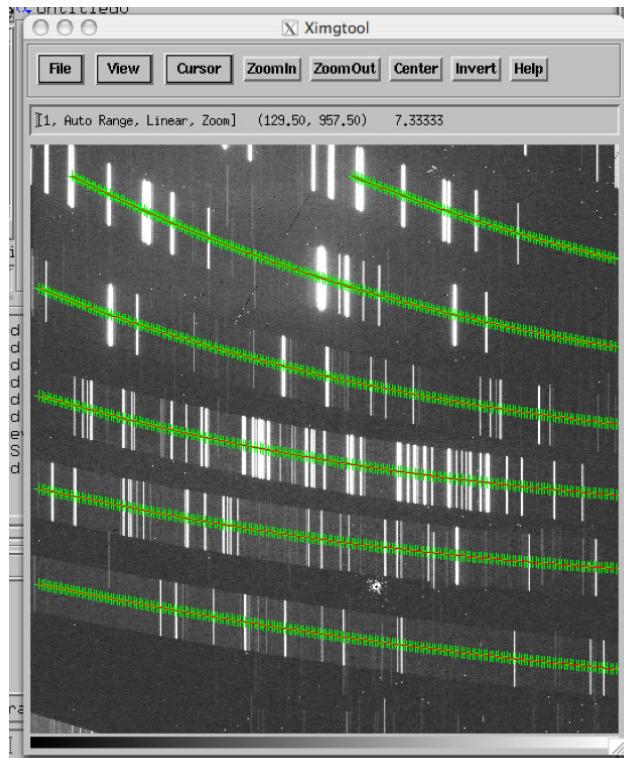
Calibration:

Now we will load the calibration macros (arc and flat files) and create wavecal files. So click on the “Cals” tab.



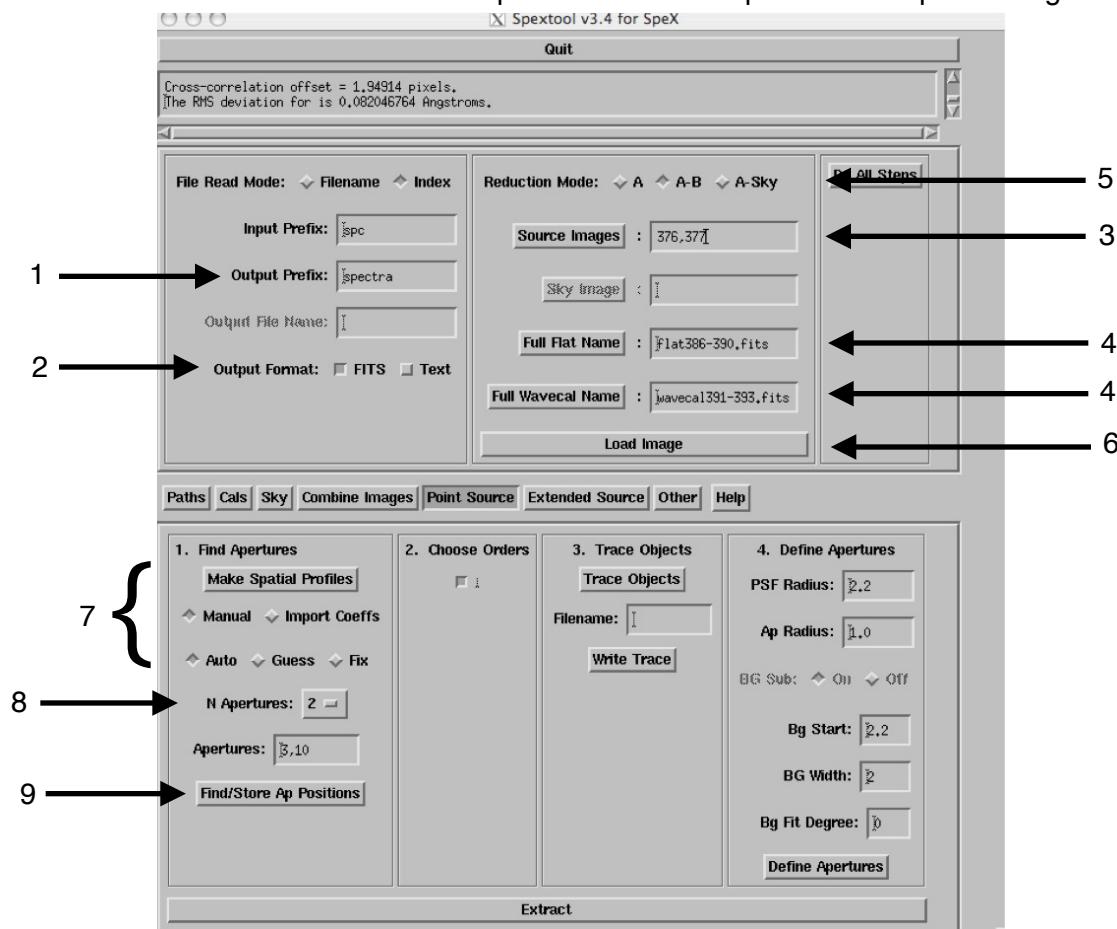
In the table, enter the range of indices of the files to load. To do that, click the cell that you want to enter the data into, type the range, **and press ↵**. Make sure to press ↵, this will not work if you don't. Arc and flat files should be entered in the first column of the table. If you are processing LXD data, a pair of AB sky files must be specified as well in the second column, so that the telluric emission lines can be added for the longer wavelengths.

Once you are finished entering, press “Construct Calibration Frames”. A new Ximgtool window should appear:

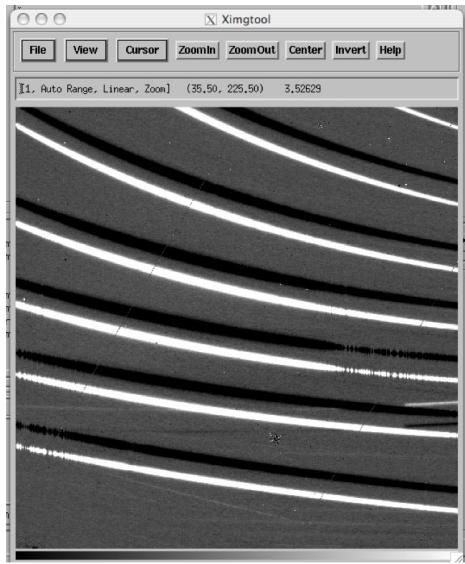


After all the arc and flat files are processed, the window should look like this. You should also have a merged arc file, a merged flats file, and a merged wavecals file in the folder you specified as the data path. (The file endings will be a range of indices, like the ranged entered in the "Cals Files" column).

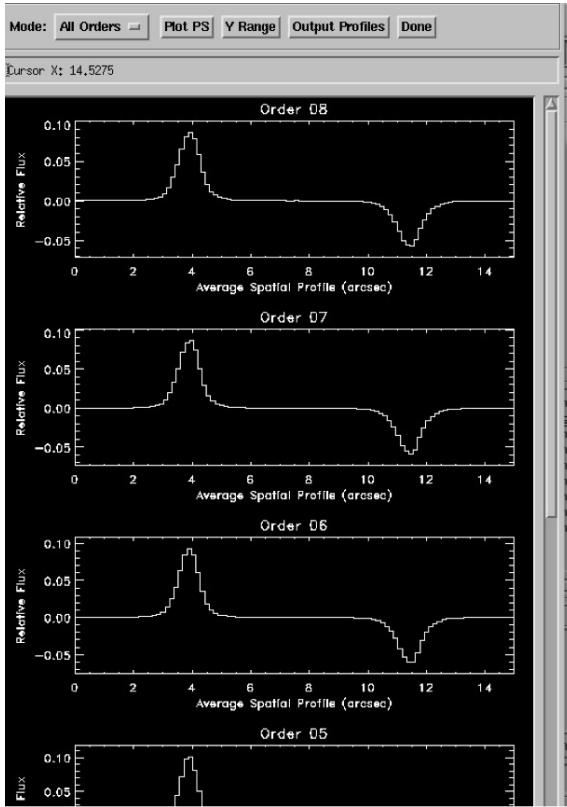
Now click on the "Point Source" tab on the xSpexTool window to proceed with processing the stellar sources.



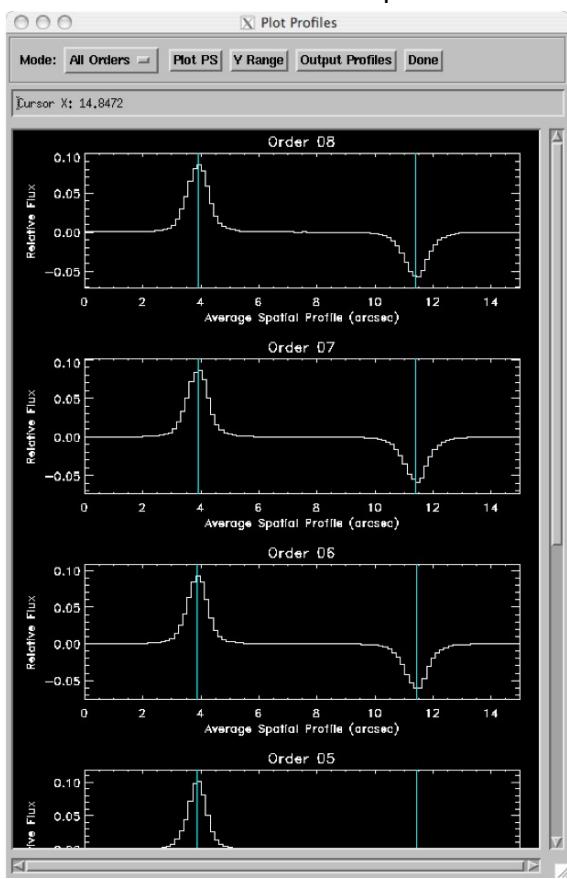
1. Enter your output prefix. I would make the prefix the star name, then whether it is SXD or LXD data, then “spectra”.
 - Like “V1247Ori_SXD_spectra” or “V1247Ori_LXD_spectra”.
2. Make sure the Output format is “FITS”
3. Select the first two stellar files. Generally the A and B beam files. (These will be the first two files in a long sequence of files.)
4. Click the “Full Flat Name” and “Full Wavecal Name” buttons to load the last files that xSpeXTool created. Otherwise fill in the boxes appropriately.
5. For A-B beam subtraction, click the “A-B” Box.
6. Press the “Load Image” Button to load the data. A new window should open:



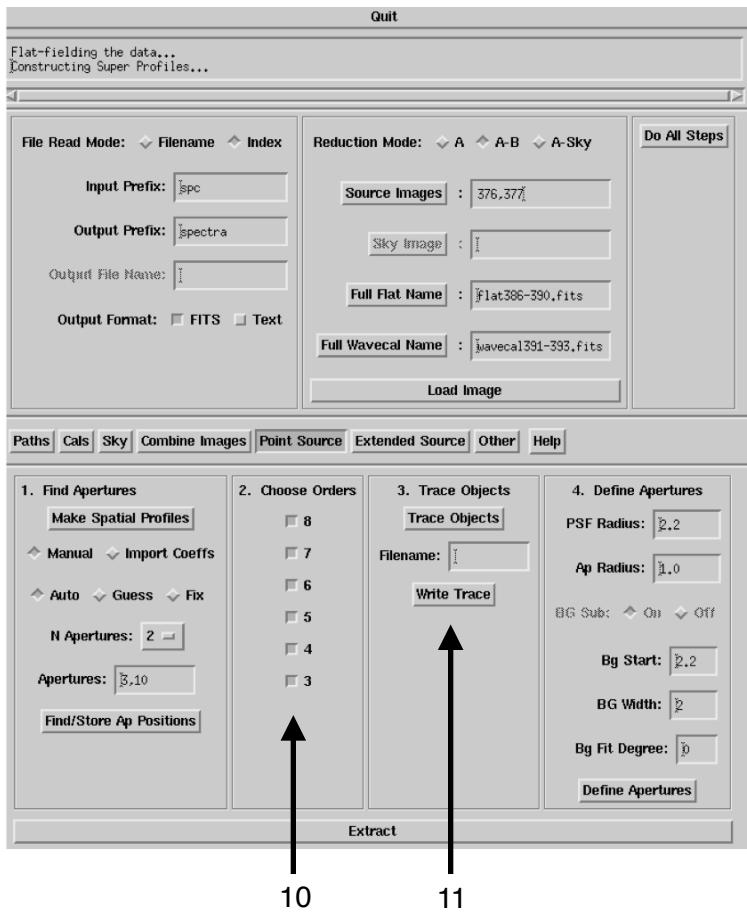
7. Next we will find the apertures. For finding the apertures, we will generally use Manual (manually determined from the data) and Auto (to automatically fit the actual observations.) Press “Make Spatial Profiles”. A new window will appear.



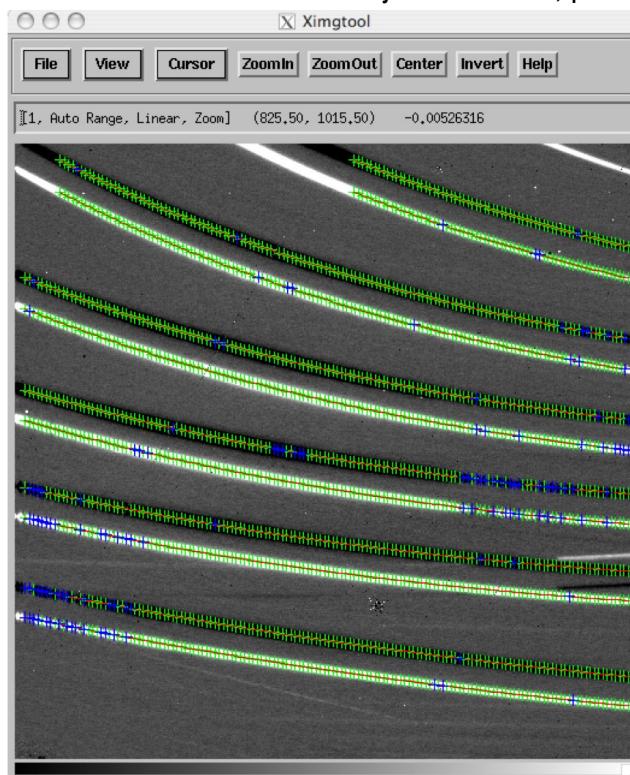
8. Set the number of apertures to “2” for two-beam (A-B) data.
9. Press “Find/Store Ap Positions” to find the centers of all the apertures.



SpexTool will mark the centers of the apertures over all the spectral orders. The centers of the A and B beams are indicated by the cyan lines.

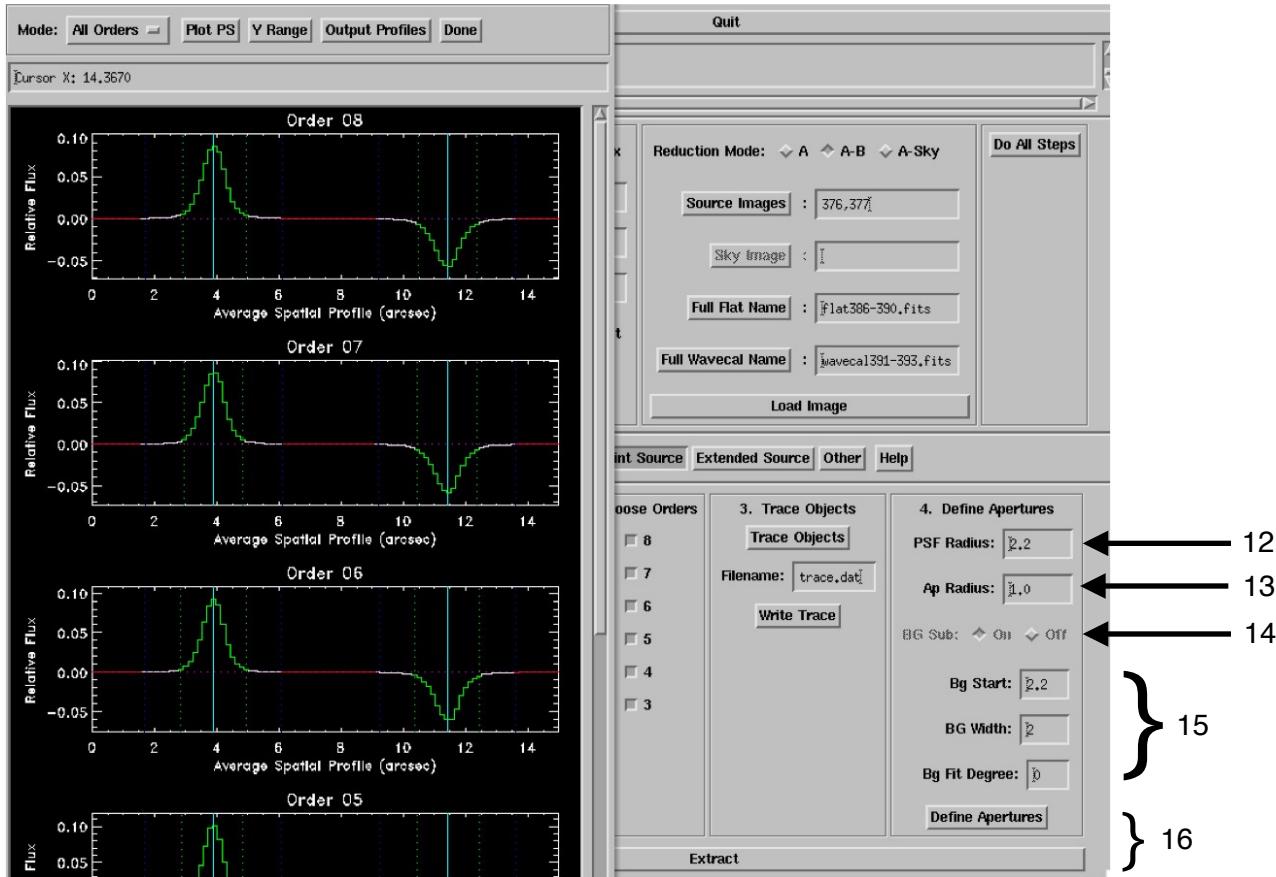


10. Select the spectral order to be processed (usually all of them) by selecting all the boxes in the “Choose Orders” section. Then in the “Trace Objects” section, press “Trace Objects”.



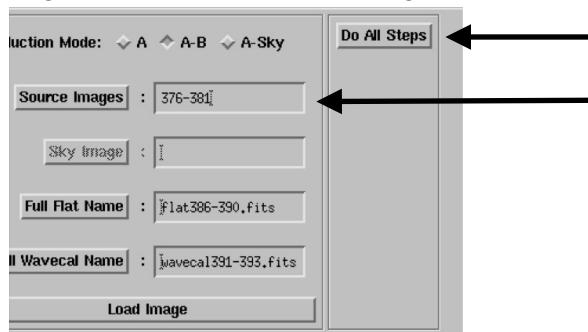
SpexTool will trace the peaks of the spectral orders for the A and B beams.

11. In the “Filename” text box type in the name of the trace file to save. I suggest the star name, sxd or lxd and trace. (V1247Ori_SXD_Trace or V1247Ori_LXD_Trace.) Then press “Write Trace” to save it to the “RawData” folder.



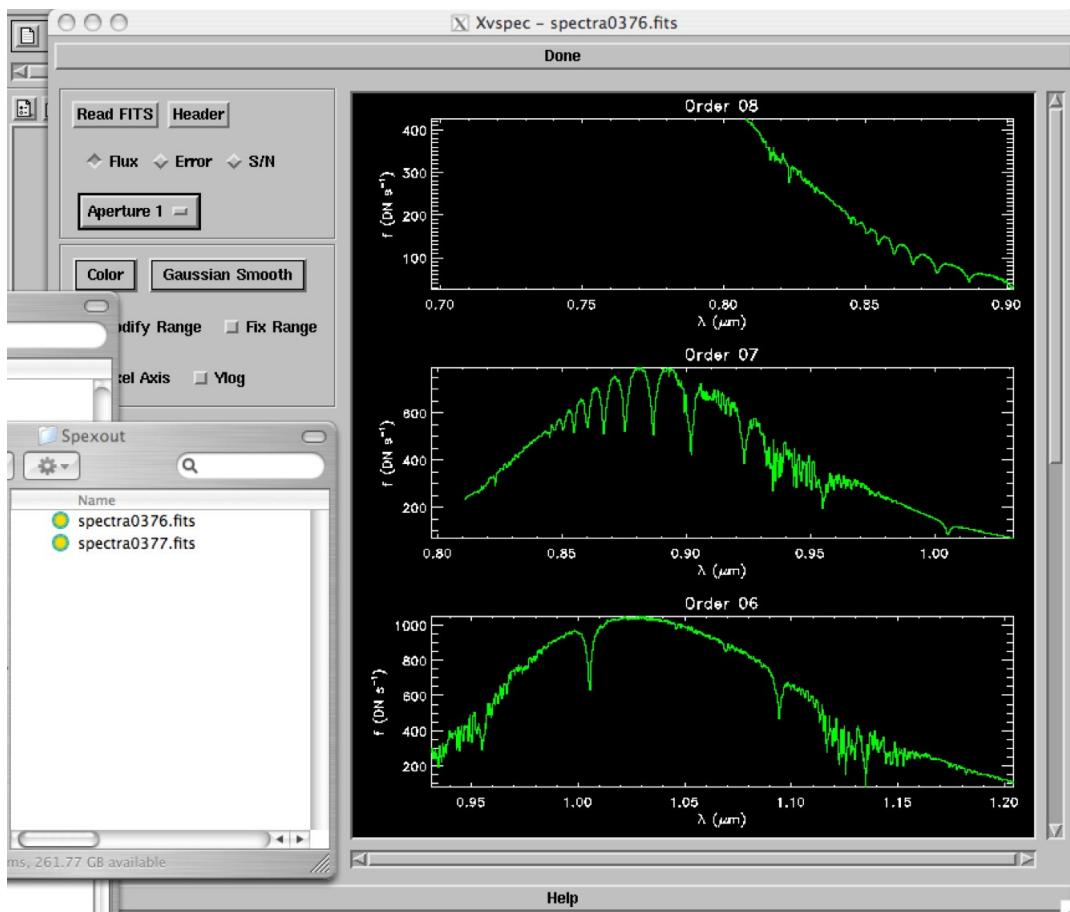
12. Define the PSF Radius in arc seconds. Usually about 2.0-2.5 for the IRTF.
 13. Define the Aperture Radius, which must me less than or equal to the PSF Radius.
 14. For most of the data, a background subtraction is necessary. (The background is outside of the A and B apertures.)
 15. Define what data is the background. Start the background outside of the PSF Aperture, and give it a width wide enough to get a lot of data, but not wide enough to encroach on the other aperture. You must pick a polynomial order number for the fitting, which will usually ve zero (flat.)
 16. Press “Define Apertures”. This will place the background subtraction locations on the plot. If these are ok, Hit Extract. If not, go back and modify your parameters.

We are almost done with the first phase of the extraction process. Once you are satisfied with what you did on these two data files (The A and B beams of one set), do the same to the rest of the data on that target with that grating. To do that in the “Source Images” field, put the whole range of file number in, and press “Do All Steps”.

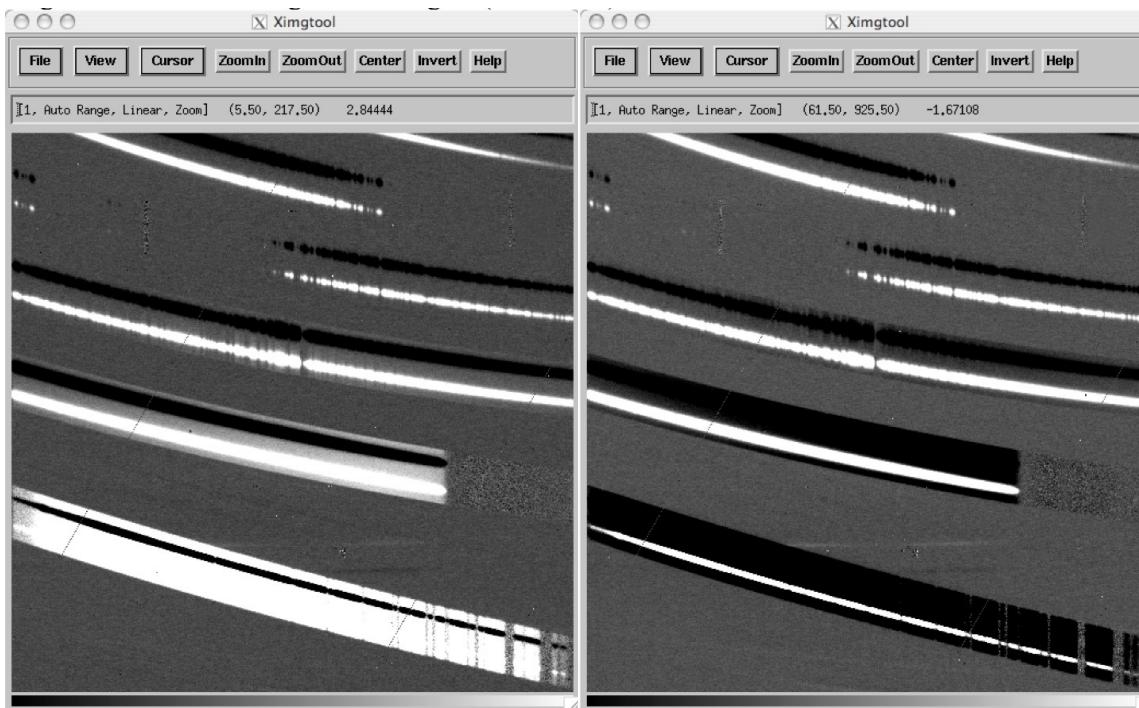


This will extract the rest of the data, and write it to the output directory defined earlier. And show the background subtracted A-B data merged.

LXD With Cloudy Skies:

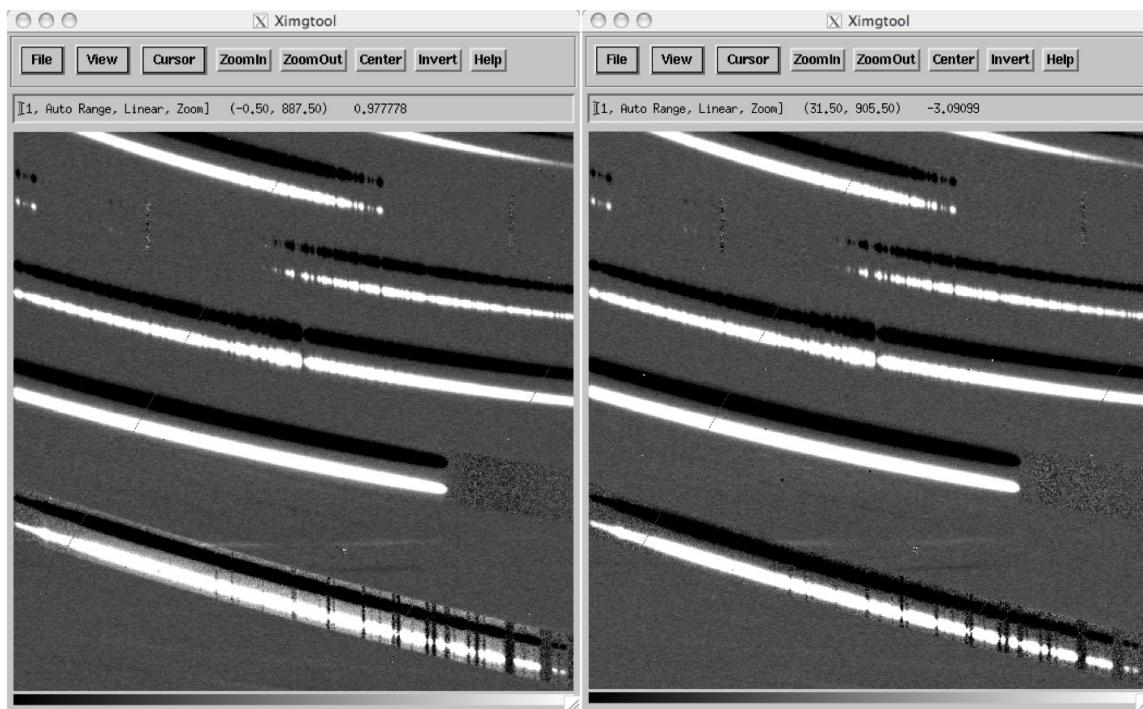


Sometimes you will have spectra taken through thin clouds (cirrus). These not only cause the sky transmission to be low and variable, but cause the background to be high and variable at longer wavelengths (lower orders) in LXD.



The clouds might be more prevalent in either the a-beam (left) or the b-beam (right).

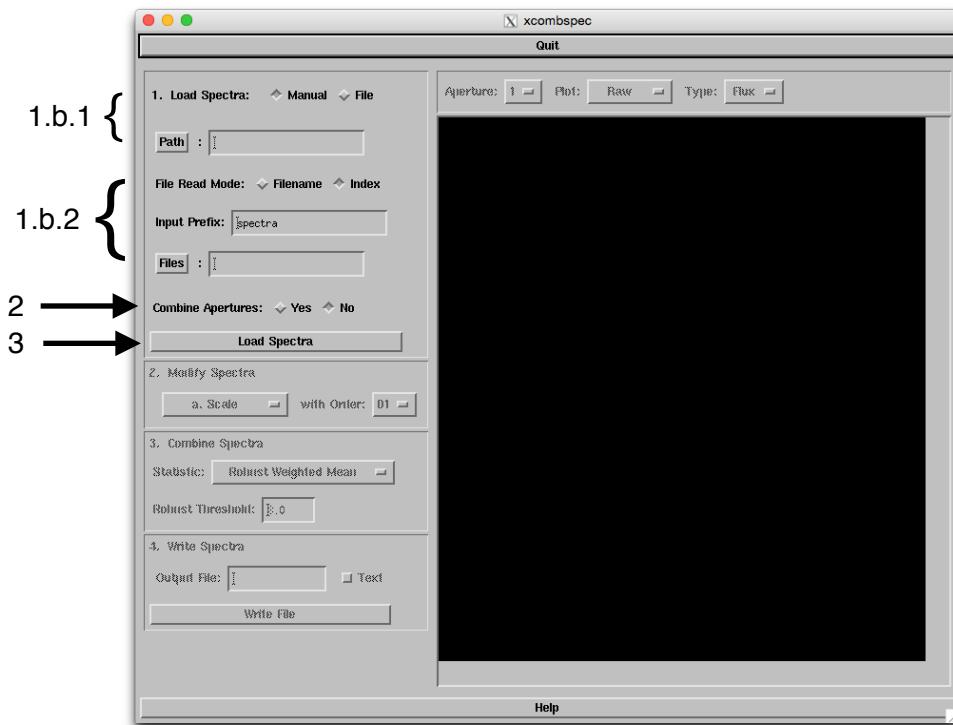
Sometimes there is only a little bit more cloud in one beam than another (left), while the next set might be essentially cloud-free (right).



Here I had to “hand-pick” individual pairs by examining them using the “Source Images” and “Load Image” buttons in xspectool prior to making the spatial profiles.

XCOMBSPEC

The next phase is to merge the spectra you have just extracted for a specific target stars and grating setting. To do this, go back into the IDL window, type in “xcombspec” and hit ↵.



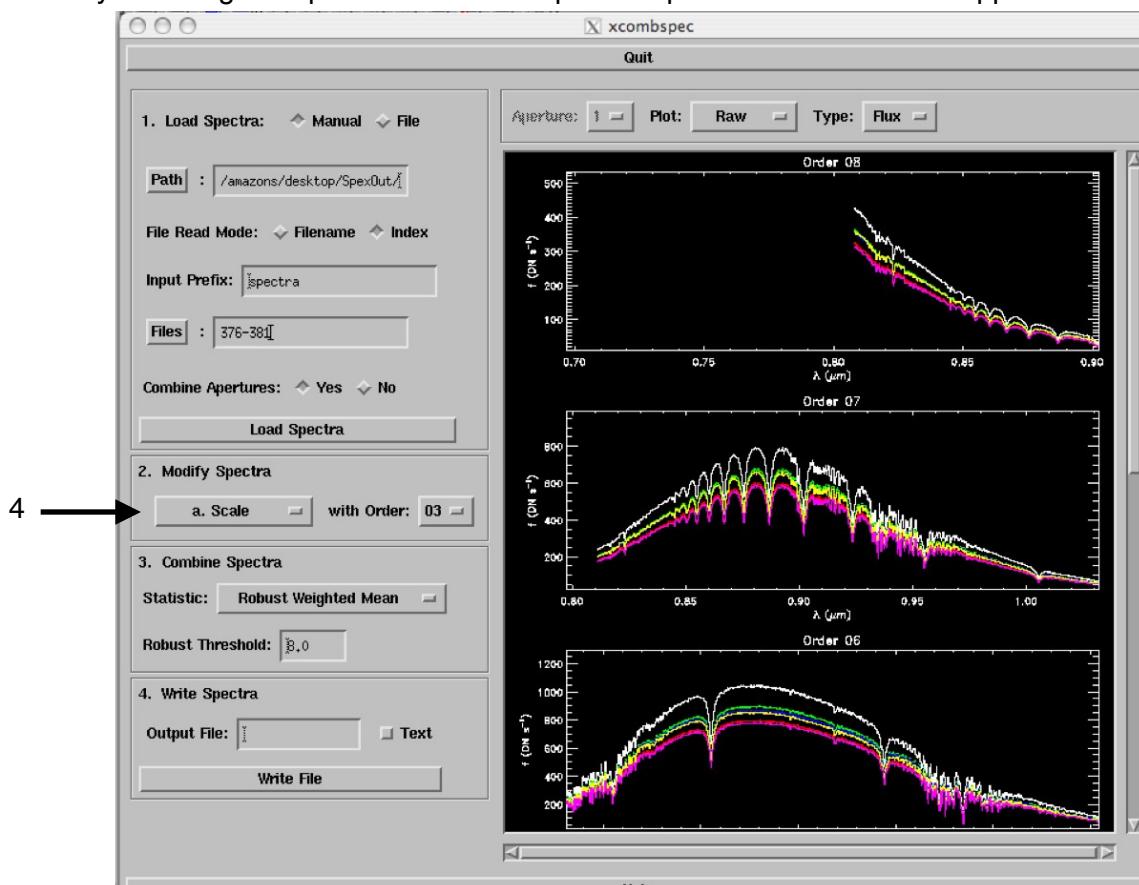
The following is the help file from xcombspec which can be viewed by pressing the help button. It is retyped here with a few additional notes.

1. Choose the input mode for the file names.
 - a. If the user wishes to combine several spectral files located in various directories, s/he should create a separate text file listing the full path names and filenames of these spectra, and then choose the "File" input mode. The filename of the text file can then be input into the "File Name" field, or selected from the list presented when the "File Name" button is clicked.
 - b. If "Manual" mode (the default) is chosen, the user must enter the directory path where the data files reside in the "Path" field, or click on the "Path" button and choose the path from the list shown. (If xcombspec is started up in the directory containing the data files, no Path name is required.)

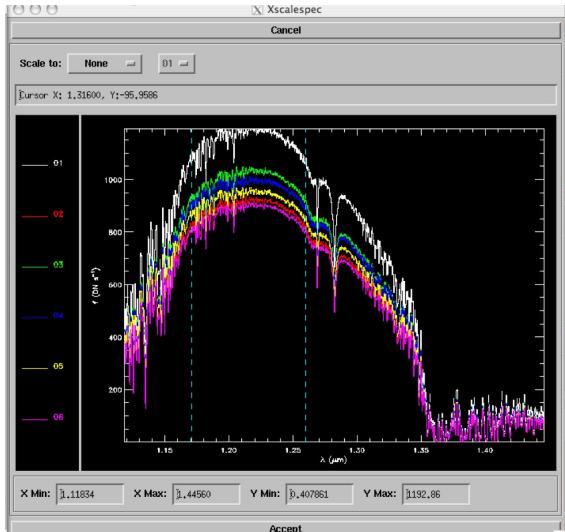
The user must then specify the manner in which the files are to be read into xcombspec. If the files to be combined all have the same prefix, but different numbers (e.g., rspec10.fits, rspec11.fits, rspec12.fits, etc.), then the user should choose "Index" mode (the default). The input prefix must be typed into the "Input Prefix" field (e.g. rspec), and the file numbers entered into the "Files" field (e.g., 10-20).

If the files have different prefixes, the user should choose the "Filename" mode, and specify the file names in the "Files" field. Alternatively, the files can be selected by clicking on the "Files" button and highlighting the file names from the list that is displayed.

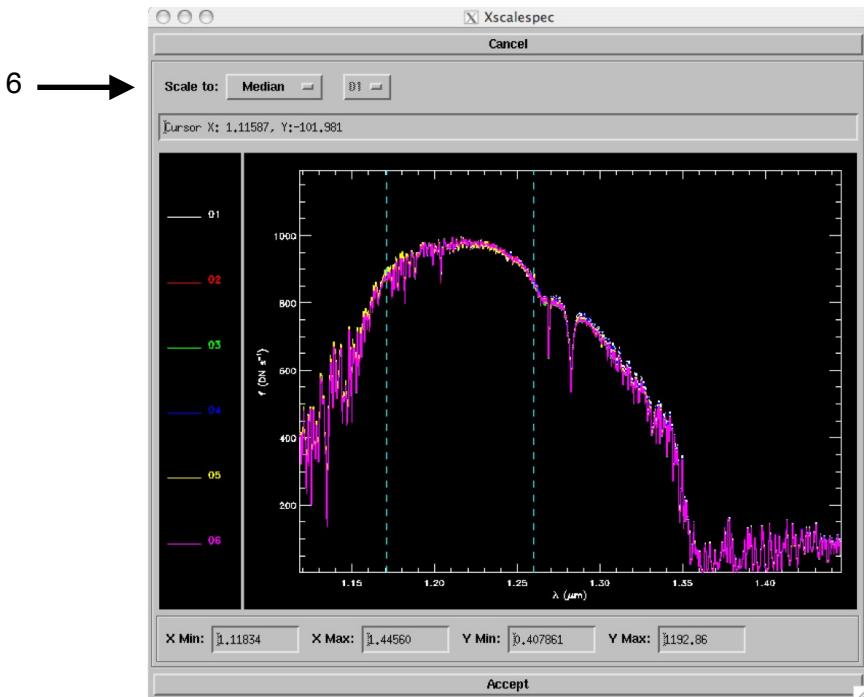
- If the input files contain spectra from multiple apertures that are to be combined, the user should select the "Yes" button next to the "Combine Apertures" label. Multiple aperture data in a single spectral file would be generated if, for example, a faint point source were observed in A-B mode and the individual two-dimensional SpeX data image files were combined before the spectra were extracted, or if multiple sources were positioned on the SpeX slit and spectra of each object were extracted from the individual files. (In the latter case, however, the user would generally not want to combine the aperture spectra.) If the data files consist of observations of a single point source, observed in AB mode and extracted individually, the input files will not contain multiple apertures.
- Click on the "Load Spectra" button to load the files. The spectra will then be displayed in the display panel on the right hand side. If the data files contain multiple orders, all of the orders will be displayed. The user has the choice of displaying the fluxes, the errors, or the S/N ratios as a function of wavelength, by choosing from the "Type" pull down menu in the upper right hand corner of the panel. The different apertures (if available) can also be viewed by selecting the aperture from the "Aperture" pull down menu in the upper left hand corner.



- Under "Modify Spectra", select "Scale", then select an order, usually one in the middle. (For SXD usually 6, and LXD usually 5.)

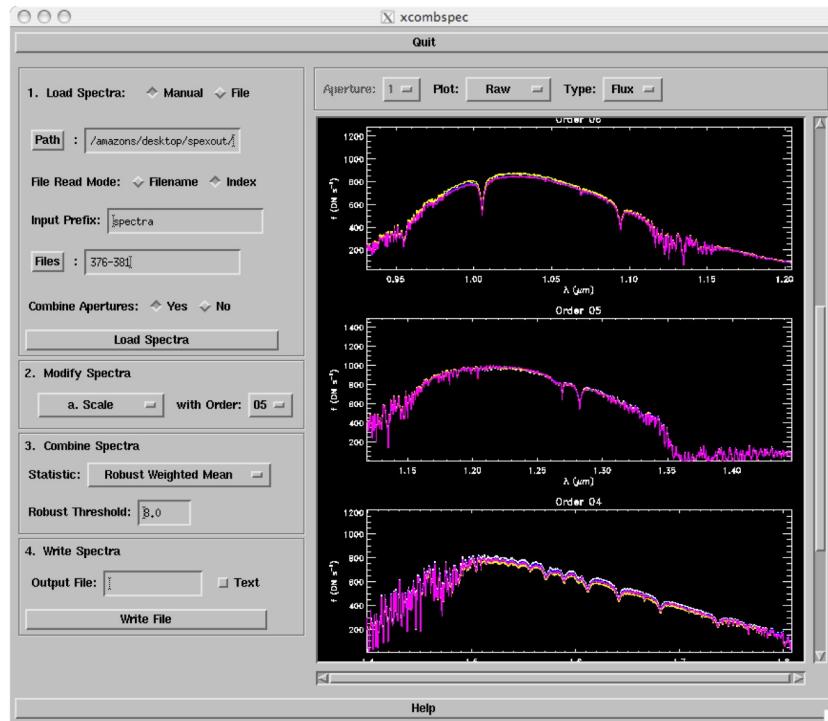


5. The user should choose a wavelength range to be used to determine the scaling factors. Again, it is usually best to select the range with the highest count rate. Type "s" and click with the left most mouse button to specify the short wavelength limit of the selected range. Click again with the left most button to specify the long wavelength limit. Dashed blue vertical lines will show the designated wavelength range. The user can also zoom in on any region with the "z" command or use any of the cursor commands. (See the Appendix for more information.)

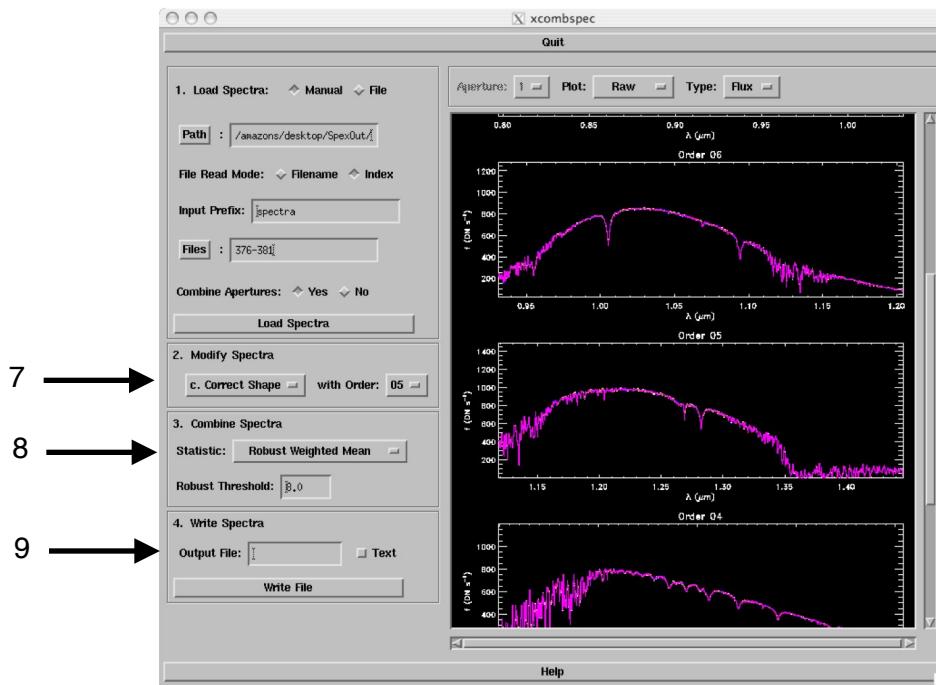


6. Choose the type of scaling to be done from the pull down menu next to the "Scale to:" label in the upper left of the plot window. The user can choose no scaling ("None"), scale all spectra to the median of the plotted spectra ("Median") over the selected wavelength range, or scale to the level of a particular reference spectrum ("Spectrum") over the selected wavelength range. If "Spectrum" is selected, the number of the desired reference spectrum should be chosen from the pull down menu next to the "Spectrum" button. Xcombspec will then determine the scale factors to be applied to each spectrum, and will scale them appropriately. The user can inspect the result with the zoom commands, change the axis limits of the plot, re-select new reference spectra or scaling wavelength ranges, etc. If "None" is chosen at any point in this process, the original unscaled spectra will be re-plotted. This will apply the same scaling to all orders. Once satisfied, press "Accept".

Note: If you want to try to get close to the real absolute flux level, you can use Spectrum instead of Median. For LXD only Order 5 is clean enough.

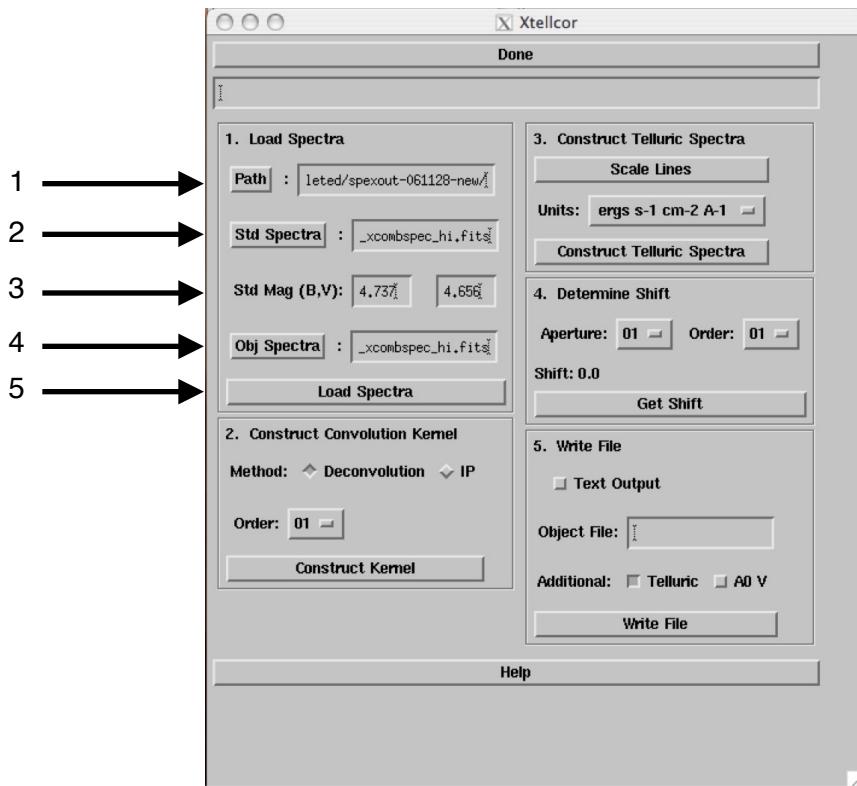


Often orders far from the chosen order will be off in scale.



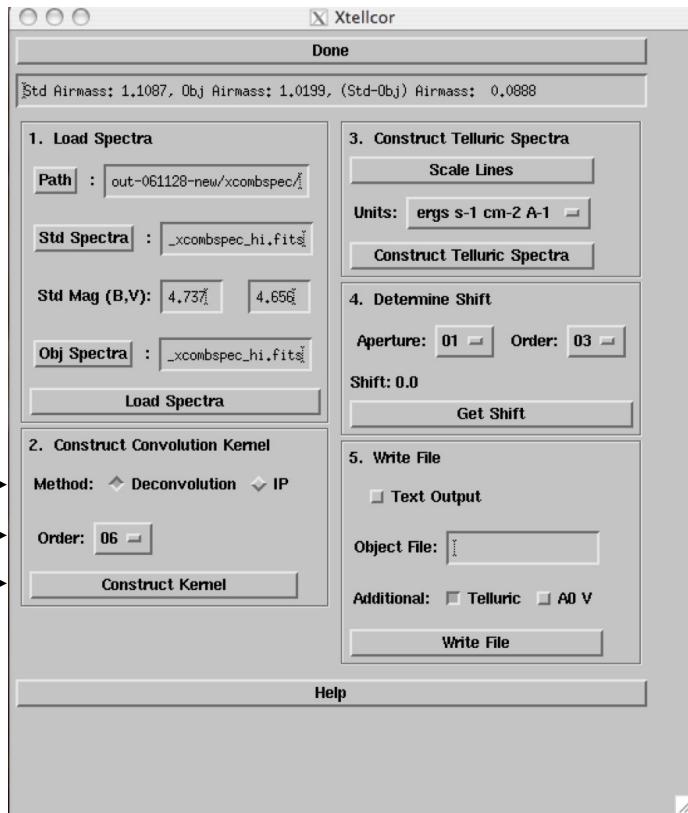
7. To fix that, choose “Correct Shape”. This will make the spectra within each order match better.
8. Choose a method for doing the statistics of the spectrum merger. Usually Robust Weighted Mean is preferred.
9. Finally, Chose a file name for the final result, and press “Write File”. It will write your file to the output folder. The suggested file name is the star name, SXD or LXD, then “xcombspec”.

XTELLCOR



1. First you need to set your paths, indicate you A0 STD star input files, the B & V mags of the A0 STD star (from SIMBAD), and the science target star's files:
 - A. If the user is not working in the directory where the data is stored, enter the path either by clicking on the Path button and selecting the directory, or by typing the path into the field. IF THE USER IS IN THE DIRECTORY WHERE THE DATA IS LOCATED, THEN THIS FIELD CAN BE LEFT BLANK.
2. Load the standard star spectra by giving the filename in the "Std Spectra:" field, either by typing the filename or selecting it from the pop-up menu after clicking on the "Std Spectra" button. NOTE: THE STANDARD STAR FILE MUST CONTAIN ONLY A SINGLE APERTURE. IF THE STANDARD STAR WAS EXTRACTED FROM A SET OF COMBINED IMAGES, THEN THE TWO APERTURES MUST FIRST BE COMBINED USING XCOMBSPEC (SELECT COMBINE APERTURES IN BOX 1.).
3. Give the B and V magnitude of the standard star. If the magnitudes are unknown, the user can enter the same magnitude (any number) for B and V. In this case, the pseudo flux calibration will not be correct.
4. Load the object spectra by giving the appropriate filename in the "Obj. Spectra:" field. This should be the file from xcombspec.
5. Click the Load Spectra button. Note that the Airmass values for both spectra, and the airmass difference, will appear at the top of the Xtellcor panel.

The instrumental profile then needs to be constructed. For SXD, this should be done via “Deconvolution”, since you can use actual stellar lines in regions free of telluric contamination to do this. For LXD, you are not so fortunate (although it is claimed that it can be done if the data is VERY high SNR). For LXD, choose IP. We will look at LXD later. Right now, we will concentrate on SXD.



6. If the user is building the convolution kernel from the spectra of the observed A0V standard star, s/he should select "Deconvolution". Then choose an order from the pull down menu in section 2 of the Xtellcor panel. The selected order should contain a H absorption line in the spectrum of the A0V standard star that is *unaffected* by telluric absorption and has high S/N. The Deconvolution method is recommended for reducing SXD spectra, and LXD spectra whenever the A0V standard star spectrum has high S/N. For SXD, Pa delta (1.005 microns) seems to be the best line to use, and it is located on Order 6. For LXD, Br (4.052 μ m) is in Order 5, while Br (2.166 μ m) is in order 9. But Order 9 will NOT be accessible in LXD_2.3 mode! Note also that LXD is so cluttered with telluric lines that you may be better off using the IP ("instrumental profile") mode.
7. Click on the "Construct Kernel" button. The Xconkern panel will appear and the spectrum of the A0 V star in the chosen order will be plotted in the upper half.

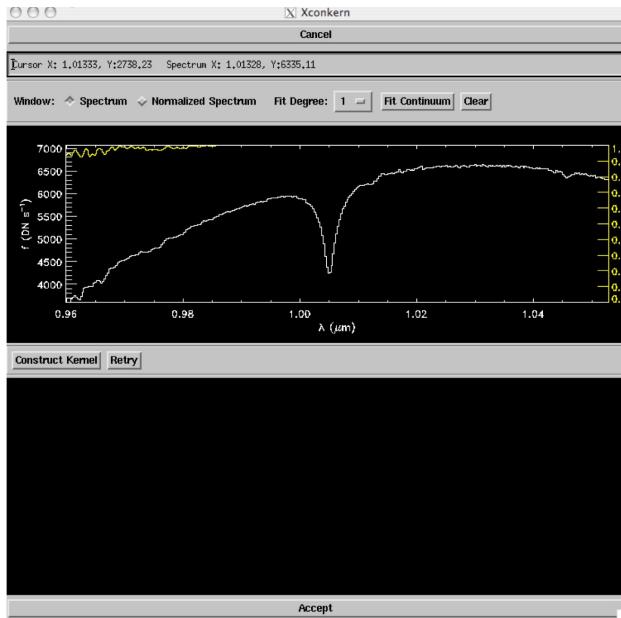
XCONKERN



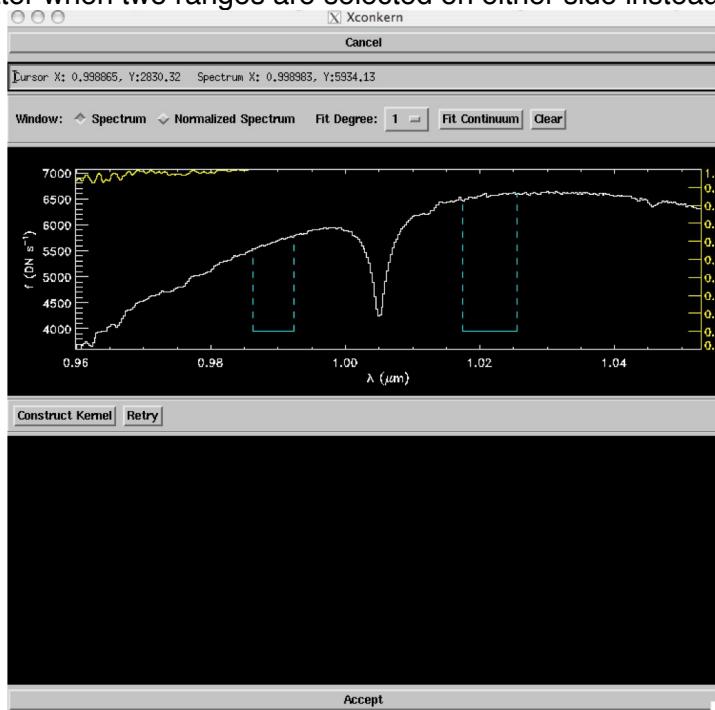
8. **From the help manual:** The user now must normalize the stellar spectrum by selecting "continuum" regions from the data and fitting them with a polynomial. Select the "Spectrum" button for the choice of "Window" at the top of the Xconkern panel. Typing "n" will put the user into the "normalization mode". Spectral regions representing continuum that are to be fitted by the program can then be chosen by clicking twice with the left-most mouse button, once for the short wavelength limit of the region and once for the long wavelength limit. The region thus selected will be designated with dashed blue vertical lines. The boundaries of a region can be shifted in wavelength by typing "m" (for "modify") and dragging the blue line leftward or rightward by clicking on the left-most mouse button and keeping it held down while moving the mouse. The user can continue in this manner to select any number of normalization regions. If the user wishes to erase all regions and start the selection process over again, they can do so by clicking on the "Clear" button at the top right of the window.

Successful kernels can be generated by normalizing either the entire order spectrum (as shown in the PASP article) or by expanding the region around the chosen H line (with the "z" cursor command - see the list of cursor commands in the appendix) and normalizing the continuum only around that line.

More simply: Select "Spectrum". Then press "z" to activate the zoom tool, and zoom around the Pa delta line. You should see something like this:

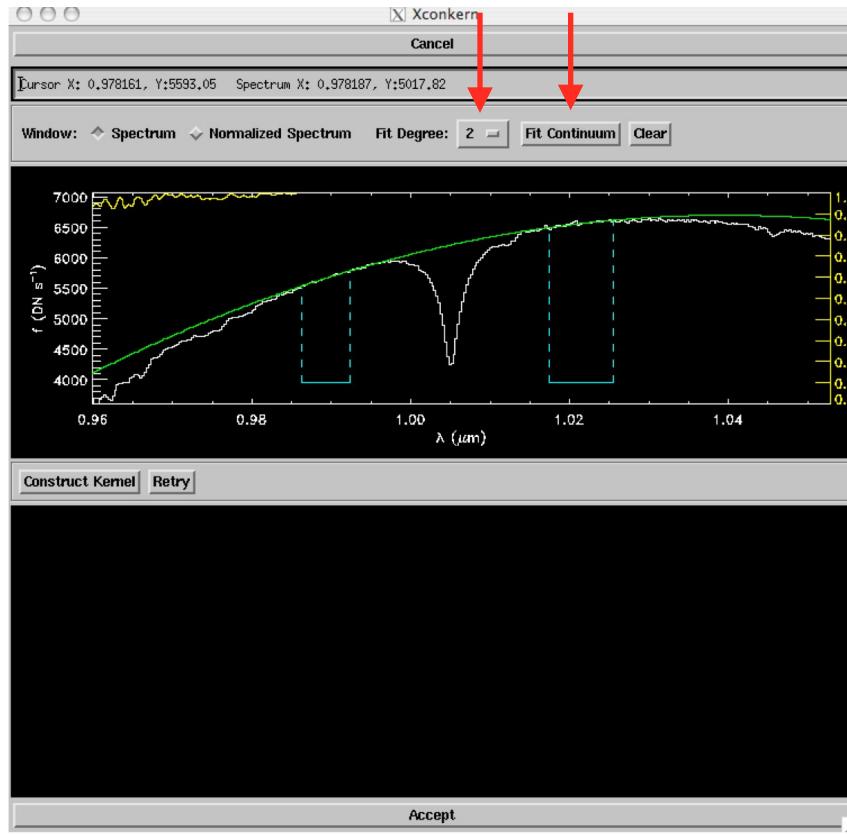


Press "n" (we need the normalize tool, not s for select.) To select the region to normalize over, left click on the plot with your mouse. It works better when two ranges are selected on either side instead of just one.

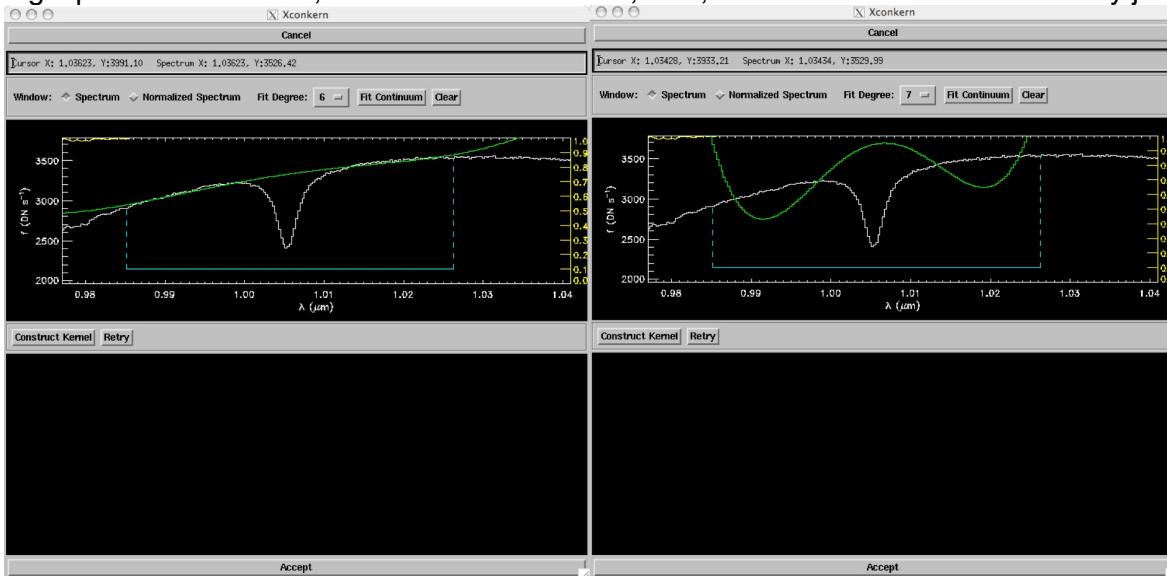


The goal is to find a smooth curve that fits the continuum, from which the spectrum can be normalized to get a level continuum. This is done using a polynomial fit of order 0-24. This can be tricky. You may need to iterate on a "best choice" wavelength region and polynomial fit order, and go back between the two in order to find something reasonable. The experts tell us 6-9 works well. Let's find out, but sometimes even 2 give acceptable results.

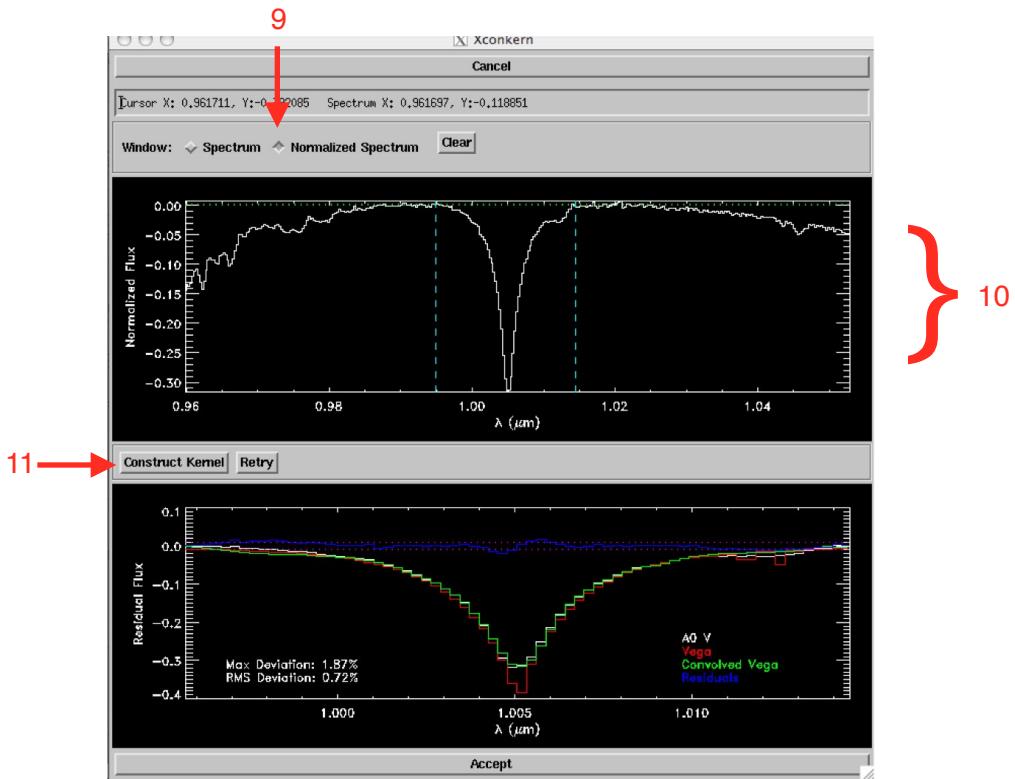
Select the polynomial order using the pull-down menu and hit the Fit Continuum button. Here is an example of a 2nd order fit. Looks pretty decent:



With just a single point on each side, a 6th order fit does fine, also, but a 7th order one is obviously junk:



Let's stick to our original choice.....



9. Now, select “Normalize Spectrum”.

10. Hit “s” (select) and select the wavelength range over which you plan to do this with a left-click on the short wavelength side, and another left-click on the right. Be sure to get all of the line!

11. Press “Construct Kernel”

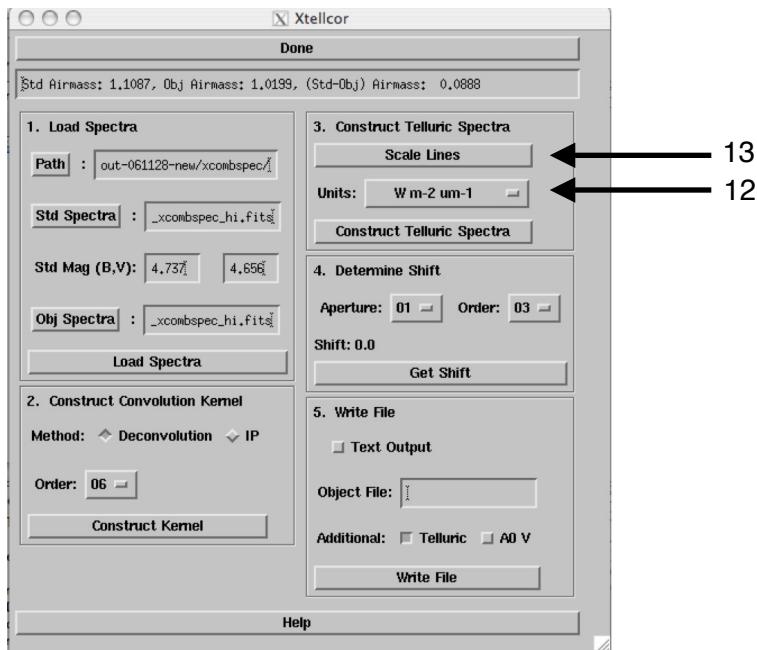
Now, note that the lower panel shows 3 curves. One is your A0 STD (white), one is the spectrum of Vega (red), the third is Vega shifted in velocity to match your A0 STD and convolved with a broadening function to attempt to match your A0 star. The upper blue line is the difference.

Usually the largest residuals will be near the line core. Vega is a rapid-rotator, but is seen pole-on, and the exact shape of the line depends on a lot of factors.

Click “Accept” once everything is to your liking.

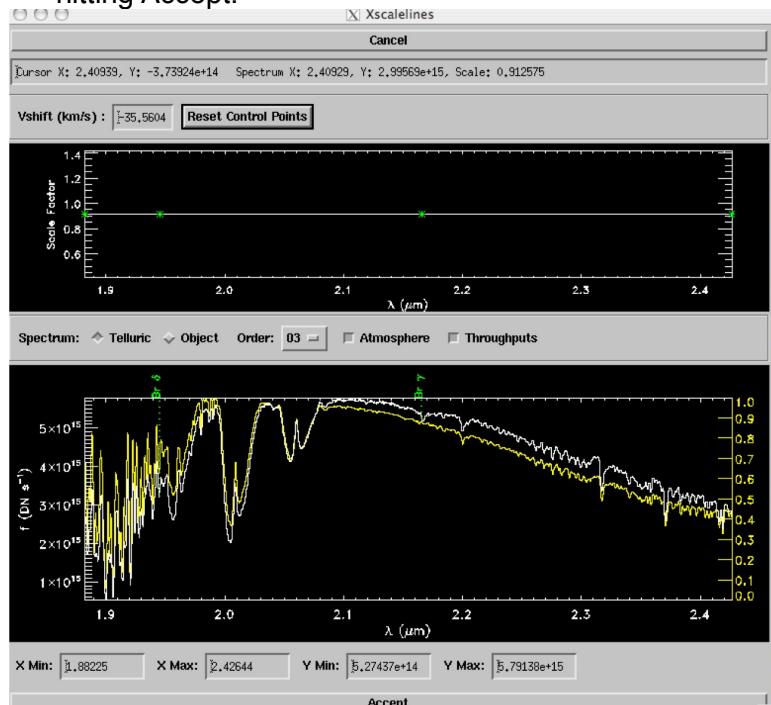
XSCALELINES

Now it's time to scale the lines in the Hydrogen likes using this profile. *One has to do this line by line, in all of the orders, before finally accepting the result by hitting the **Accept** bar at the bottom of the window.*



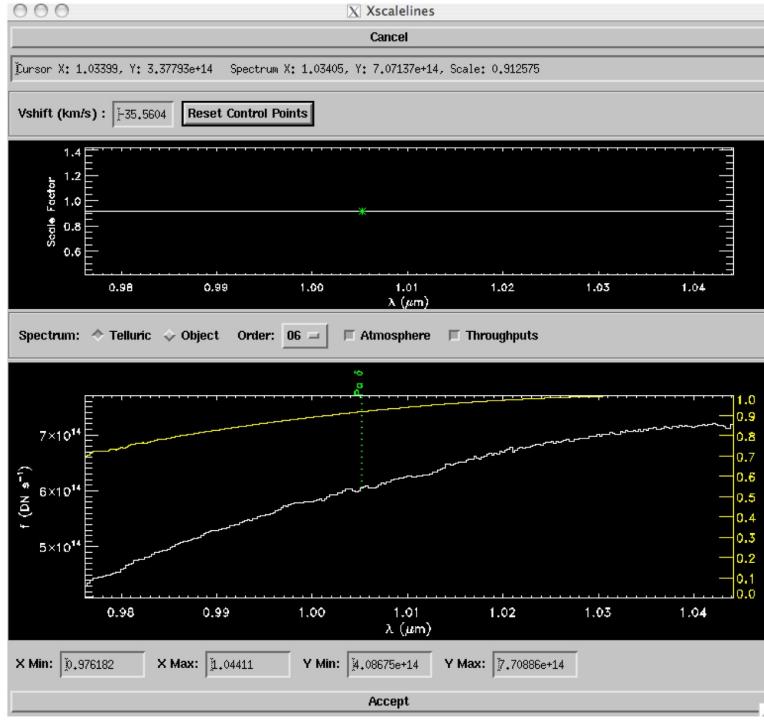
12. Choose what units to use. "Wm-2 um-1".

13. Then press "Scale Lines". REMEMBER – Again, when scaling the lines, you need to do *all of the orders before hitting Accept.*



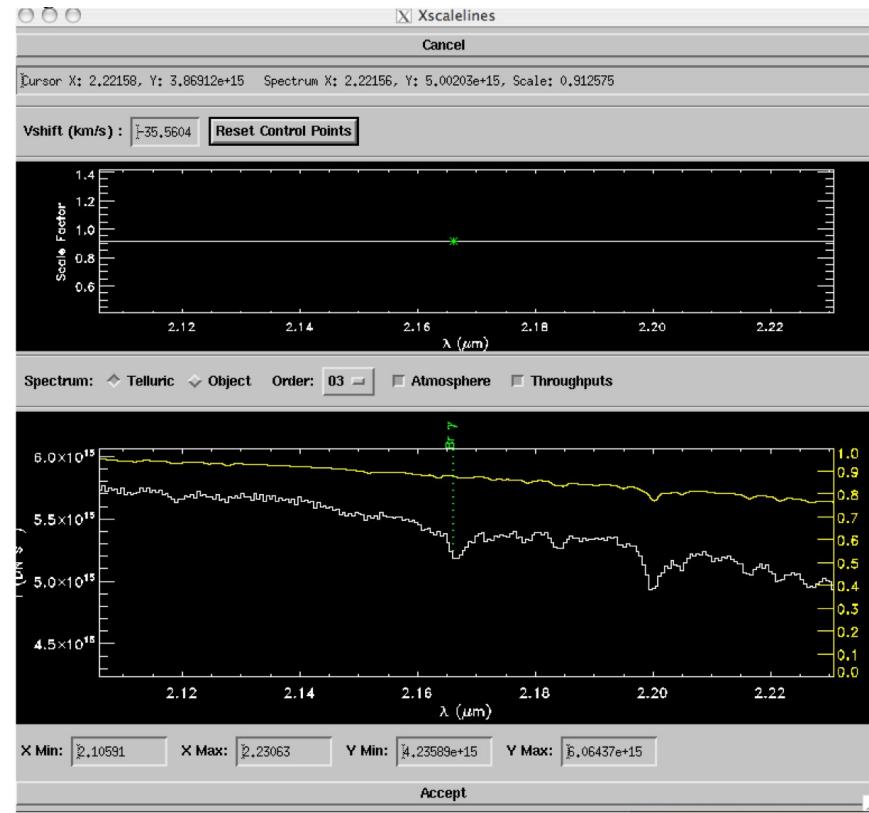
Now, if all A0V stars had spectra identical to that of Vega, then all would be effectively removed by constructing the kernel (in this case Pa). But since they do not all have the exact same effective temperature nor surface gravity, small residuals in either emission (over-corrected) or absorption (under-corrected) will remain. (Note: because Vega is a rapid rotator, these parameters will differ from its pole to its equator, so it is almost guaranteed that there will be residuals). Here, we see that the remaining residual for Br is present in absorption, but it is weak, only a few percent of the continuum level.

This is pretty typical of the lines in SXD Order 3.



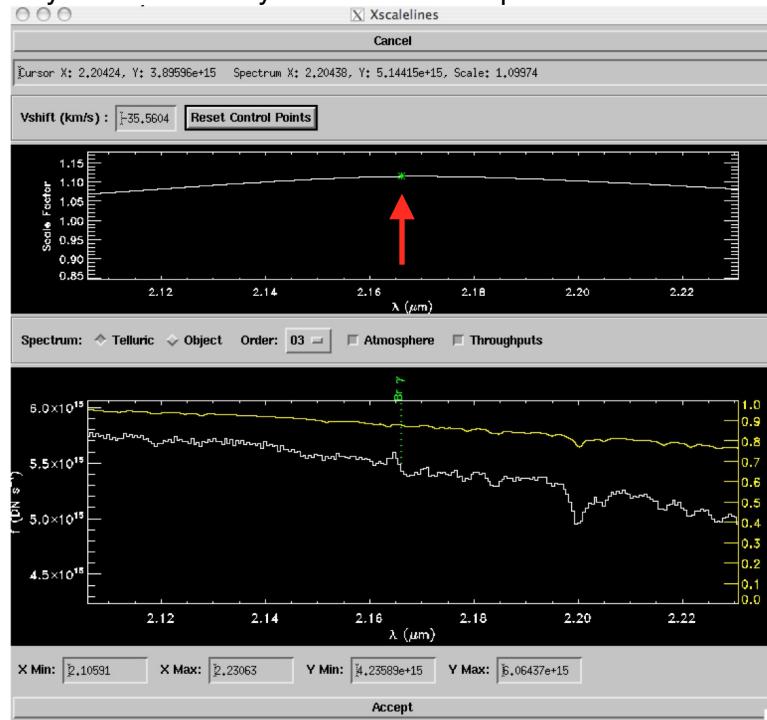
In Order 6 we see the Pa line that was used to construct the kernel, and the effect that the non-zero residuals had on the final product. Again, the structure is at the 1-2% level.

Now, the idea here is to try to make your “rectified” A0V star look like the “standard” telluric spectrum shown in yellow. Now one might want to smooth out the residual hydrogen lines.



Estimate Tool

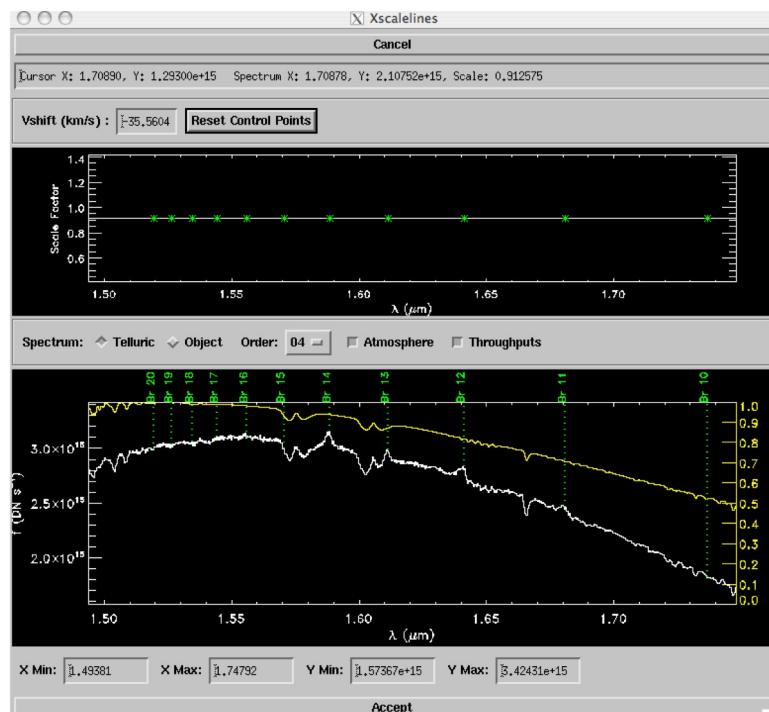
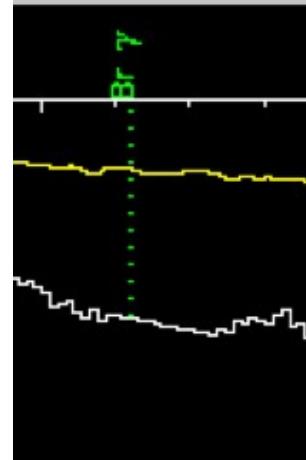
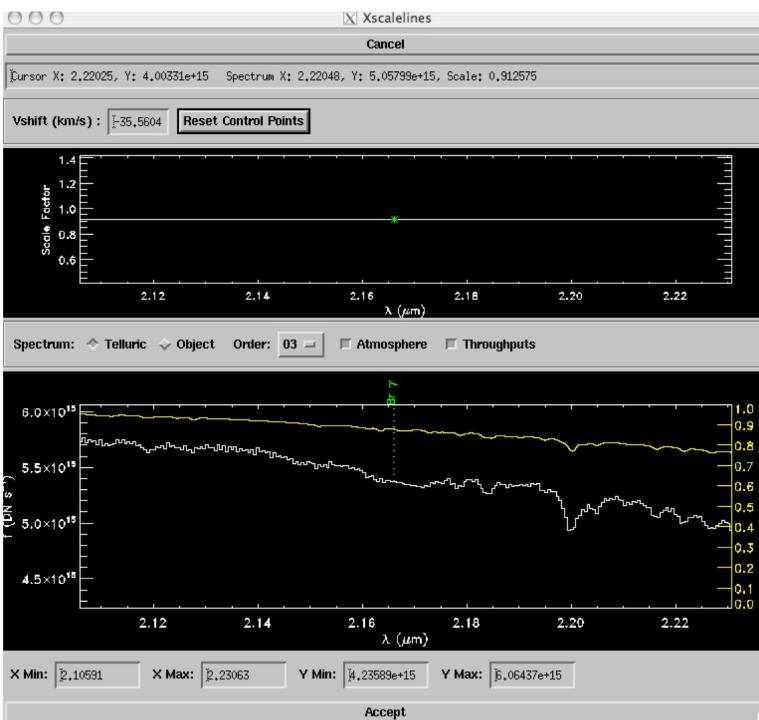
There are three ways to do this. One uses a program-generated estimate by hitting “e” and clicking on either side of the feature. I have personally found this tricky. Here is an example:



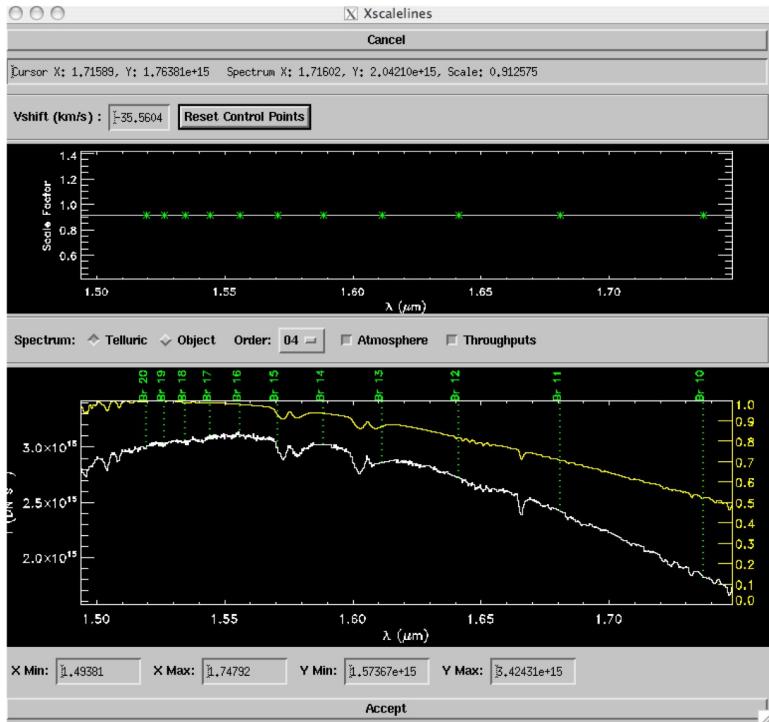
Note that the “set point” in the spline fit (the green asterisk in the upper panel) has now been raised. A second way to do this is to “grab” the set point by dragging it with your mouse to the desired level. More on this later....

Fix Tool

Finally one can use the “fix” routine by hitting the “f” key and then clicking on the spectrum on either side of the line. This produces a linear interpolation over the feature. In the case of lines with noise or blended with real telluric lines, you may want to (read “need to”) do this a few times. It will remember what you did before, since it has already replaced those data! But it also means that if you over-do it, you will have to go back and try again.



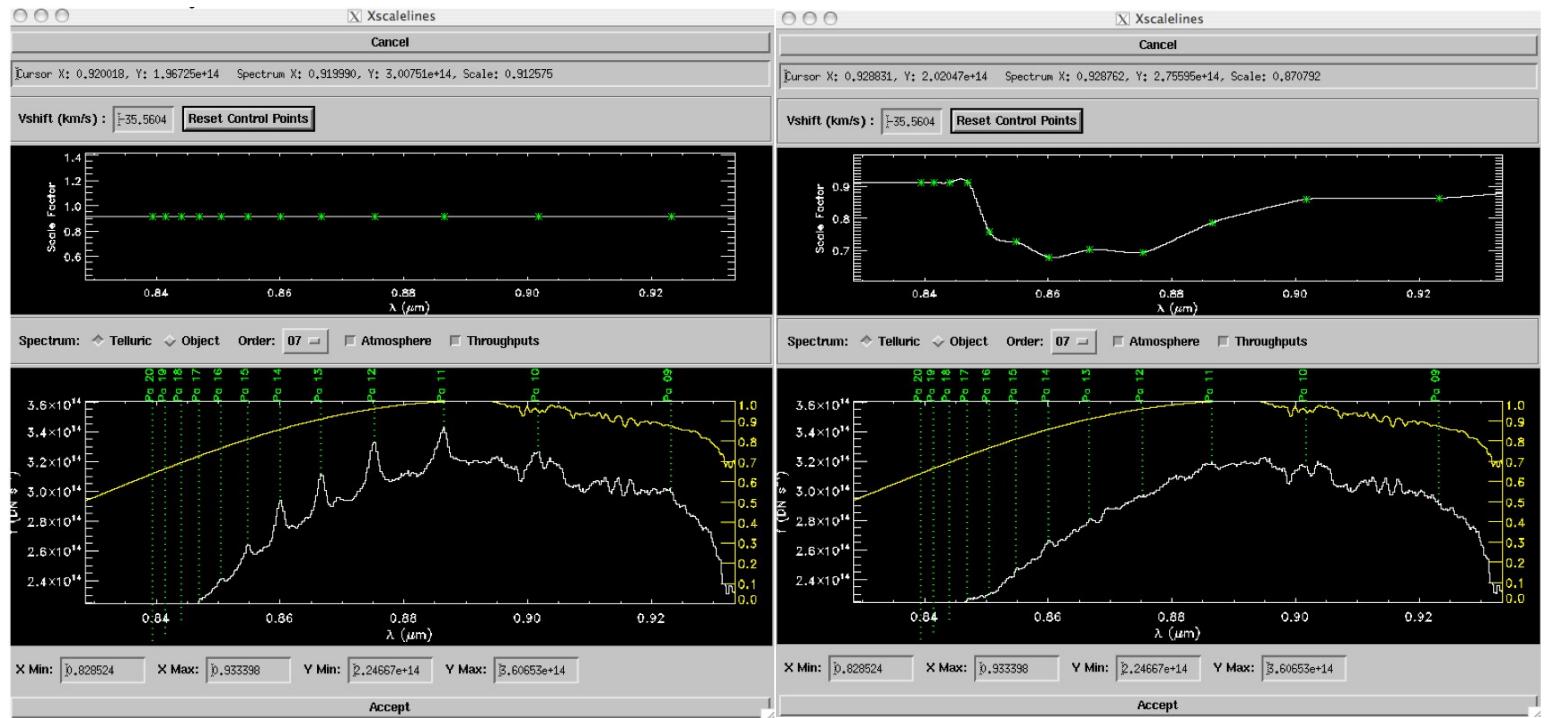
Here we see a case where the lower order Br lines have positive residuals (“emission”) and should be corrected, but by Br 15, the residuals are tiny, and even switch over to weak absorption by Br 18. This is normal.



Here I used “f” to fix the Br 11-13 (Br 12 sits in a telluric feature, so this is a bit tricky). Granted, this is a bit “brute force” but it seems to work.

These residuals are more noticeable. The Pa lines often have strong residuals are closely packed. I have found that dragging the set points in this region is a bad way to go if you are interested in the *continuum shape* in this region. This is because dragging any one point also move the adjacent ones¹. That’s the nature of splines. So if you go back & forth dragging the set points around you will affect the continuum levels but 10%, 20%, or maybe more.

Splines Tool

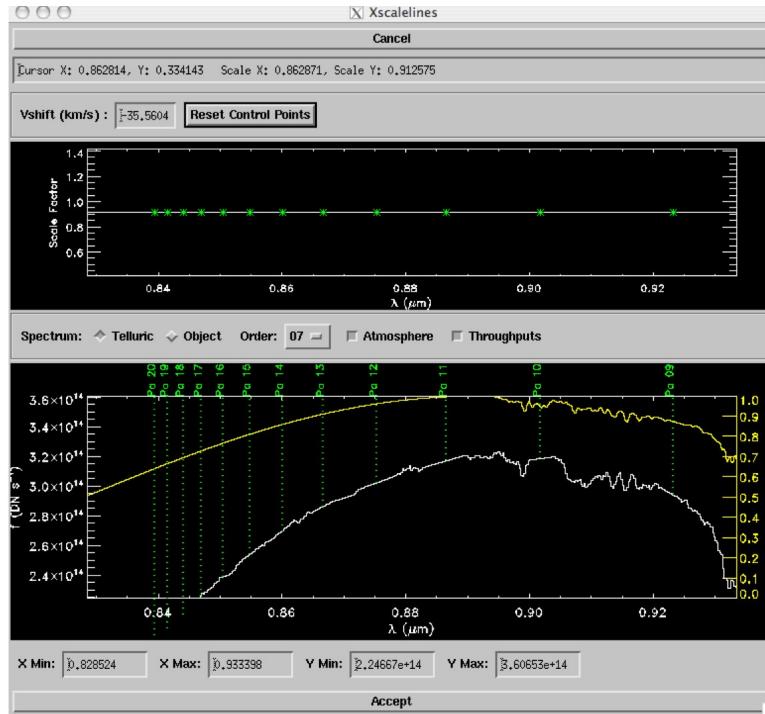


In the example shown here, the residuals were removed by moving the set points downward by about 20%. But the entire continuum in that region has been depressed by that amount. (The residuals from the Pa₄ are still present).

*Note: I have seen where using the spline adjustment technique will produce a very noticeable “kink” in the spectra in Order 4 if the high order Br lines have strong residuals, for the same reason. **Very bad of maintaining the integrity of the continuum shape.** However, if you are not concerned with overall spectral shape or photometric integrity, the method of moving the set point is fine. If you are interested in these, however, be sure that using this technique does not move the overall continuum more than your own tolerance level. Because I want to get close to 3% tolerance or better, I avoid this technique if it does anything to the continuum level.*

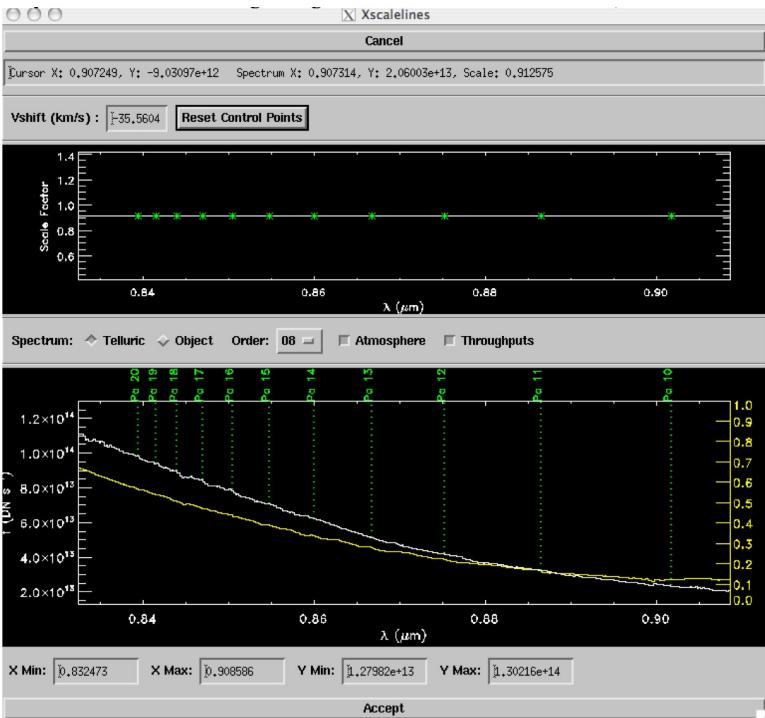
More on Fix Tool

Now using “f”:

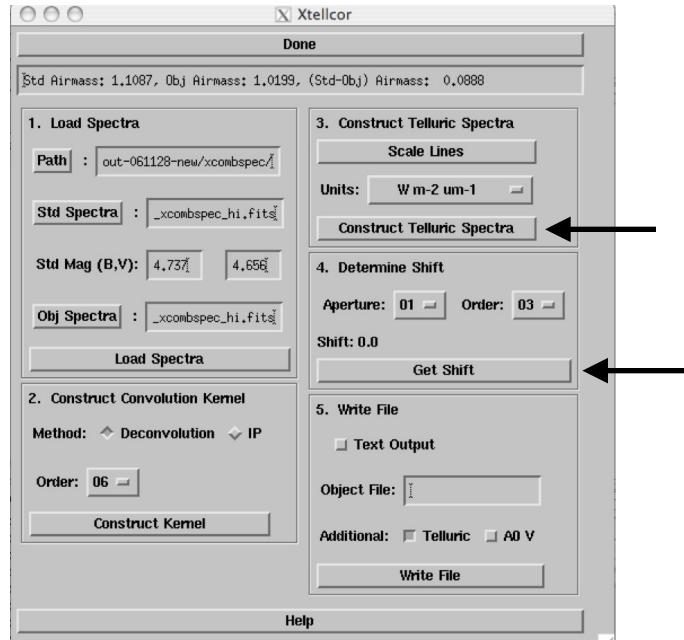


With that said, I have seen cases – late type PMS stars (i.e T Tauri stars) - which show a “hump” between 1.5 and 1.8 μm , even after using the “f” method. It may be intrinsic. I am still looking into this.

Now, in the LXD 2.3 mode of SpeX, many of the lines on Order 7 also appear in Order 8. They should still be adjusted, since you will later merge the various orders. Usually one order will have higher signal/noise at one end of the order, and smaller at the other end.



Once you have done ALL of the orders, hit the **Accept** button.



With all these “tweaks” in place it is time to construct the telluric spectrum by clicking on the appropriate button.

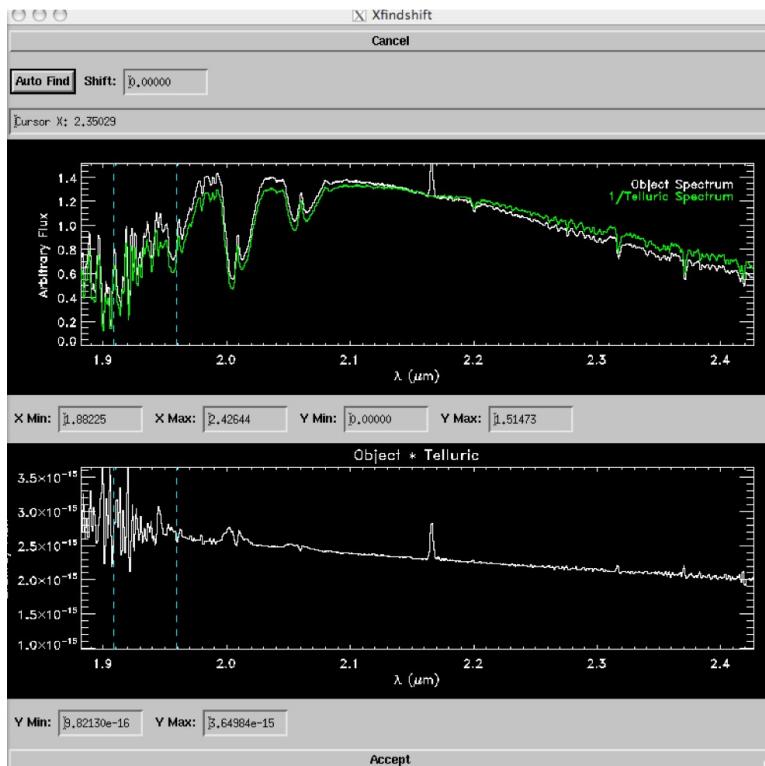
You won't see anything happen, but you are ready to go to the next step, to line up your telluric spectrum with that of your science target and divide it out – hopefully removing the telluric features. If conditions are really good, and the airmass of your science target and A0V calibration star match well, some of the weaker features divide out well. Really strong ones, where both spectra hit close to zero, you get a result that you might expect by dividing zero by zero.

Now, your telluric spectrum and target star spectrum may not be precisely matched in wavelength, so select an order and **Get Shift**.

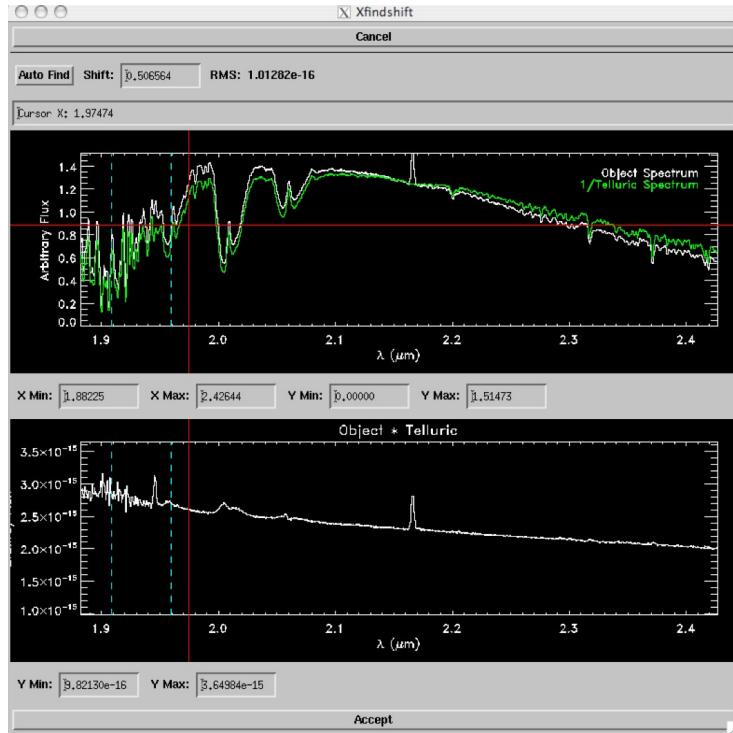
XFINDSHIFT

Here you need to select a region to shift. USUALLY, one with medium strength telluric features may be the best. The routine will shift the two back & forth until the Object*Telluric scatter is minimized. With really weak features, I suspect that this will be more highly affected by noise and other small features. Using the largest may give too much weight to fitting the 0/0 locations. LOOK at the results and use your own best judgment. So select your region, hit **Auto Find** and watch the fun. You may also enter this *manually* and hit *return*. Usually most of the shifts will be similar, so you can use other orders as a guide. Order 8 has so little to work with that it is often a candidate for the manual method.

Before:

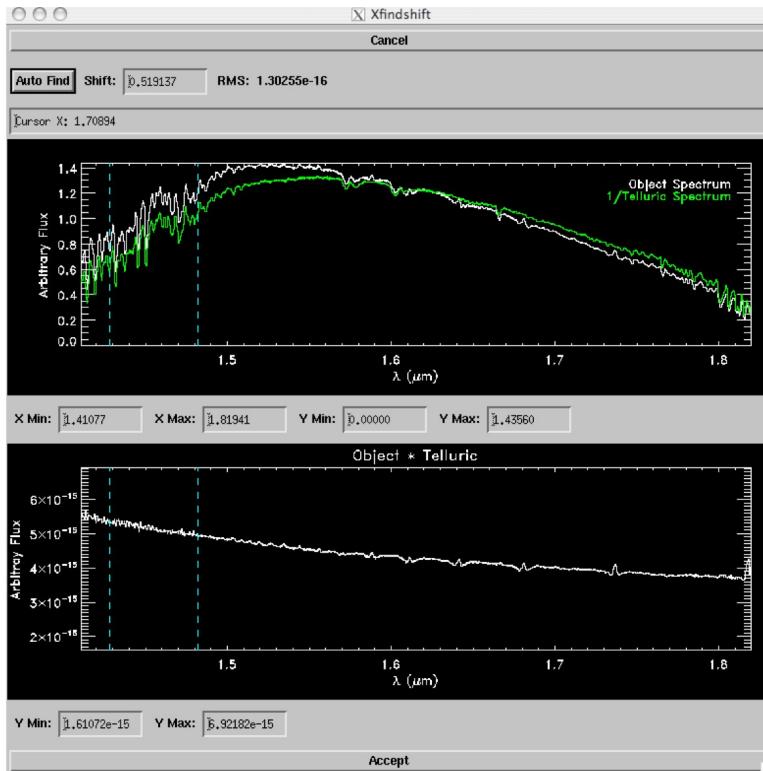


After:



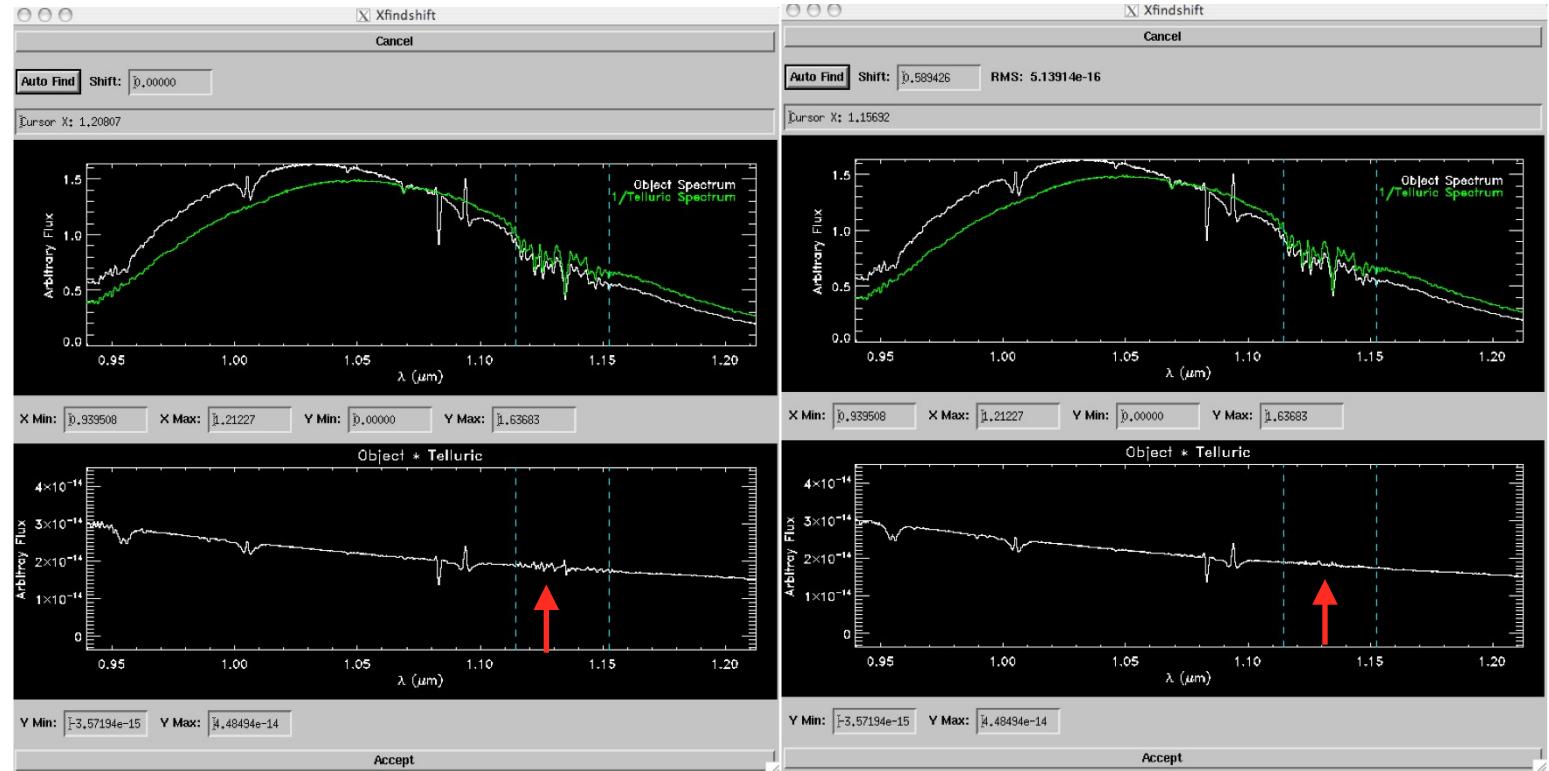
Unlike Scale Lines, here you need to hit *Accept* after every order is treated! Repeat this process for all orders.

This is an example where the telluric features are not that strong, and nearly completely get cancelled out:



Note there the weak circumstellar emission lines are poking out of the photospheric absorption lines.

Another “before” and “after” pair. Not that even before shifting the telluric cancellation is pretty good. But after shifting it is even better!

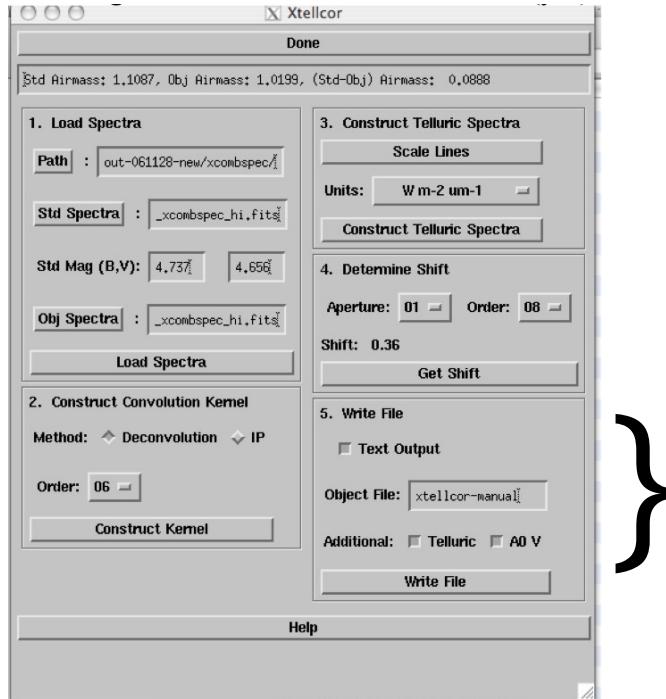


In the above example the circumstellar H lines are seen in emission, but the He I line at $1.083 \mu\text{m}$ is in absorption.

In many PMS stars this will have a P Cygni line profile indicative of a wind (either from the star or from the disk).

Here it looks like a very weak “inverse” P Cyg profile is present, possibly indicating infall.

After you have **Accept**-ed all of the orders, it's time to save your work.

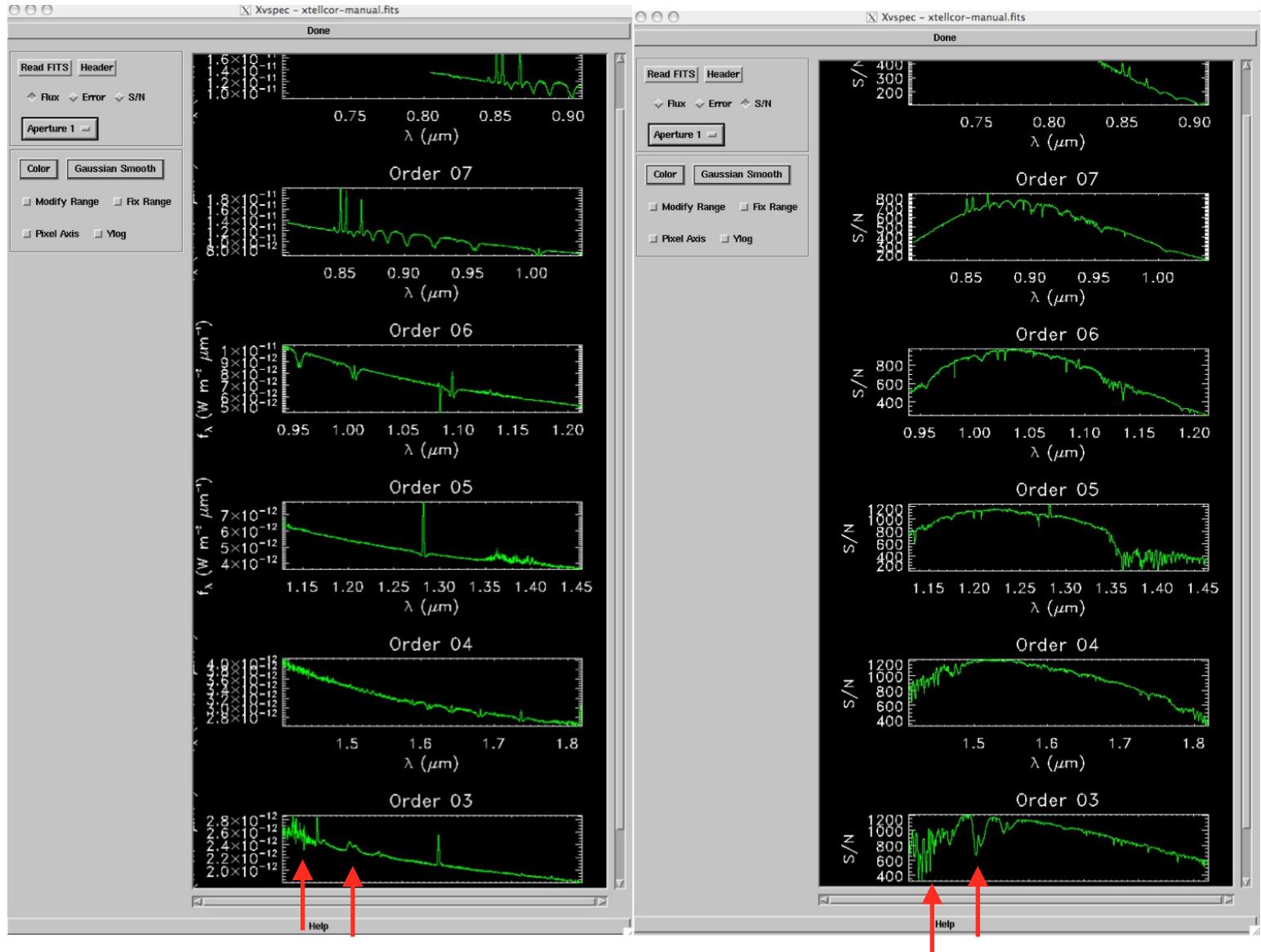


In general it is probably a good idea to **write** the telluric-corrected spectrum of the science target, but also the A0V star and especially the Telluric spectrum, which can be used again if you have another science target for which the

A0V star you just used is appropriate (airmass, telescope pointing, etc.). For that you use the **xtellcor_finish.pro** routine. This allows you to skip having to construct the Kernel & Scale Lines from scratch all over again. You also have the option of saving a Text version, although I have not found these useful (yet).

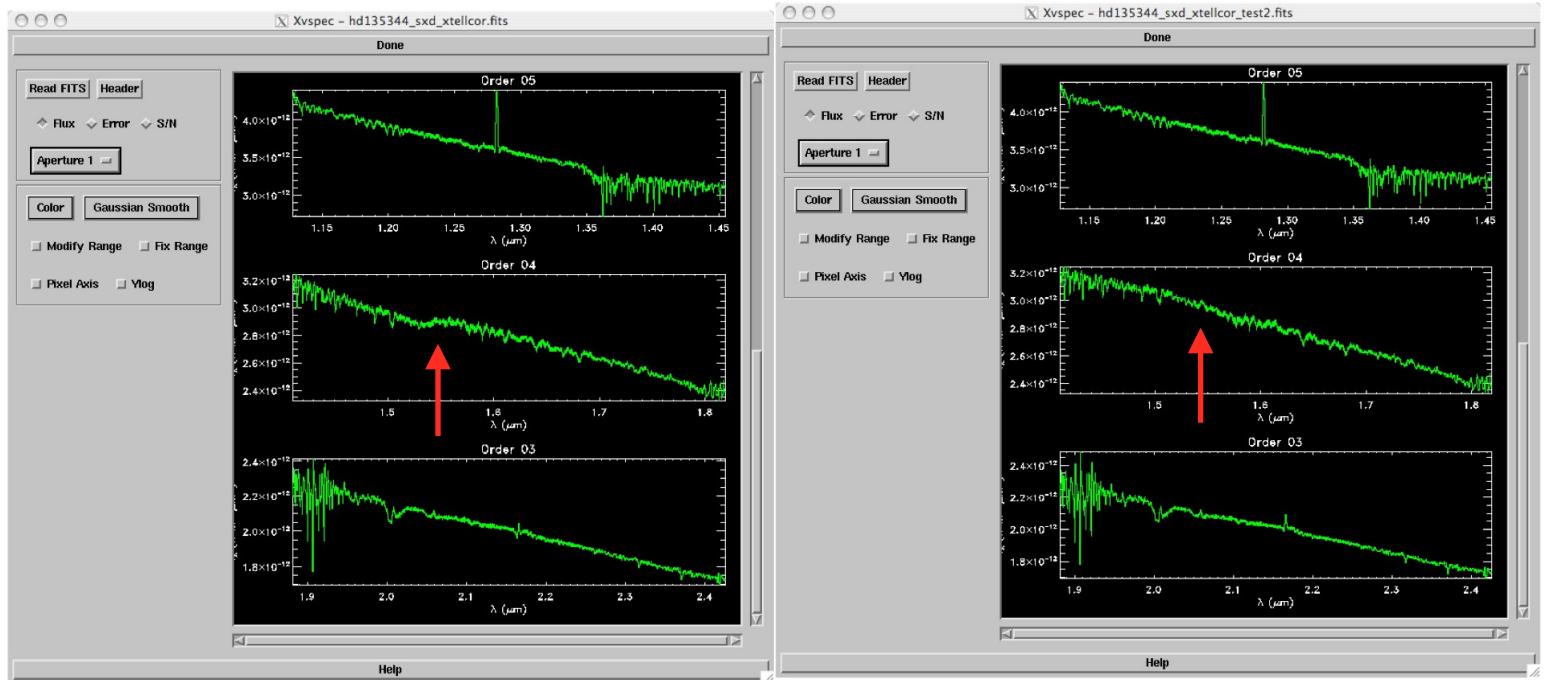
The file should be named with the star name, SXD or LXD, and “xtellcor”. Like “V1247Ori_SXD_xtellcor” or “V1247Ori_LXD_xtellcor”. The files should be saved to the output folder.

When you write the files, xtellcor will display your final spectra. You can display them as fluxes or signal-to-noise.



Note that some regions may have excellent S/N (in the hundreds) but can still have significant telluric features present if the airmass match was poor.

An example of when dragging the set points mangles your continuum. The left-hand version is victim to over-using this technique. The right-hand one shows what the continuum ought to look like at this stage:

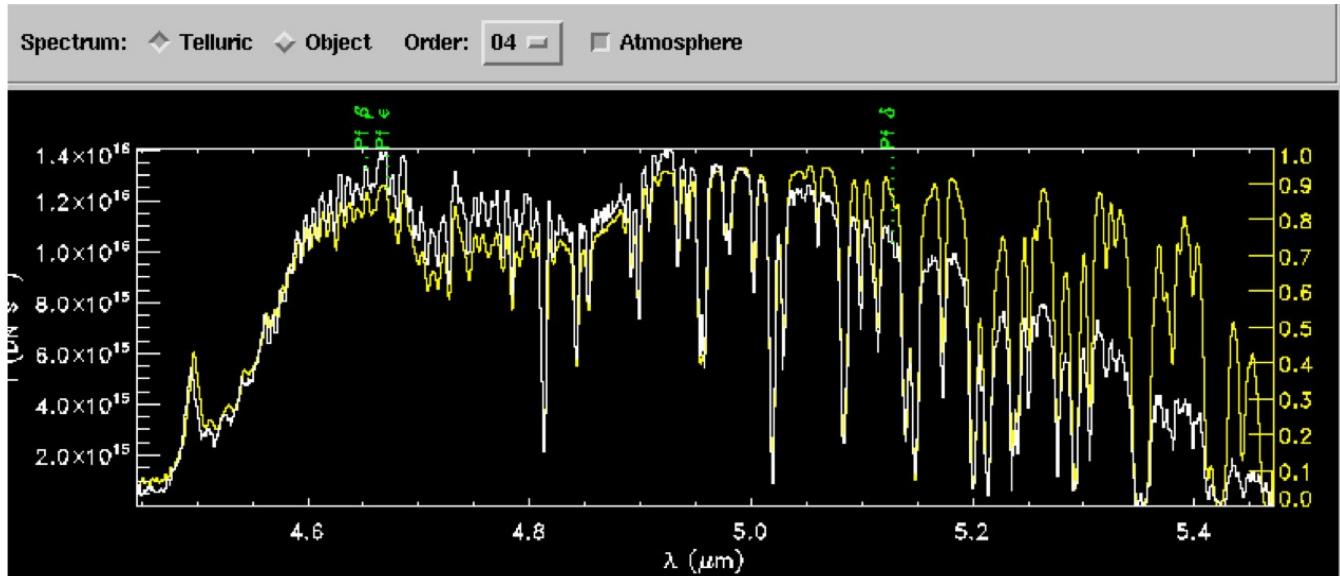


This is the “kink” that I described earlier.

XTELLCOR (LXD)

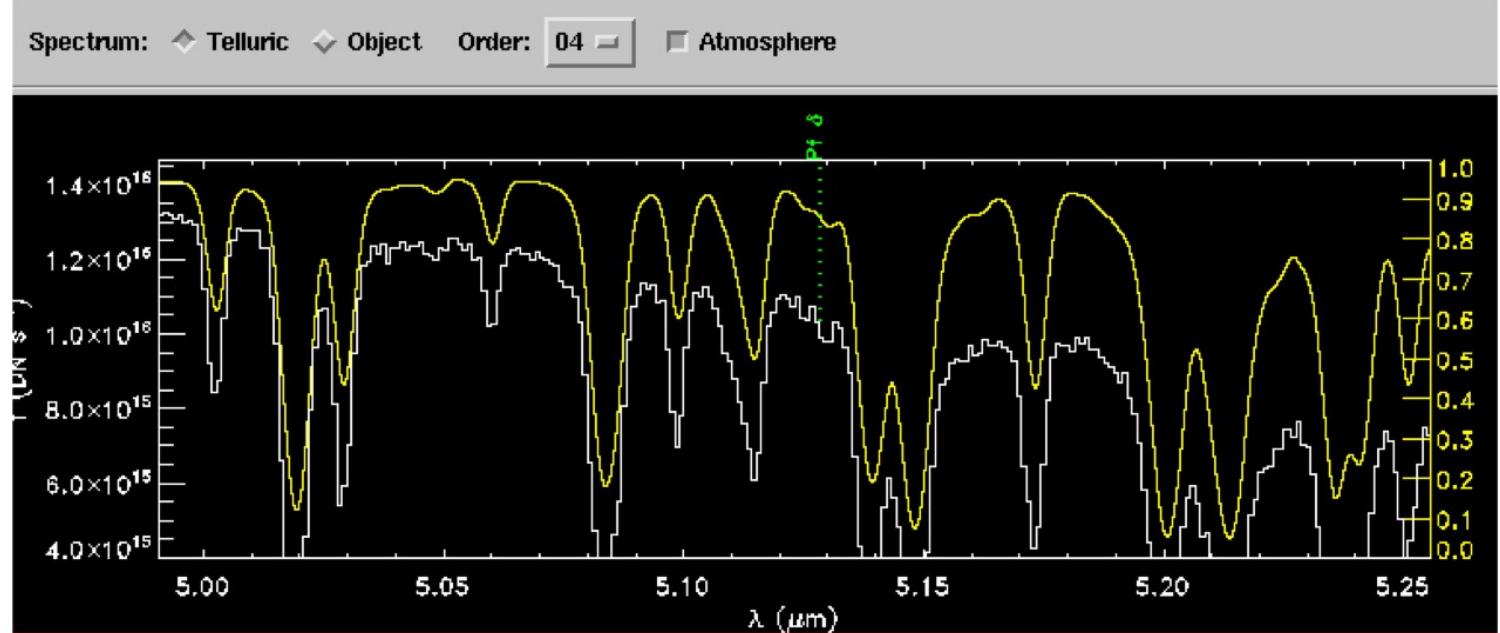
Now, for the LXD, I use the IP (Instrumental Profile) for construction of the Kernel. What about Scale Lines? Here we will examine each order. I will show the entire order and zoom in on one sample region.

Order 4:



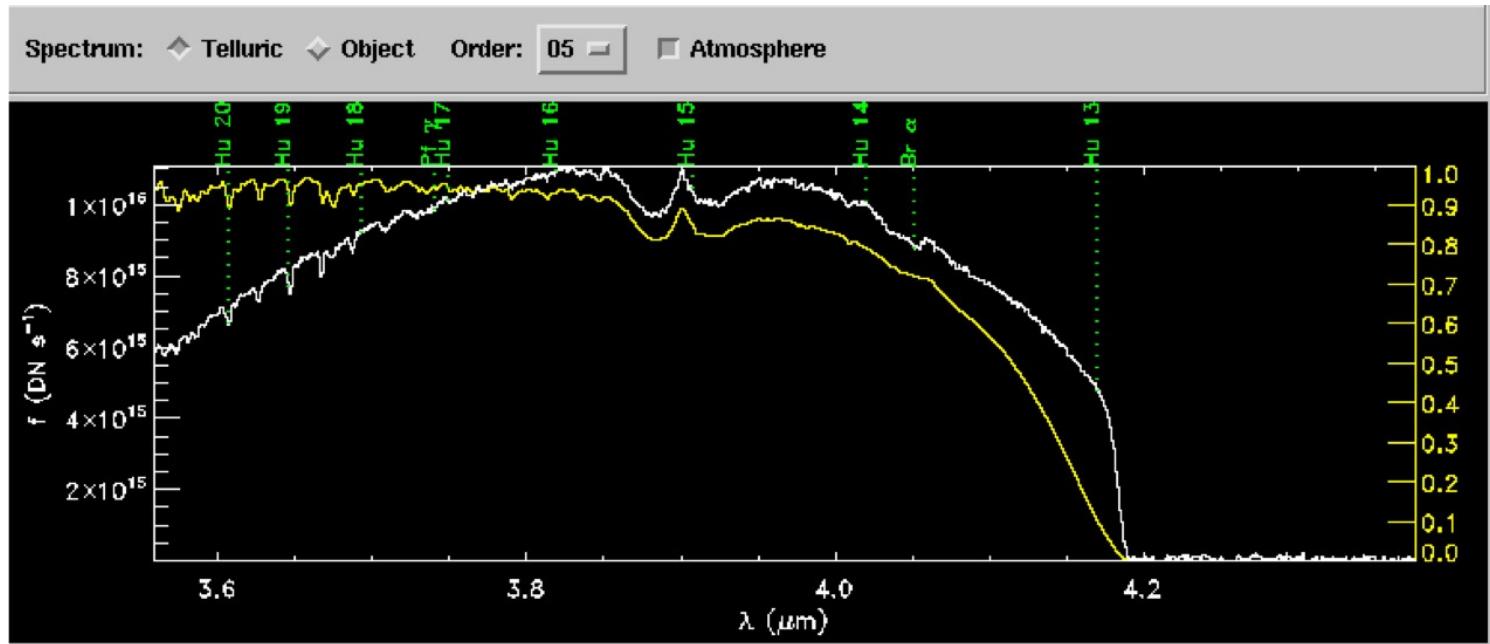
This is by far the noisiest portion, at least in terms of strong telluric lines. Can you imagine trying to do the continuum fitting here?

Now for the close-up:

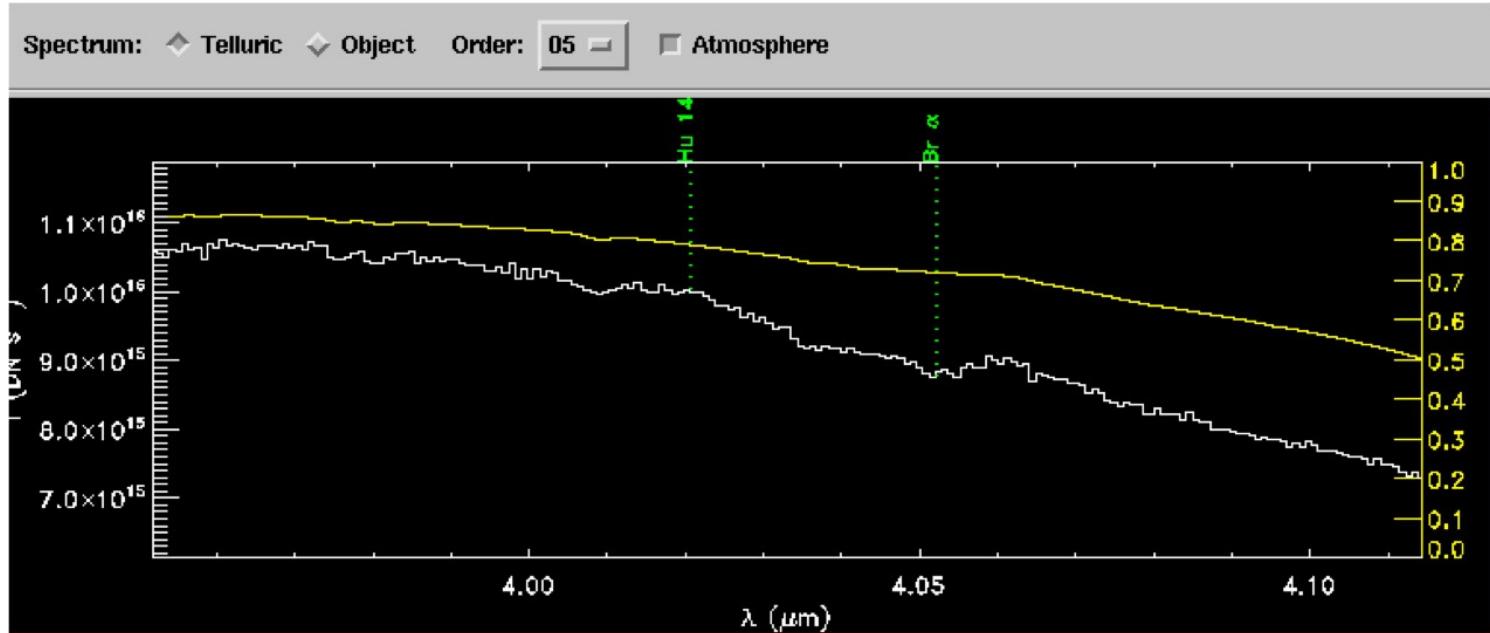


Here we see the results of applying the Kernel to the AOV spectrum. If Pf is present at all, it has apparently been corrected for pretty well.

Order 5:



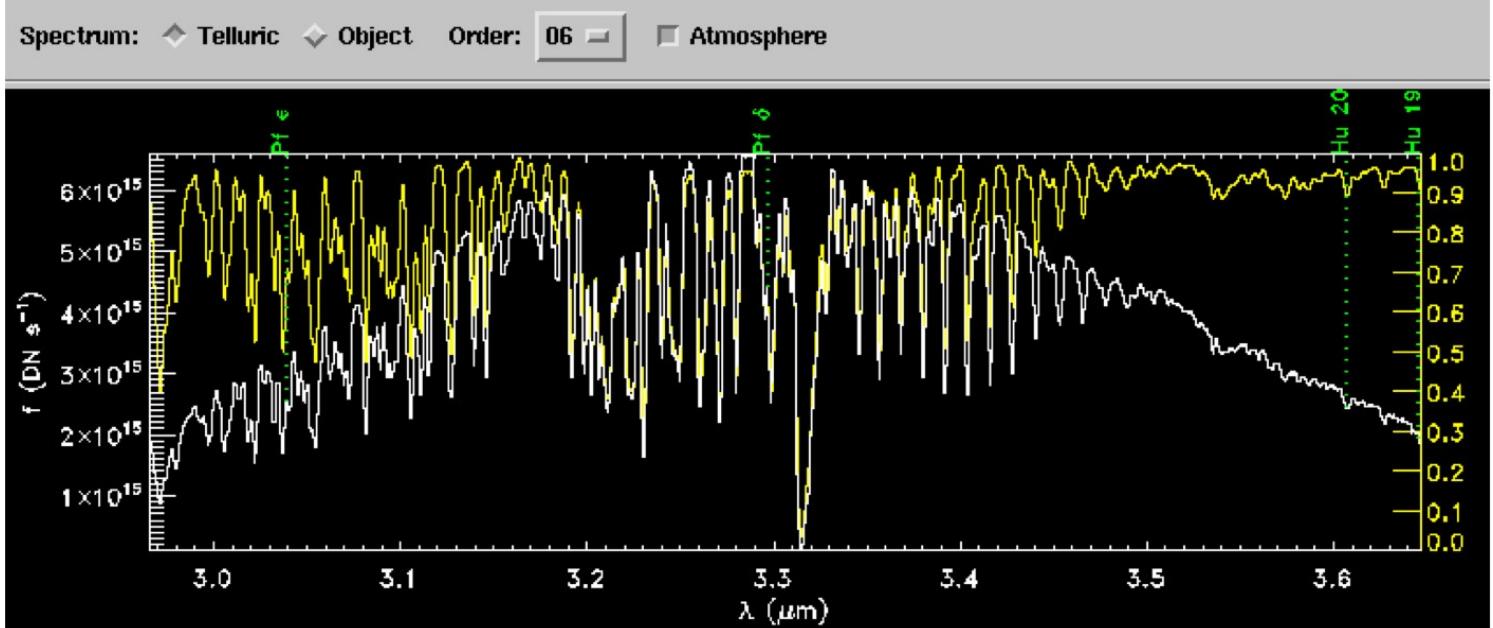
It appears that the Hu lines at the short wavelength end might need correcting, but if you look closely you will see twice that many lines there, and they don't look like others (i.e. Hu 16), suggesting that they are telluric lines that coincide with the Hu lines. The fact that the AOV spectrum here matches the telluric one means these should be left alone.



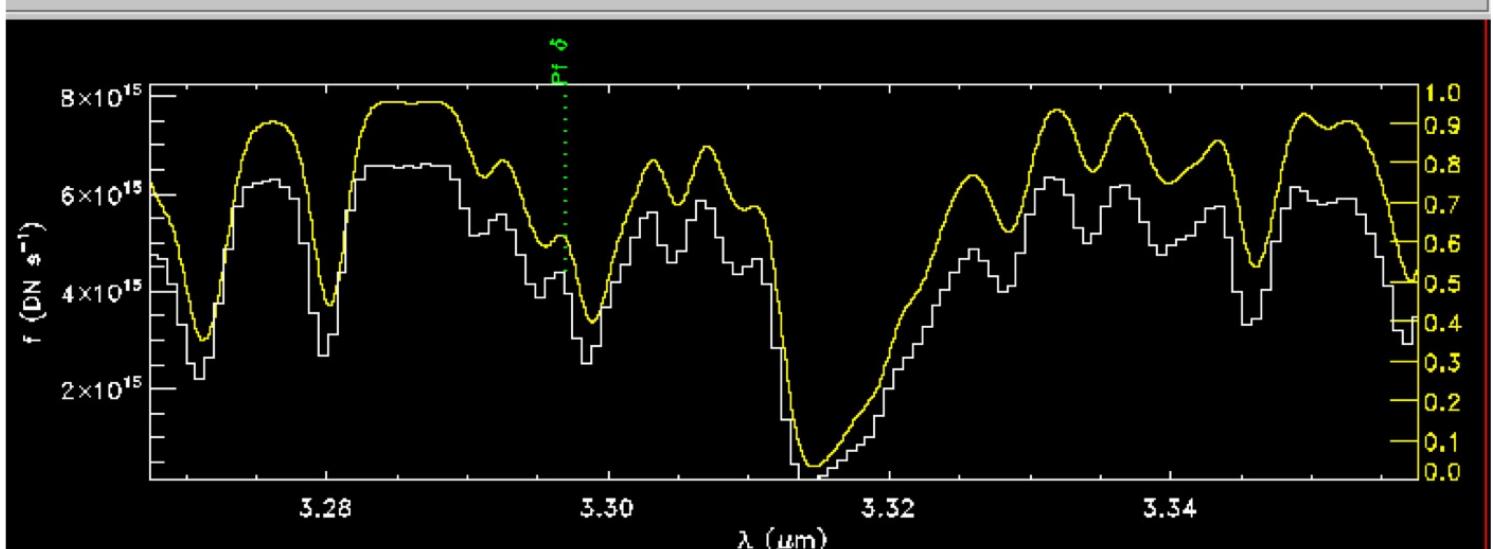
Here we see perhaps the only lines in this order that you might want to correct. Br looks real at about the few % level, and you might want to use the "fix" routine here. Perhaps also on Hu 14, but I don't. You might even skip Br if you don't need high precision in this line (like measuring its strength when it is weak compared to the continuum level).

This is the only order in LXD with any hope of fitting the continuum for constructing the Kernel, and the only line present is too weak to do this for.

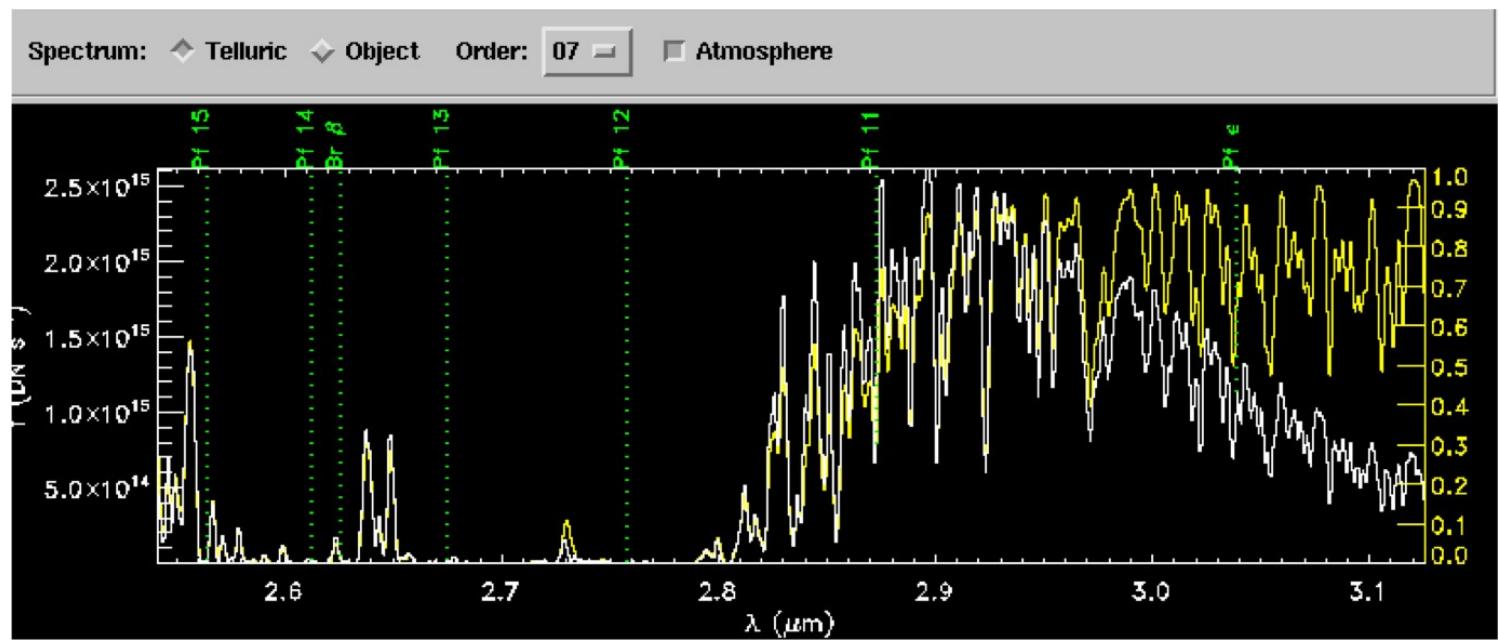
Order 6:



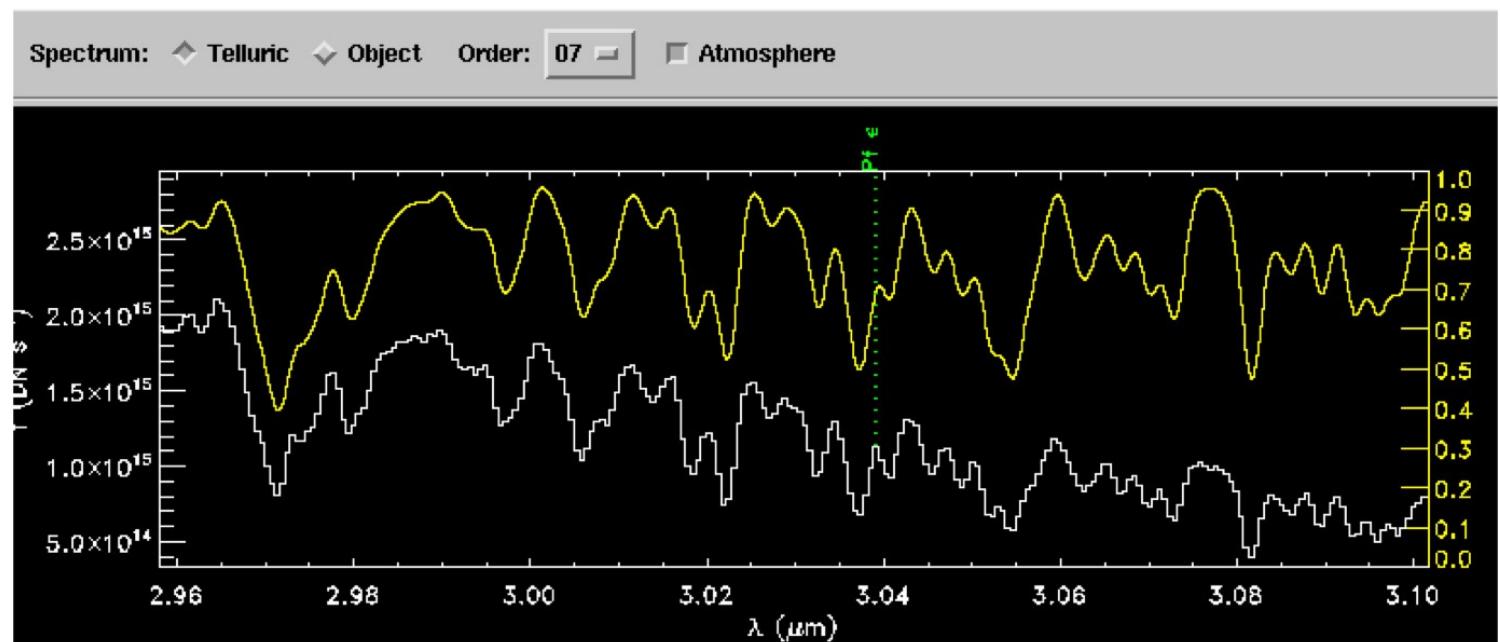
Spectrum: Telluric Object Order: 06 Atmosphere



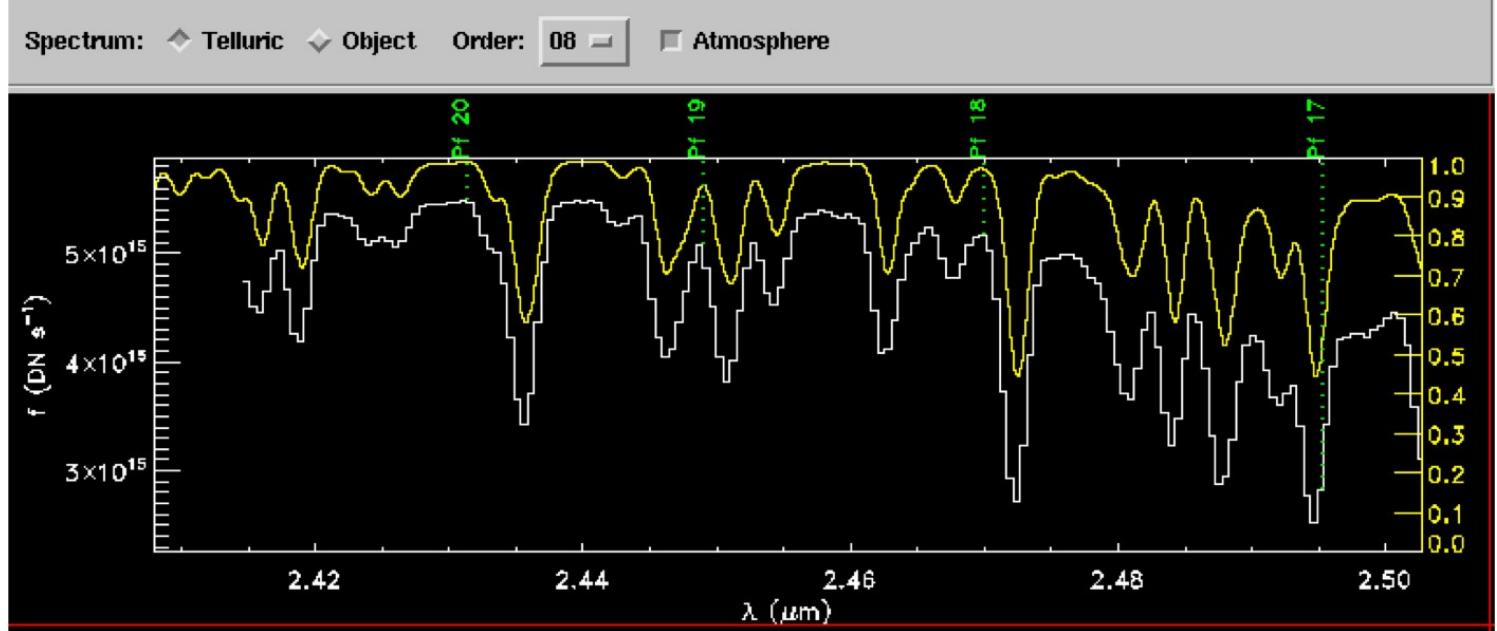
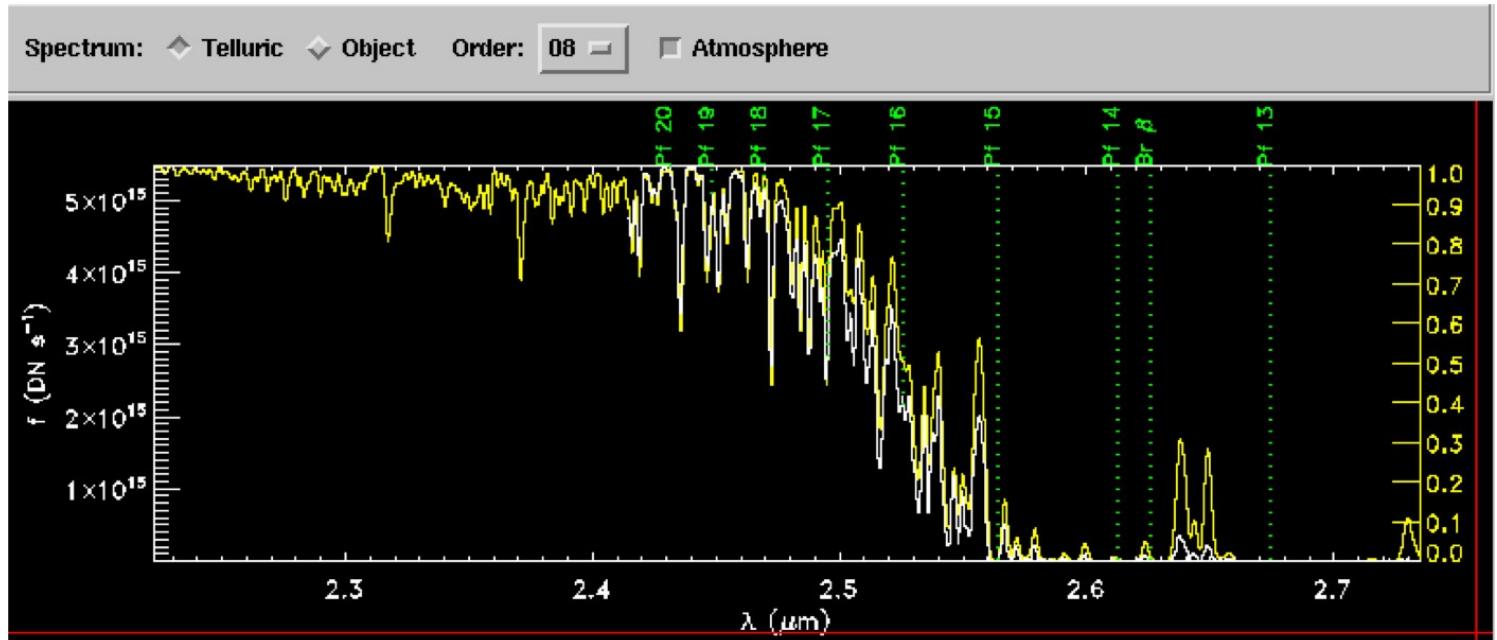
Order 7:



Another nasty order.



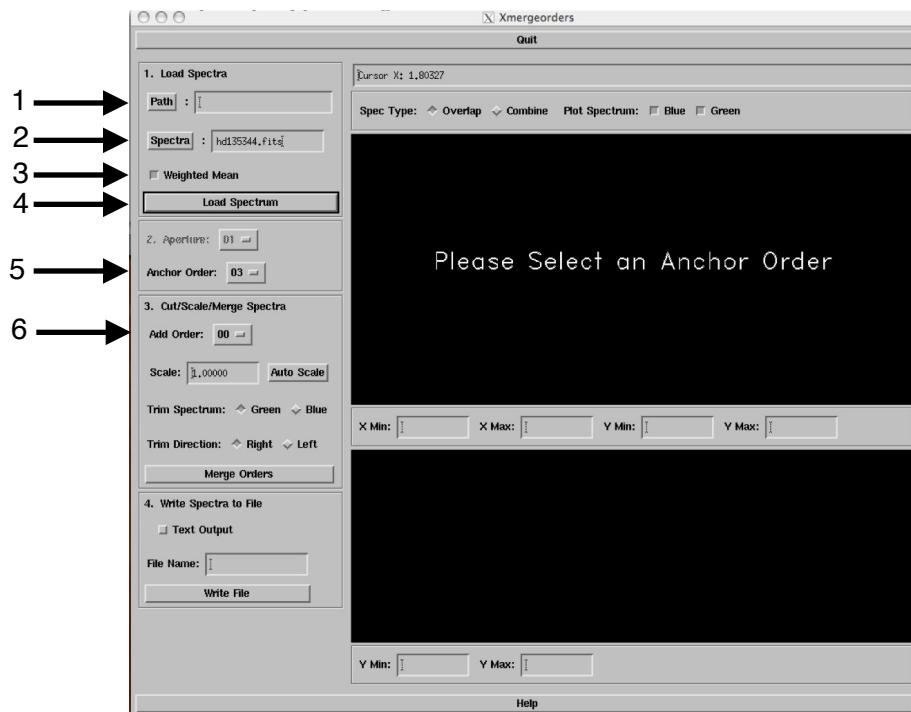
Order 8:



Enough said? Basically there does not seem much hope (point) is doing major corrections in LXD. Maybe Br in Order 5, but not much else. As a consequence, running **xtellcor** on the LD data actually takes less time than on SXD.

XMERGEORDERS

This goes quickly, and is largely self-explanatory. In the IDL prompt, type *xmergeorders*, and hit return.



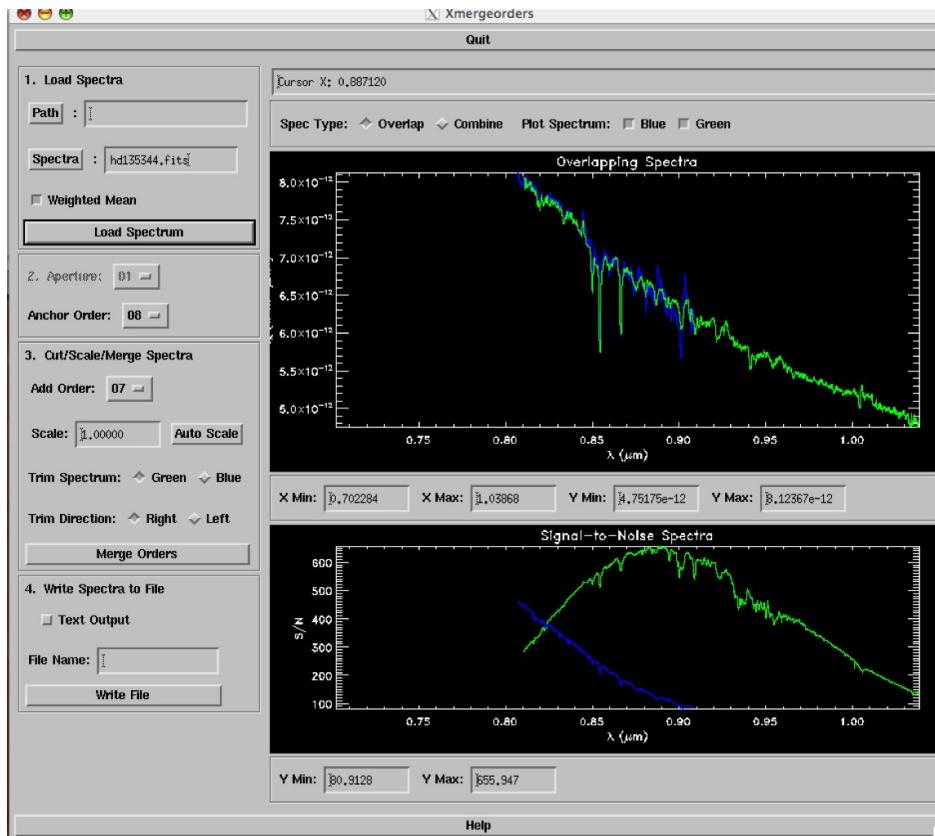
1. If the user is in the directory where the data is stored, skip to step 3. If not, type in the directory path for the data, or click on the "Path" button and choose the path from the list shown.
2. Type in the file name of the data, or click on the "Input Spectrum" button and choose the file from the list shown.
3. Select whether or not to allow errors to be propagated throughout the combining procedure. (You do.)
4. Click on the "Load Spectrum" button to load the data file.
5. Choose an "Anchor Order" from the pull down menu. The Anchor order is the order with which Xmergeorders will start. All other orders will be added to this order (with appropriate scaling factors if desired).

For the sake of illustration, I am going to start with Order 8, and build from there, starting with Order 7. Usually starting in the middle is a good idea.

6. After choosing an Anchor order, Xmergeorders will present a list of neighboring orders that can be merged with the anchor order. For example, if the Anchor order is selected to be order 7, Xmergeorders will allow the user to choose either order 6 or 8 to be merged with order 7. Select the order to be added. The anchor order and the order to be added will both be plotted in the upper plot window next to the control panel for Xmergeorders. The anchor order will appear in blue, and the order to be added will appear in green. The S/N spectra will appear in the bottom window.

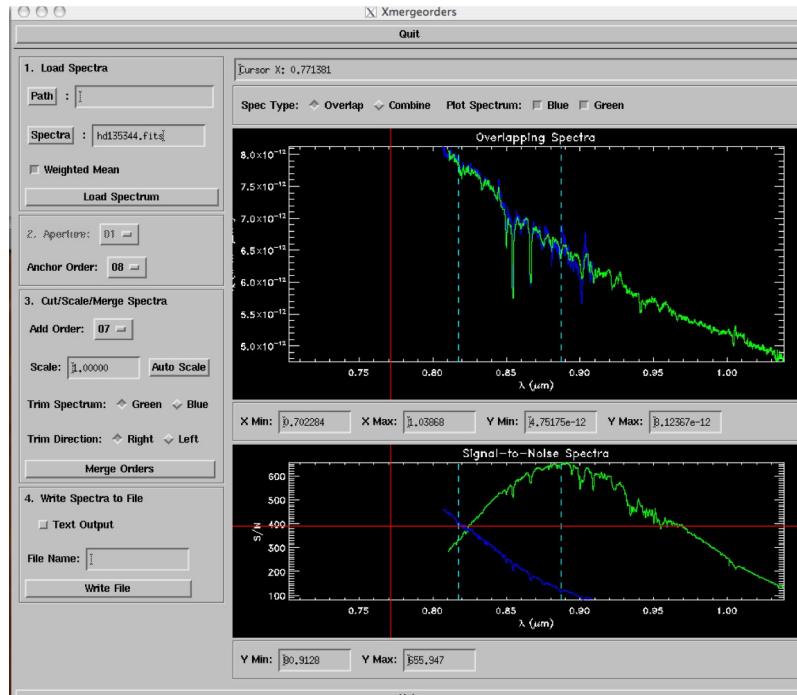
While you can use the “pull down menu” for selecting the order to be added, I found that on my machine I had to actually “click” on it to load the spectra for these orders (sometimes double-click, slowly or press-hold). If not, the next step produces an “index out of range” error!

This is what it should look like. Here Order 8 is in blue, and Order 7 is in green.

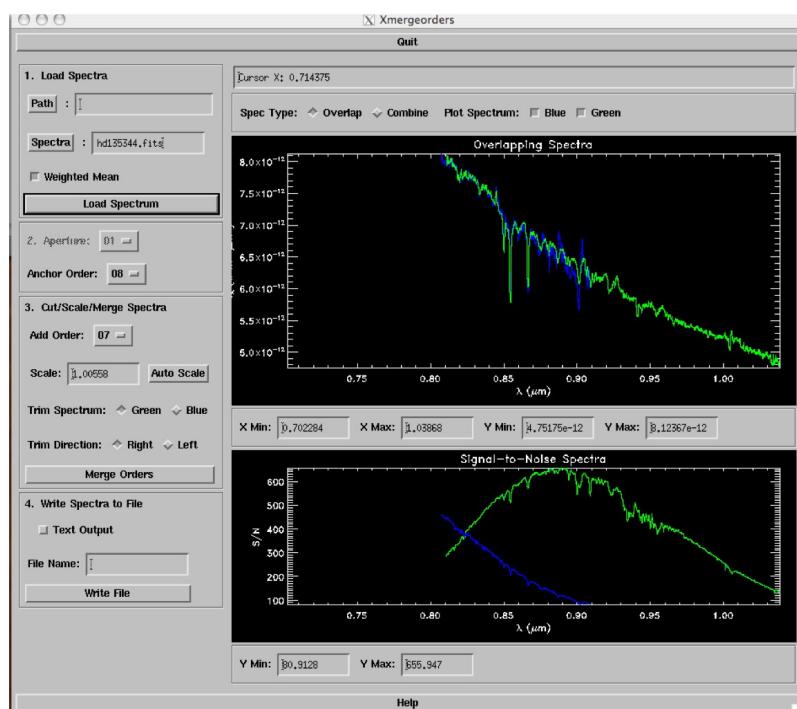


7. The added order can be scaled to match the level of the anchor order. Type "s" (for "select"), and click with the left-most mouse button on the short and long wavelength boundaries of the region to be used to compute the relative scaling factor. When the user clicks on the "Auto Scale" button, Xmergeorder will compute the best scale factor for the added order and re-plot the scaled spectrum. The computed scale factor will be shown in the "Scale:" field. Alternatively, the user can enter a number in this field and Xmergeorders will scale the added order accordingly. To reset the scale, type 1.0 in the scale field and hit return. Note the 's' button will only work if the two spectra overlap. If there is no overlap, (e.g., Orders 3 and 4 of ShortXD) the user can still scale the added order by manually typing the scale factor in the Scale field.

Here we will be scaling order 7 to order 8:



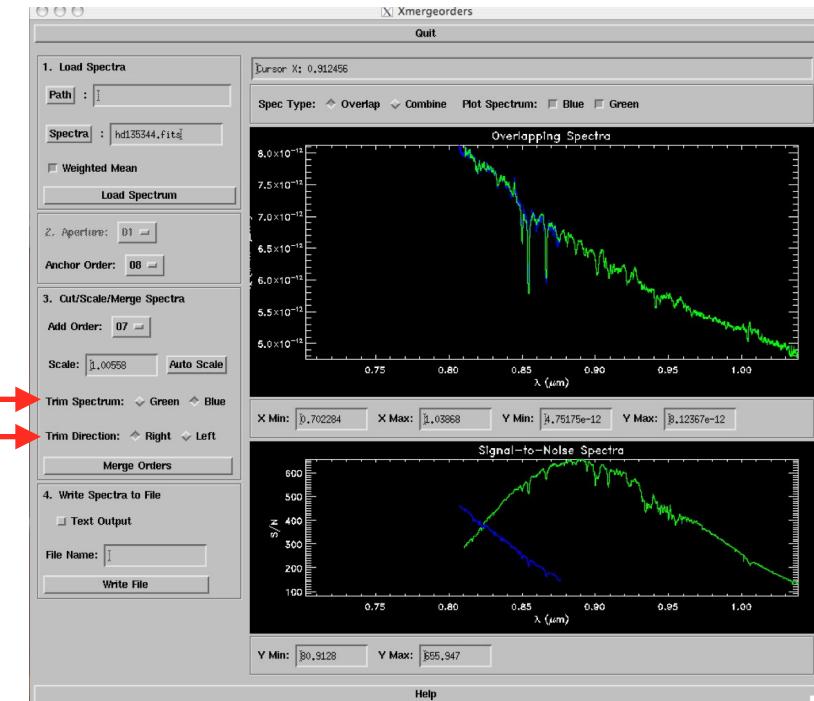
Before



After

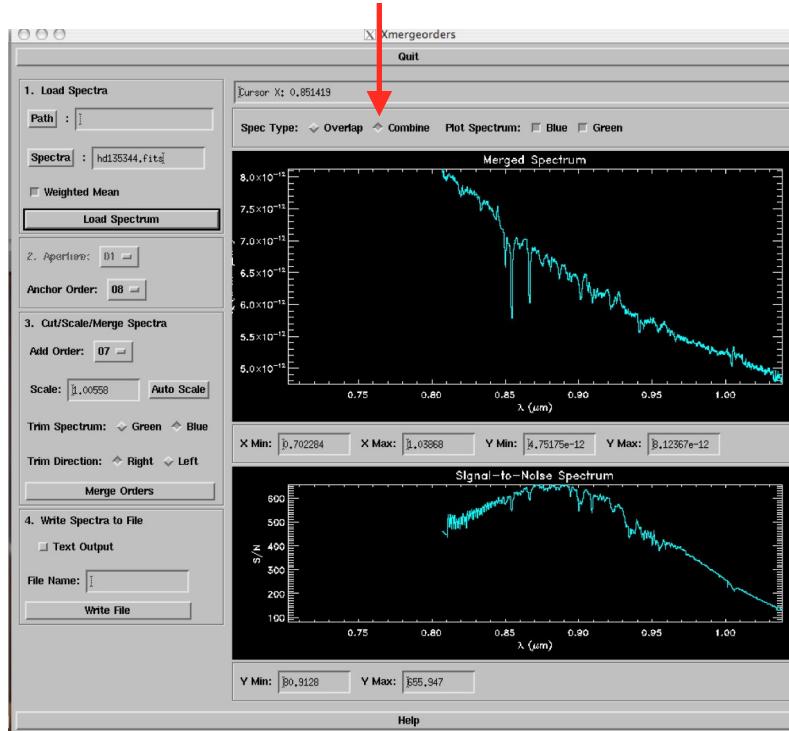
It's hard to see, but the green spectrum (Order 7) has moved just a little bit.

It is also apparent that the noise in Order 8 is a bit more than in 7 in the wavelength range where they overlap. You can see this in the spectra, and in the S/N plots. So we will trim off the noisy part before merging.



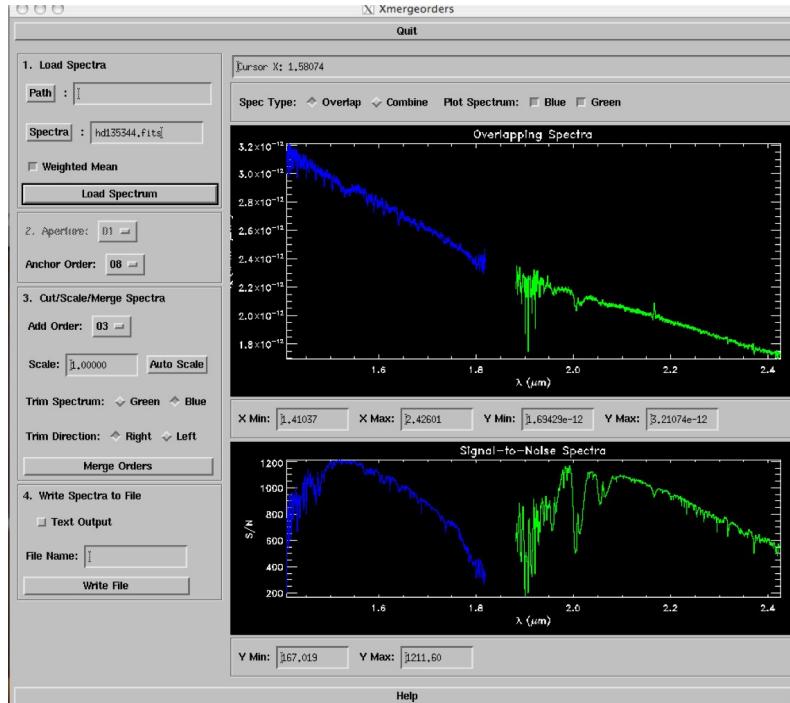
Here, I selected trimming the blue spectrum to the right of wherever I left-clicked on the spectrum.

Then, take a look at what the merged spectrum will look like by hitting the combine button:



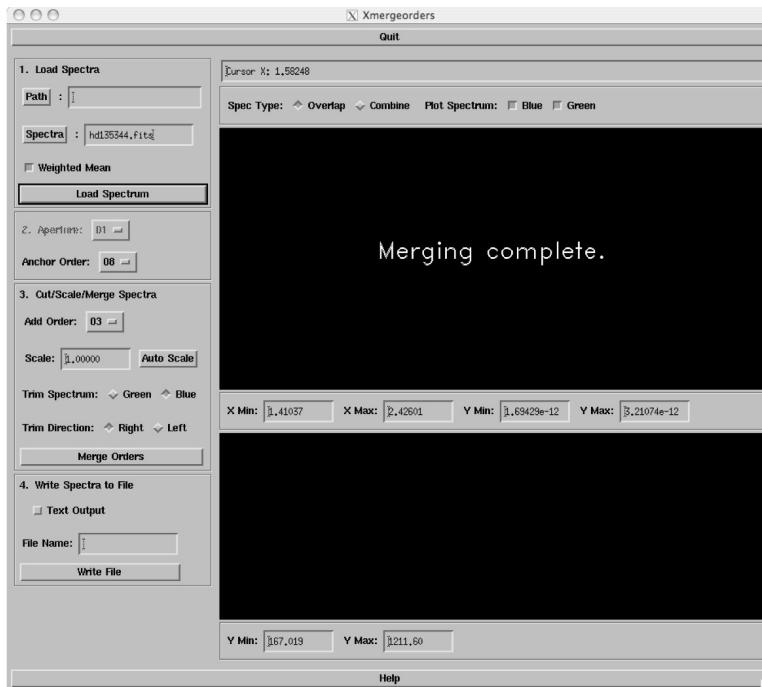
When you are satisfied, hit **Merge Orders**.

Repeat this process for the rest of the orders, but **do not change the Anchor Order**, just the Add Order. Otherwise, the program thinks you are starting all over, and will forget what you have done up to this point!



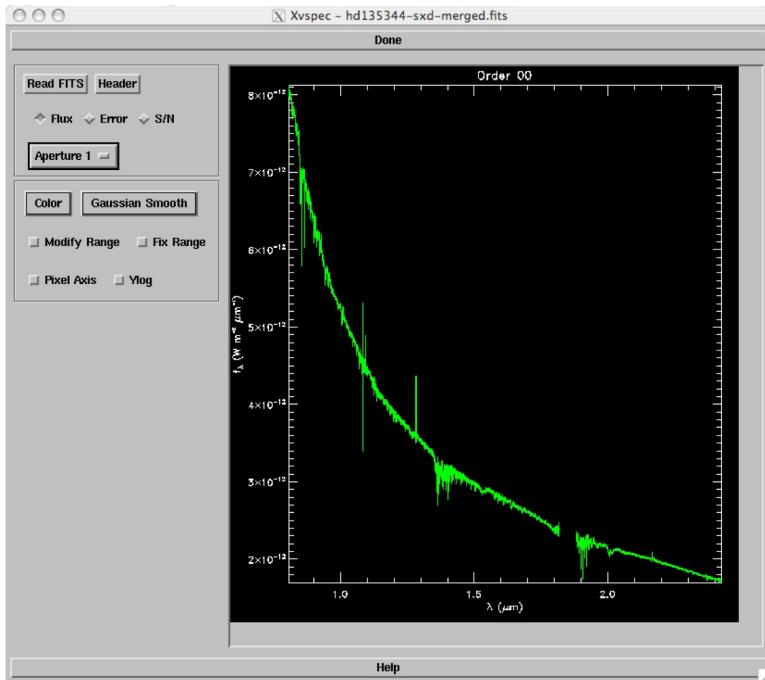
Now, orders 3 and 4 do not actually overlap, so it will require a bit of guesswork as to how much adjusting is needed. Most of the orders don't require much, so you might even leave this one alone. Or you can type a number into the scale box and hit return.

When all orders have been merged, you will be told that the merging is complete:



Pick a file name to write the result to and hit Write File. The file should be named with the star name, SXD or LXD, and "xmergeorders". Like "V1247Ori_SXD_xtellcor" or "V1247Ori_LXD_xtellcor". The files should be saved to the output folder.

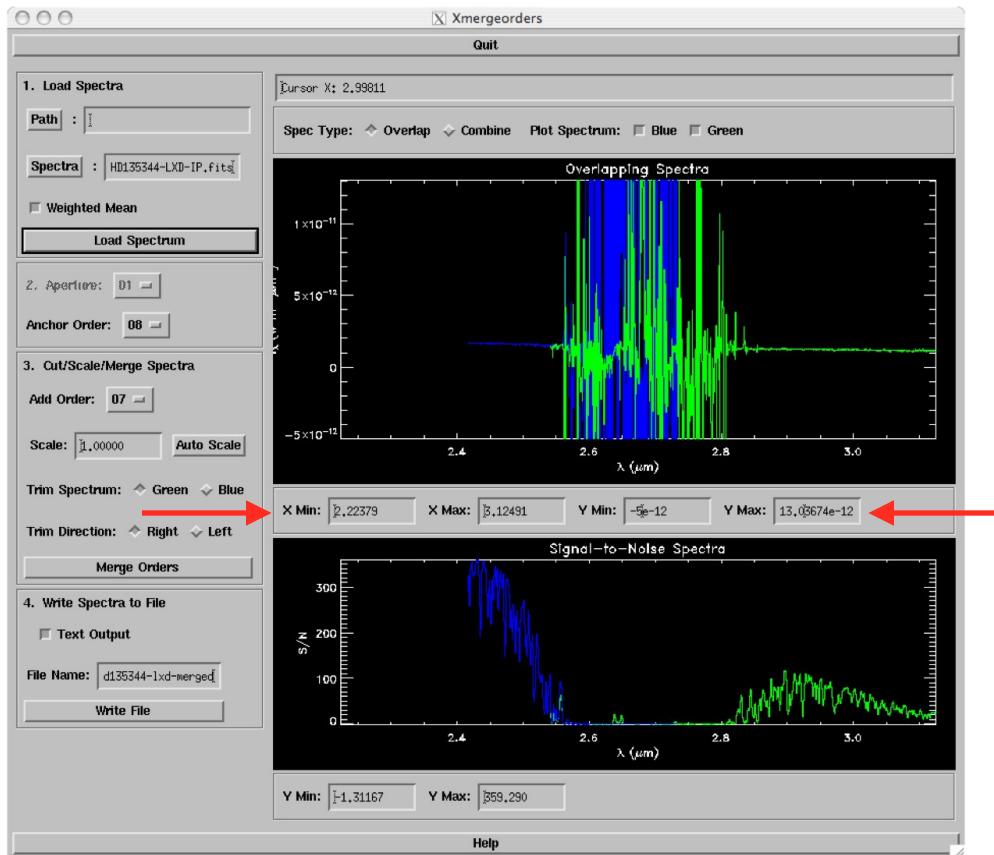
The program also shows you a plot of the final product:



Hopefully all of your spectra will look wonderful!

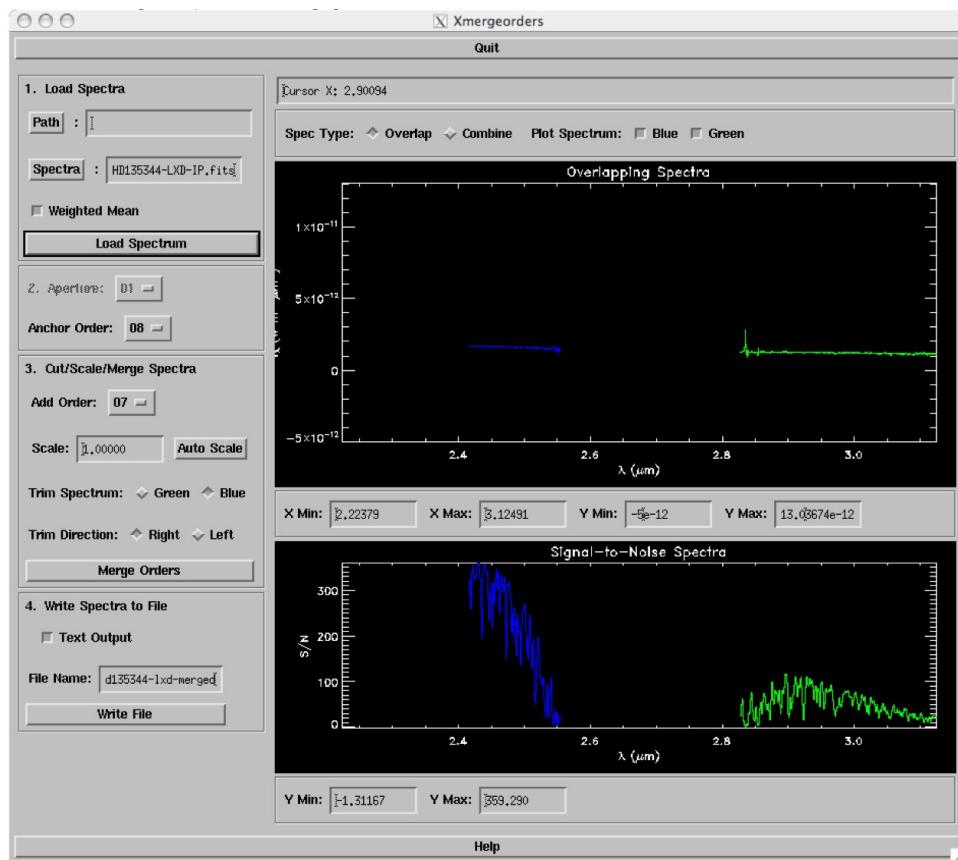
Merging Noisy LXD Data

First, re-scale the plots manually (fill in & hit return for BOTH entries):

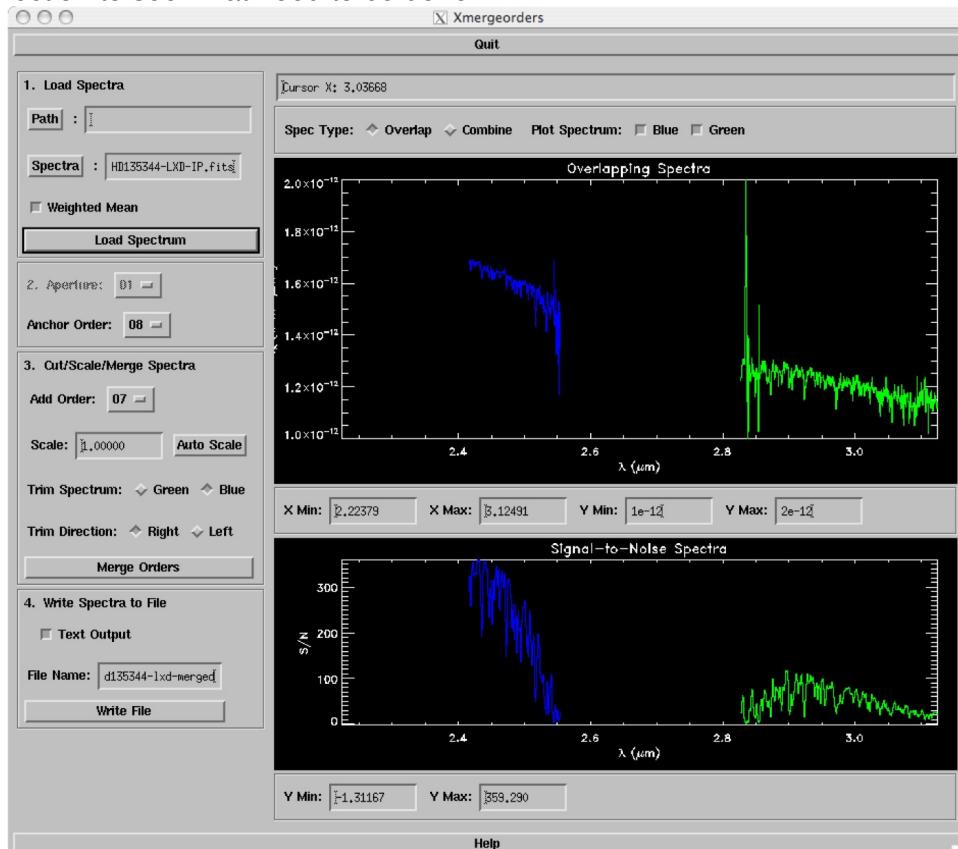


Using the Auto Scale here is pointless – it will be trying to normalize noise to noise, so you will have to use the Manual scaling here (when in doubt, use 1.0).

Then cut out the bad spots for the blue and the green (set green, cut right, then click on the cut point with the mouse; Then set blue, cut right, click on the cut point). Generally you can cut the data out that has much lower S/N of the two sets:

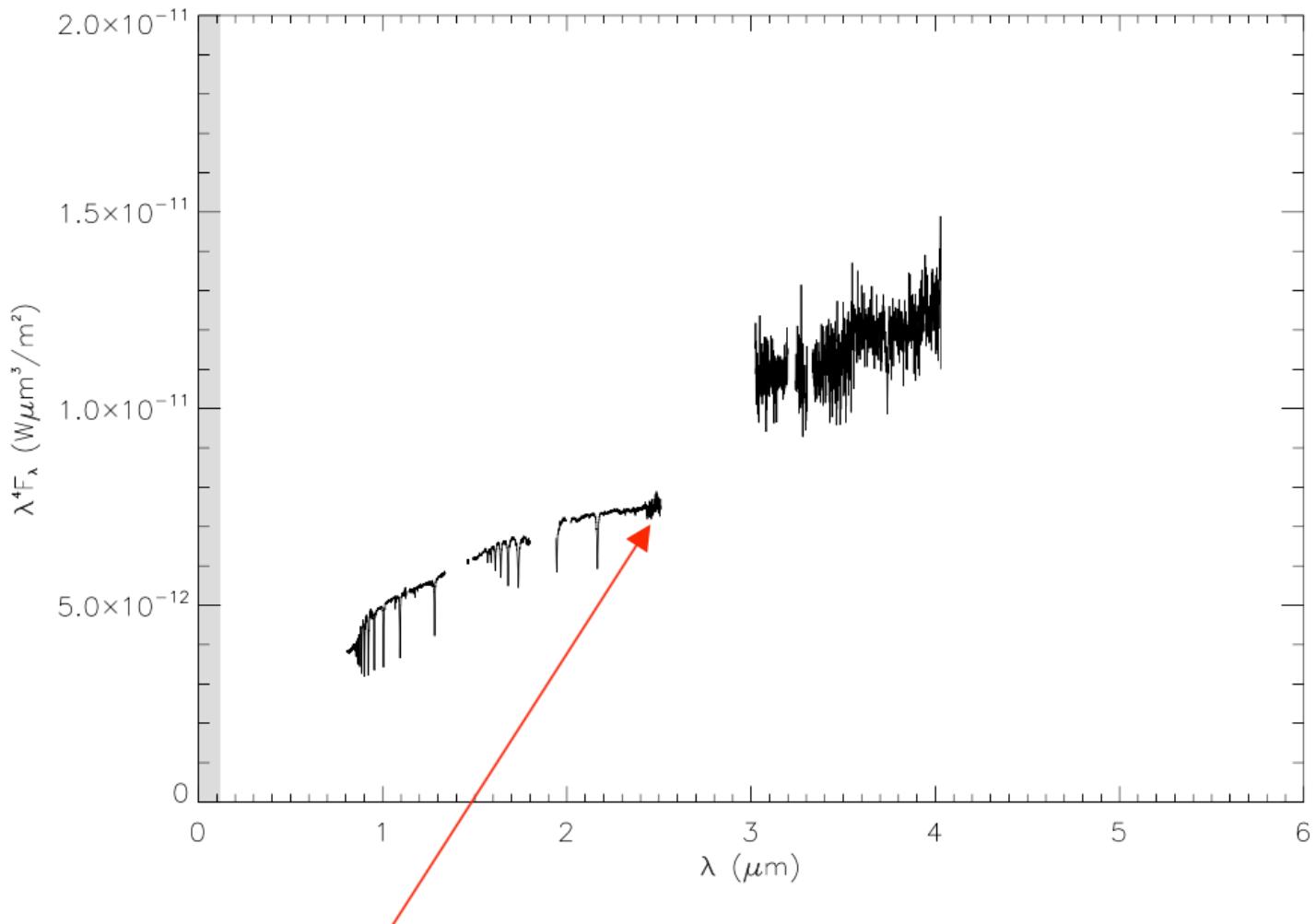


Then expand in Y direction to see what need to be done:



Reality Check

Now, as a “reality check”, it is useful to read the FITS file and make a plot λ of $F(\lambda) * \lambda^4$ or something close to that. Because at these wavelengths most stars are close to being blackbodies in the Rayleigh-Jeans tail (where $B(\lambda) \propto \lambda^{-4}$), this “flattens” the data out, making scaling errors much easier to see. Here is an example of an A0V star after the joining of LXD to SXD in xmerged (a later step):

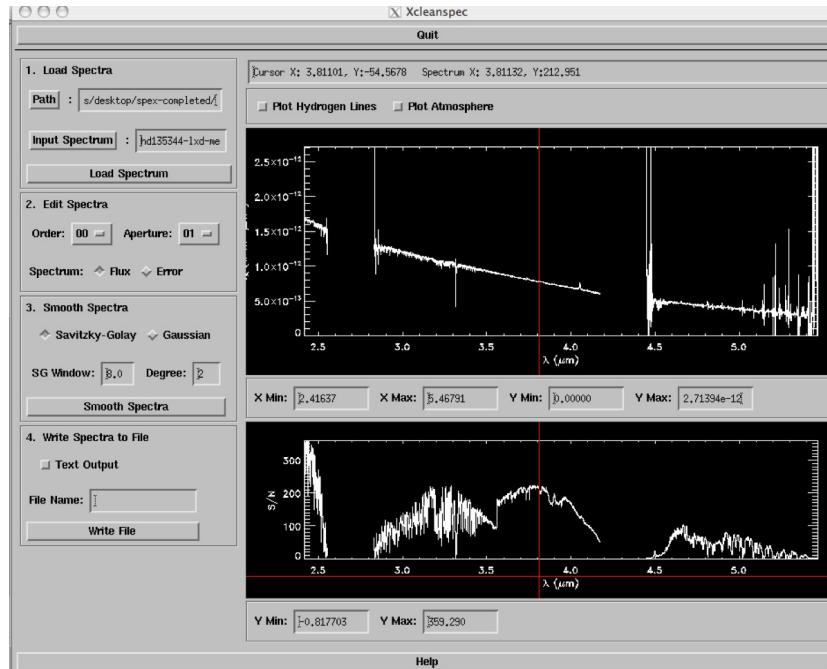


The jump between the tiny portion of LXD data near $2.4 \mu\text{m}$ and the longward of $3 \mu\text{m}$ would have been very tough to see without doing this exercise. I show this at the **xmergeorders** stage because you may need to fix your data **here before** proceeding to **xcleanspec** and **xmerged**. This is the difficulty in merging data (which may have a spectral slope) when the area used to do the scaling is nothing but noise.

XCLEANSPEC

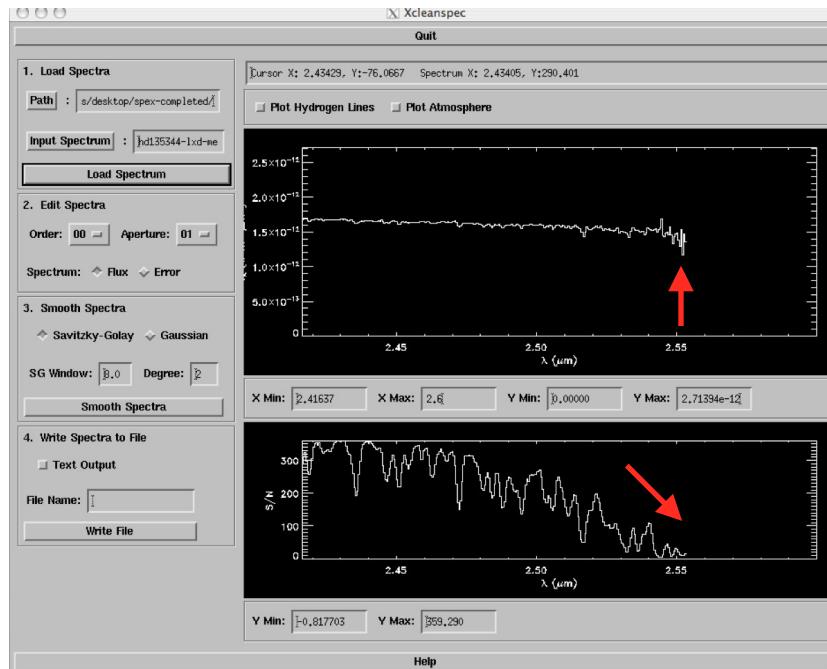
Many spectra, especially LXDs, will be noisy, especially in regions where telluric features are strong. One can snip out bad parts, fudge in straight line, and smooth them.

In the IDL prompt, type `xcleanspec`, and hit return.



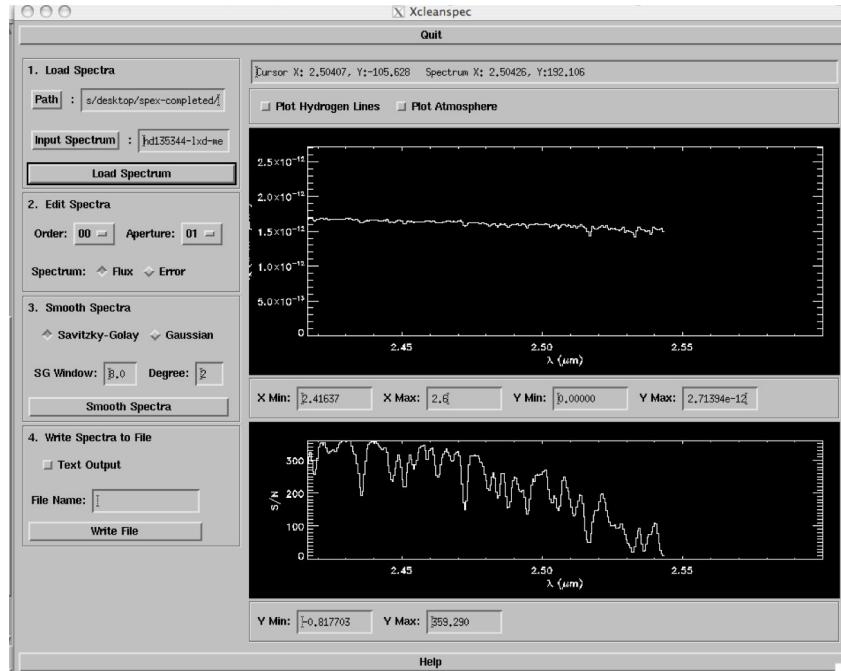
Here I loaded an LXD, but the autoscaling used the noise to set the y-range, so I manually adjusted it here to increase the scale by 10x – so I could SEE what I was doing!

In order to see things up close, I decided to expand the wavelength range around the region of interest:



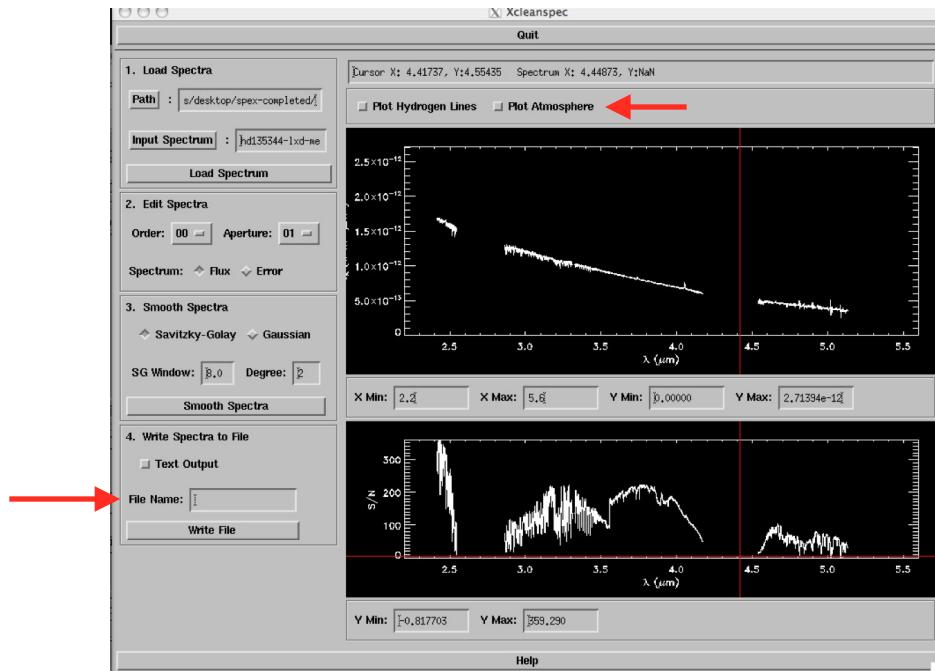
Here, it is apparent that the increased noise in the spectrum is accompanied by a drop in S/N.

I decided to snip out the “bad” stuff using “r” (“remove”) instead of “f” (“fix”) and left-clicking on the 2 wavelengths where I was snipping:



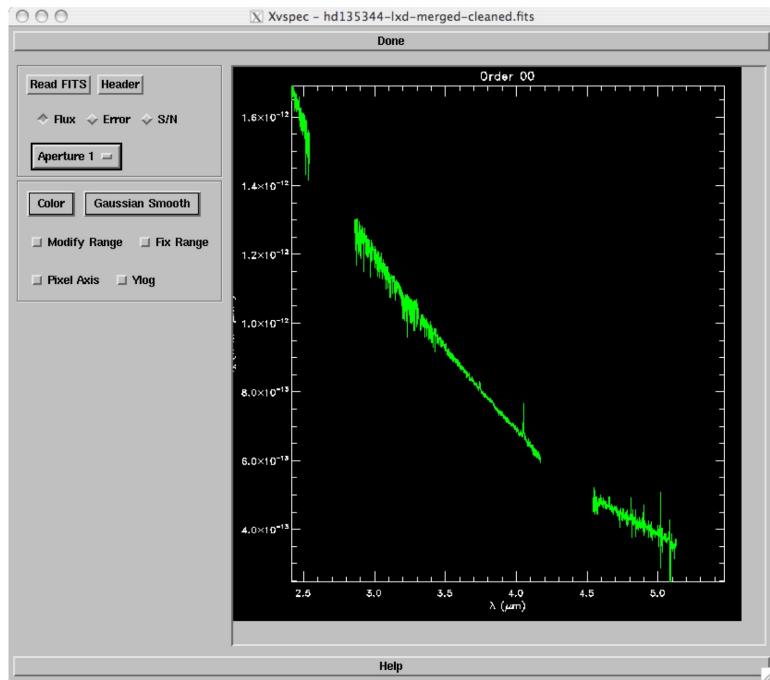
This neatly clipped the bad stuff out. I liked the result, so I hit “s” for “save”. **NOTE: You MUST do a save after every individual edit, or they will be lost!** Also note that the “meaning” of “s”: has changed from earlier routines of Spextools, where “s” meant “select”.

I went along snipping away, until I saw:



Now, at this point, one can smooth the spectra (follow along the Help File shown at left). But I thought this was pretty good, so I decided on a file name, and just saved the result.

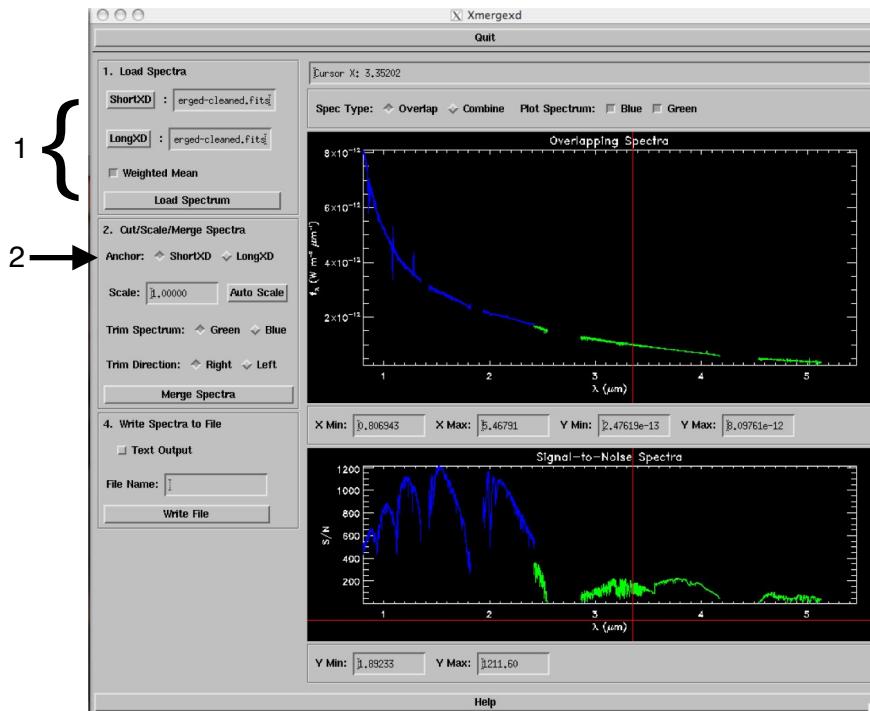
Note: You can also see the location of Hydrogen lines or the telluric features by clucking on these boxes. This is often very useful (especially in the LXD data) to know when features are real, and when they are telluric – both noise and systematics (airmass....)



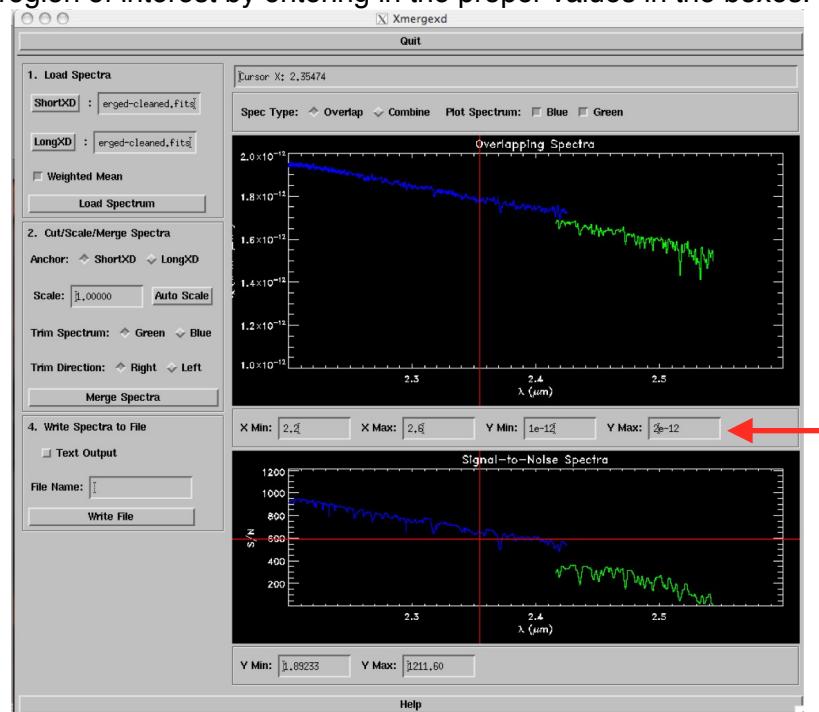
XMERGEXD

This will allow you to merge SXD and LXD data.

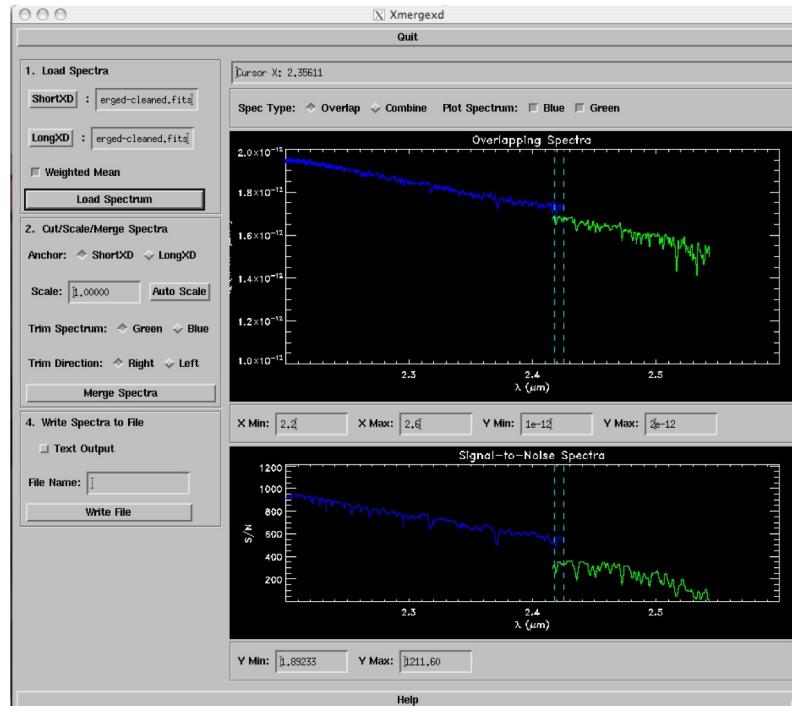
In the IDL prompt, type *xmergexd*, and hit return.



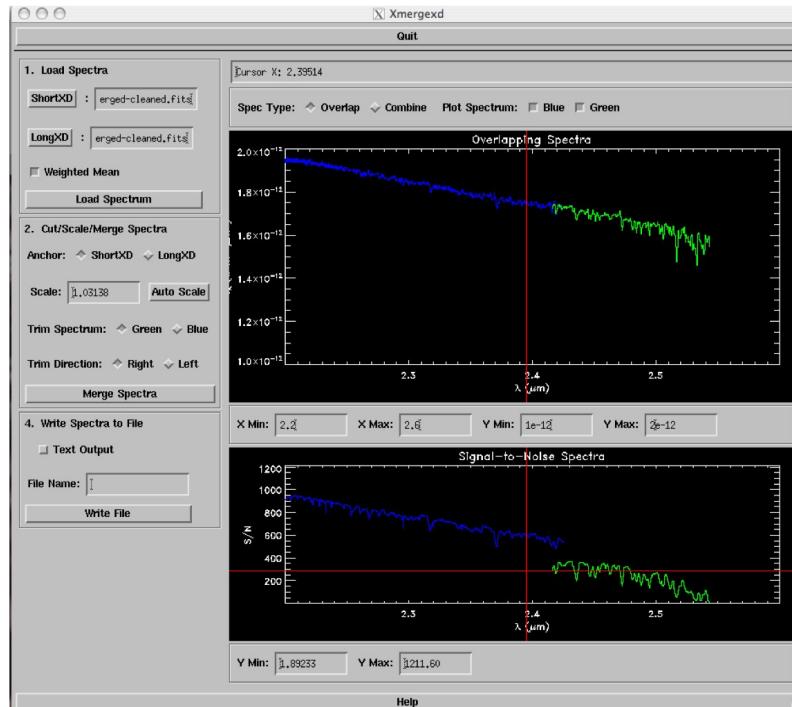
1. Load the SXD and LXD data, and press “Load Spectrum”
2. You will also need to decide which section to anchor. I like picking the one with the best S/N, which will usually be SXD.
3. Here, zoom in on the region of interest by entering in the proper values in the boxes.



4. Use “s” to select the wavelength region to do the scaling, and left-click on the ends of the wavelength interval over which to normalize. Then click “Auto Scale”.

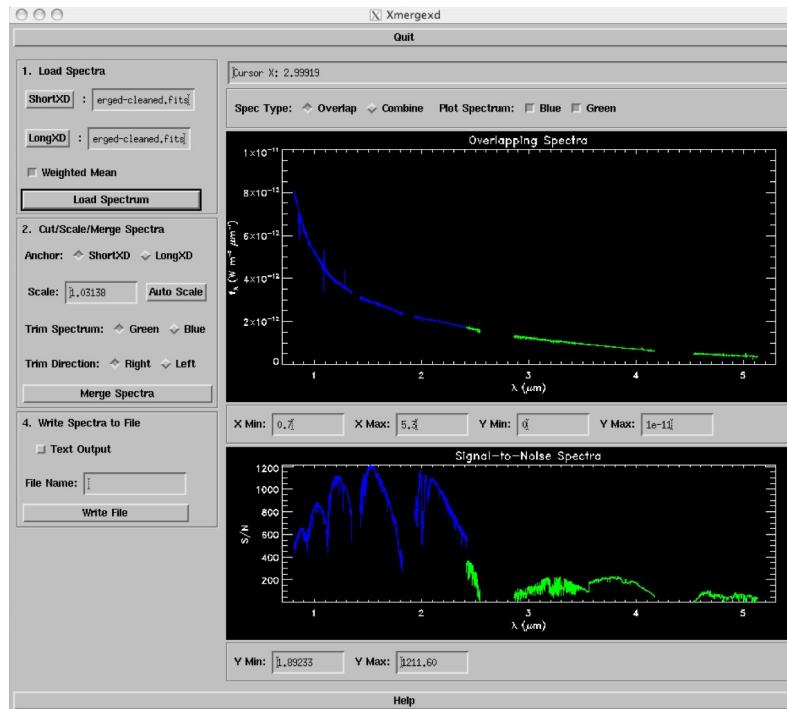


Auto scaling will bring the LXD into agreement with the anchored SXD.

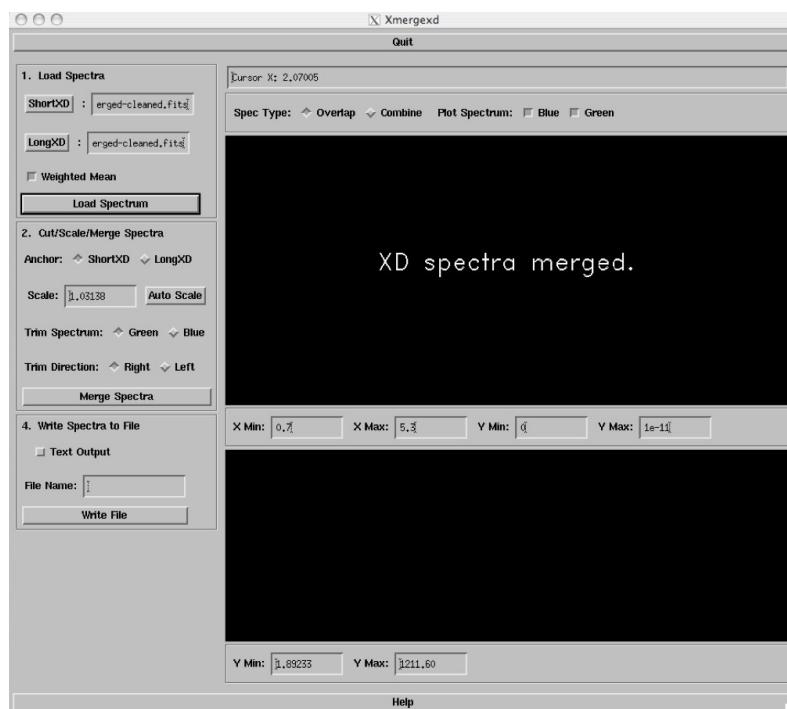


If you don't have any overlap between the two sets of data, **Autoscale** will fail. You must select a portion of one of the data sets, and manually rescale by filling in the value you deem appropriate in the box, and hitting return.

Now, back to look at the whole spectrum.

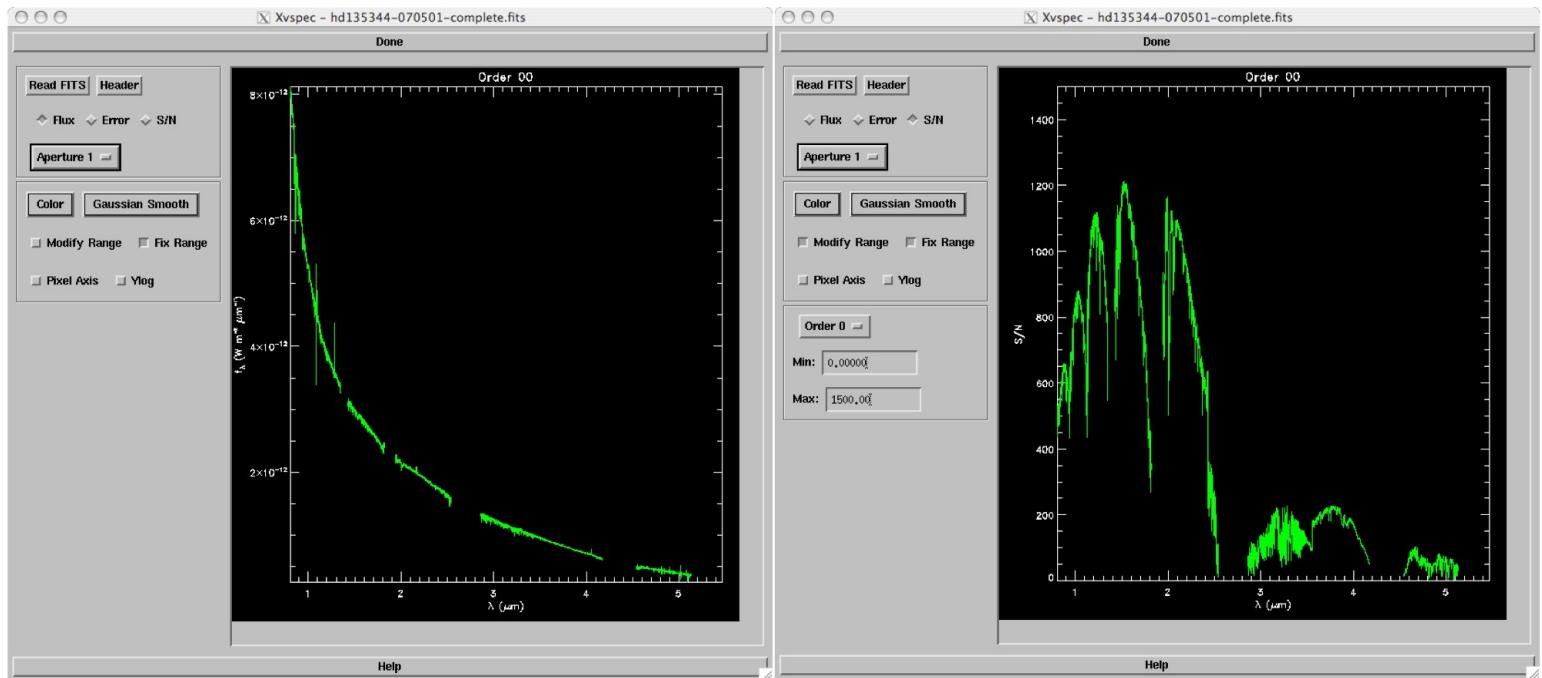


Once everything looks good, press “Merge Spectra” in section 2. Unfortunately nothing exciting appears.



Pick a file name to write the result to and hit Write File. The file should be named with the star name, SXD or LXD, and “xmergexd”. Like “V1247Ori_SXD_xmergexd” or “V1247Ori_LXD_xmergexd”.

Note: because no path name is entered in **xmergexd**, the files get written to your home directory (wherever that may be). Copy that data into the “Data” folder for the star for future use.

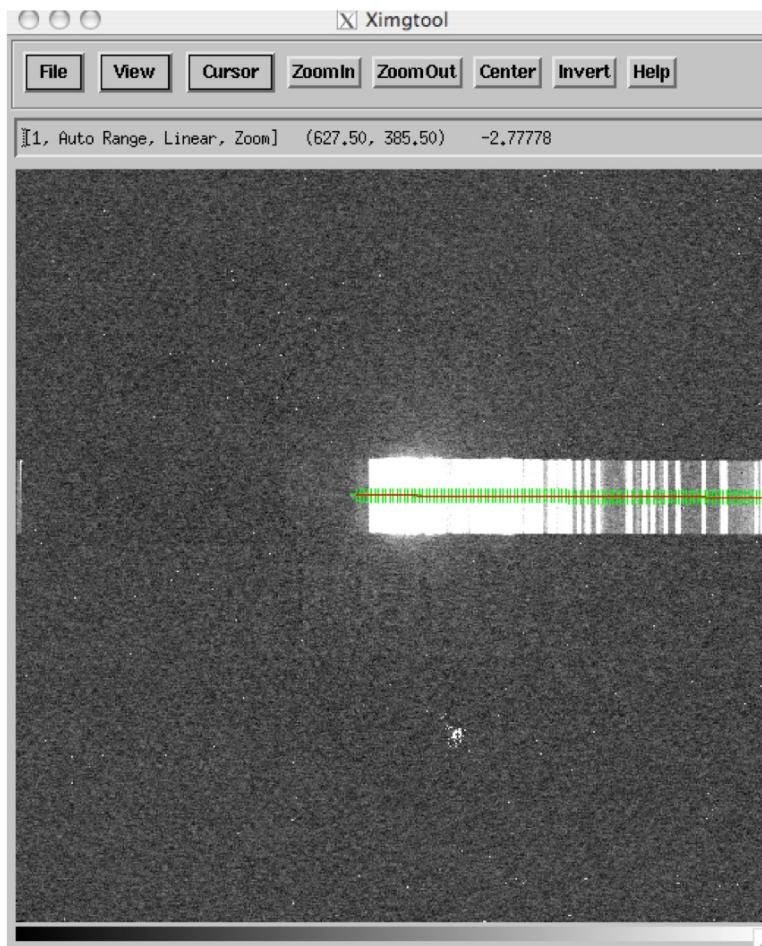


Congratulations you are done!

Handling Prism Data

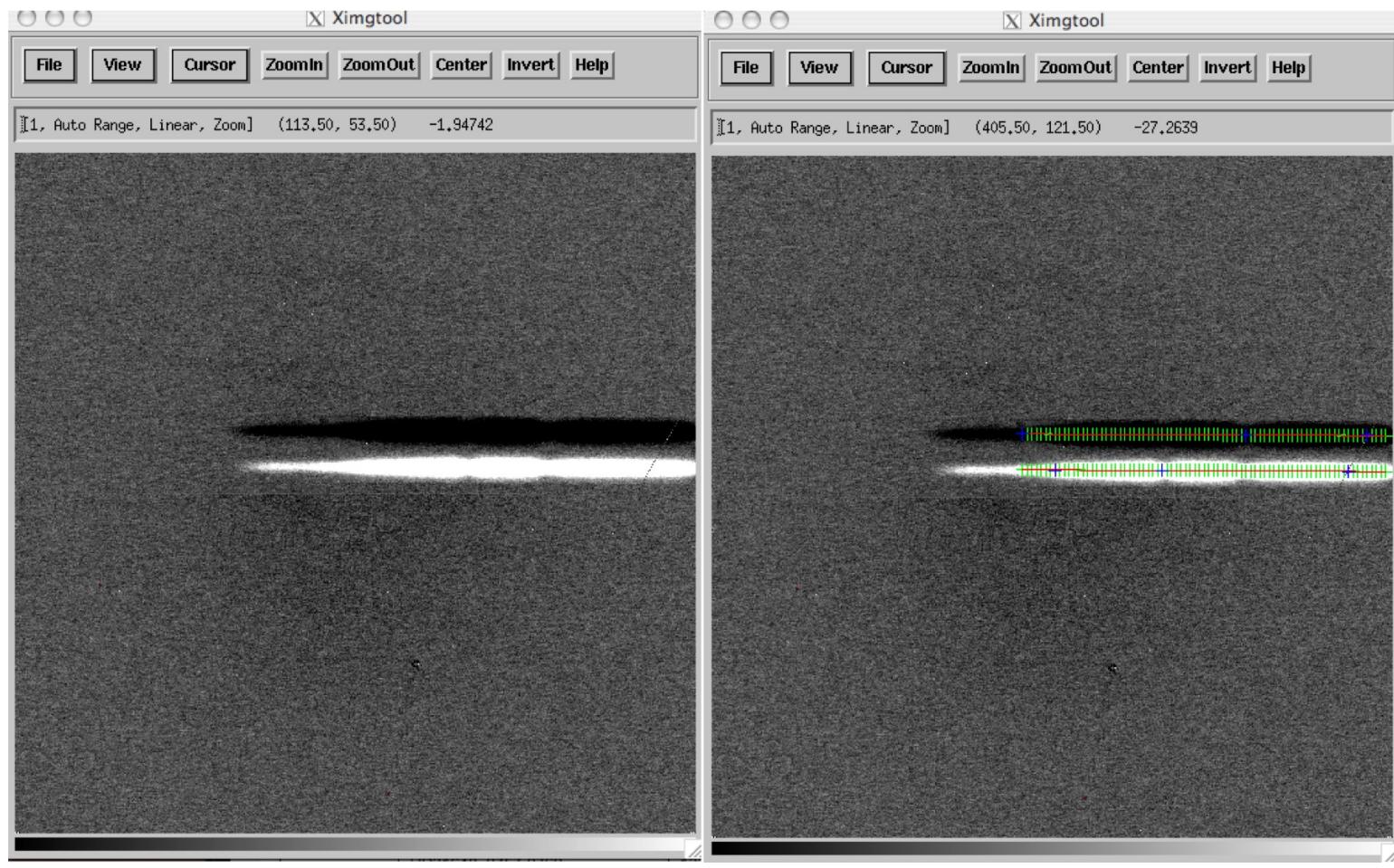
Most of procedures used to reduce the echelle data apply to data obtained with the prism. However, some differences in appearance (if not procedure) exist, and these will be described here.

XSPEXTOOL

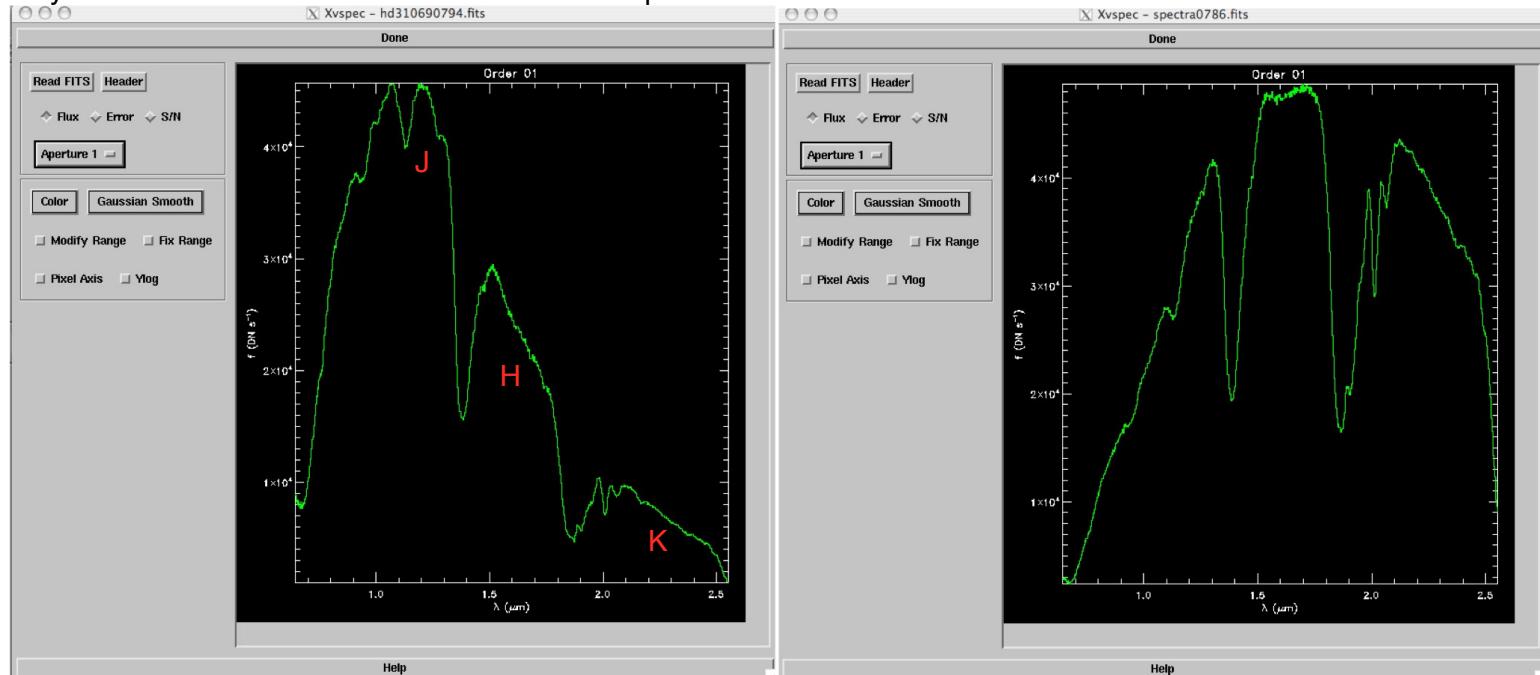


The first thing is obvious – there is only one “order”! Now, I have used the prism with a very wide - 3.0 arcsec – slit in an attempt to get something closer to “absolute flux levels” than one might get with a significantly narrower one. However, flats obtained through such a wide slit saturate, making them useless. Arcs are also rendered useless, because the lines are too wide and blended for a proper wavelength calibration. So I have done the cal macros with a narrower slit (say 0.8 arcsec) than the star observations. This is sort of a cheat, but it does seem to work.

Xspextool does a fine job in locating the spectra and tracing them.



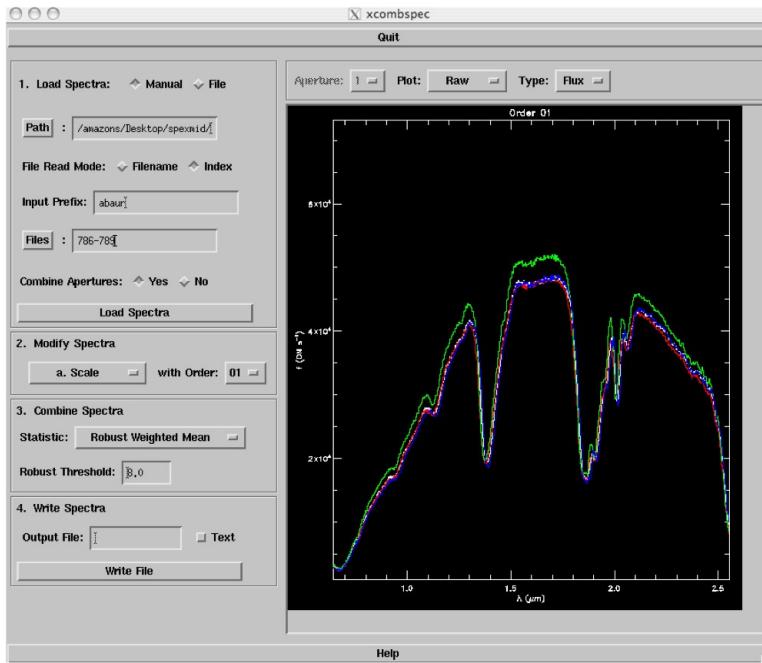
After doing the extraction you can see the low-resolution spectra, and simultaneously see all of the main telluric features. Here you can also see why the wavelengths of the photometric J, H, and K filters were defined the way they were – to avoid telluric features as much as possible.



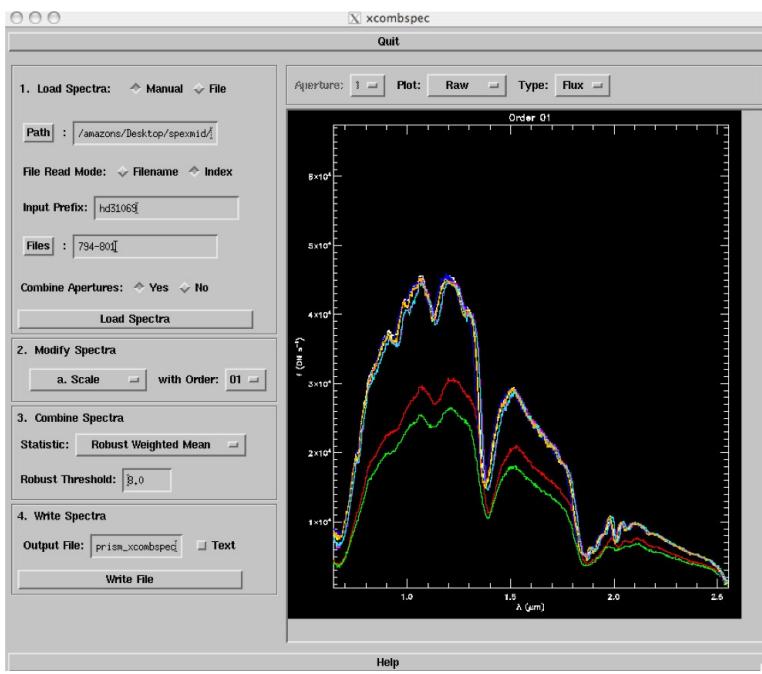
Here you can also see a significant difference between the spectra of the two stars shown here. The one on the left is HD 31069, an A0V calibrator use for this set of observations. On the right is AB Aur, an A0V pre-main

sequence star with a disk of warm dust surrounding it. This leads to a significant “infrared excess” that causes it to be systematically brighter at longer wavelengths than a “normal” A0V star.

XCOMBSPEC

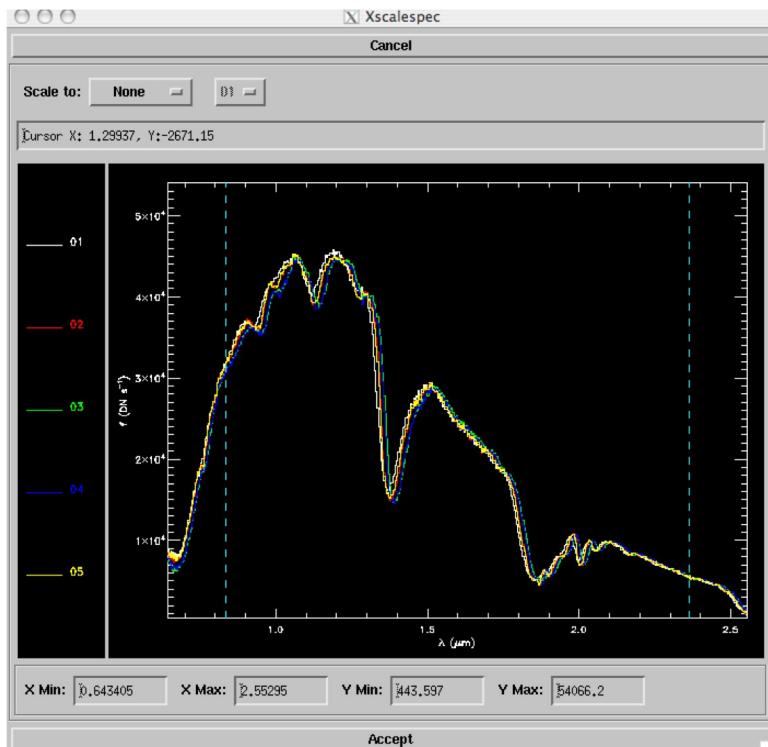


4 observations of AB Aur obtained with the 3.0 arcsec slit. Three reproduce each other nicely, with one a bit higher, perhaps during a moment of improved seeing and better light throughput.



The observations of HD 31069 were not as good. Two are obviously lower, most likely due to poor tracking, allowing half (or more) of the light to fail to make it through the slit. If you look carefully, you can see a slight shift in wavelength of the two low spectra. Remember that the spectrum is actually an image of the star occulted by the slit jaws, dispersed in wavelength.

Here one can see a wavelength shift even in the “best” exposures:



This will tend to smear out the features used in the telluric correction and introduce some distortion in the resulting spectrum of the science target.

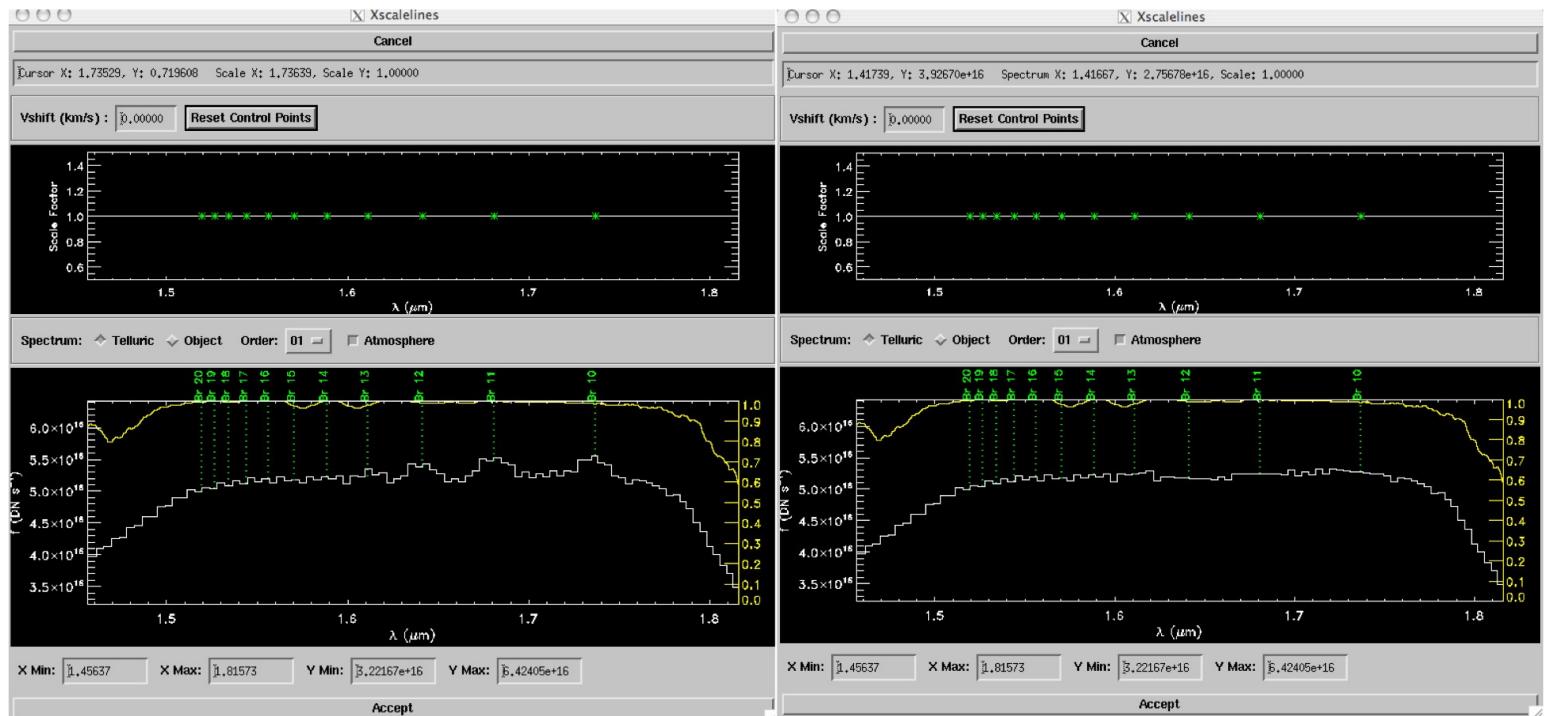
XTELLCOR

For this situation, especially where the stellar observations are so smeared out, only the “IP” option seems reasonable in Xtellcor. Using this option and going on to Construct Telluric Spectra and Scale Lines:

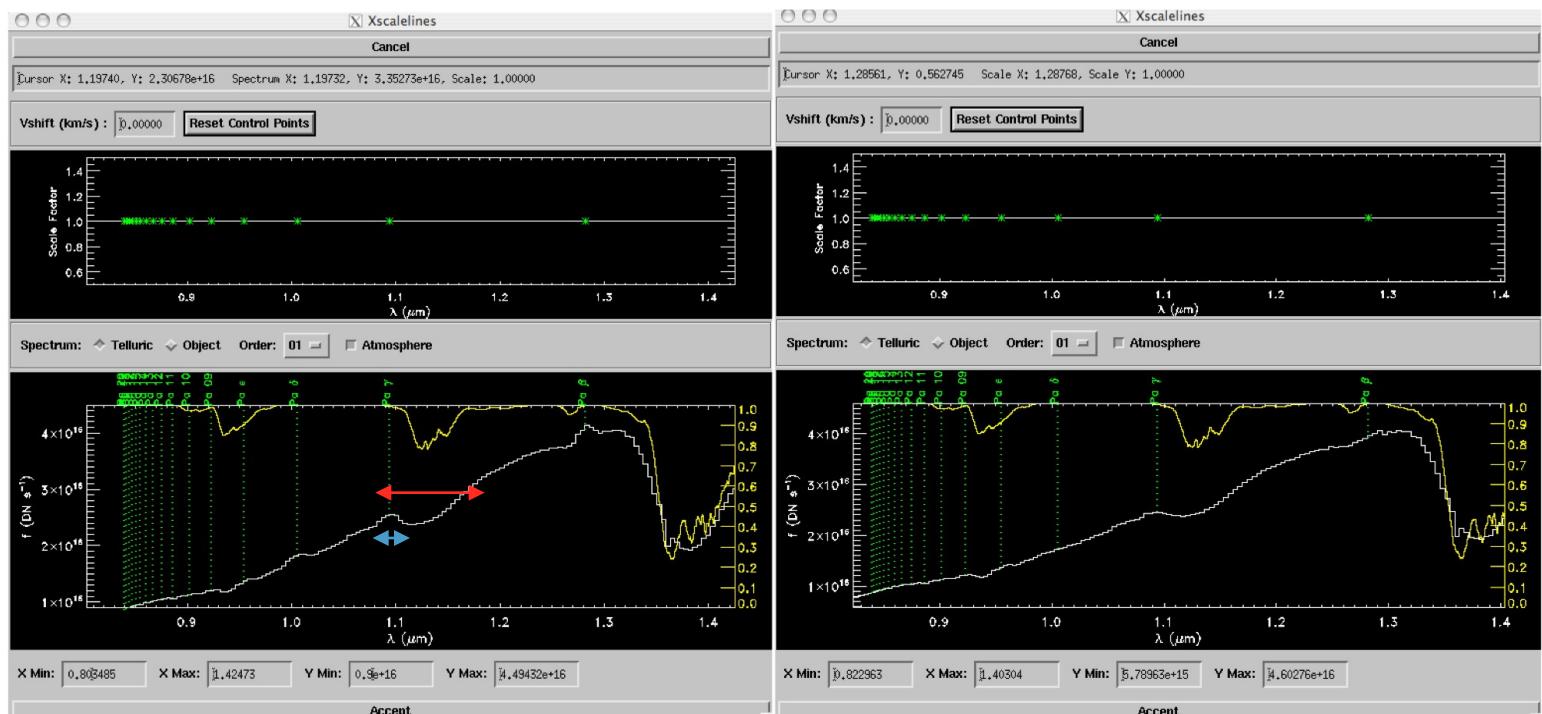


There are still some residuals to deal with.

Here we see the region in the H-band before (left) and after (right) using the “f” technique to correct the residuals:



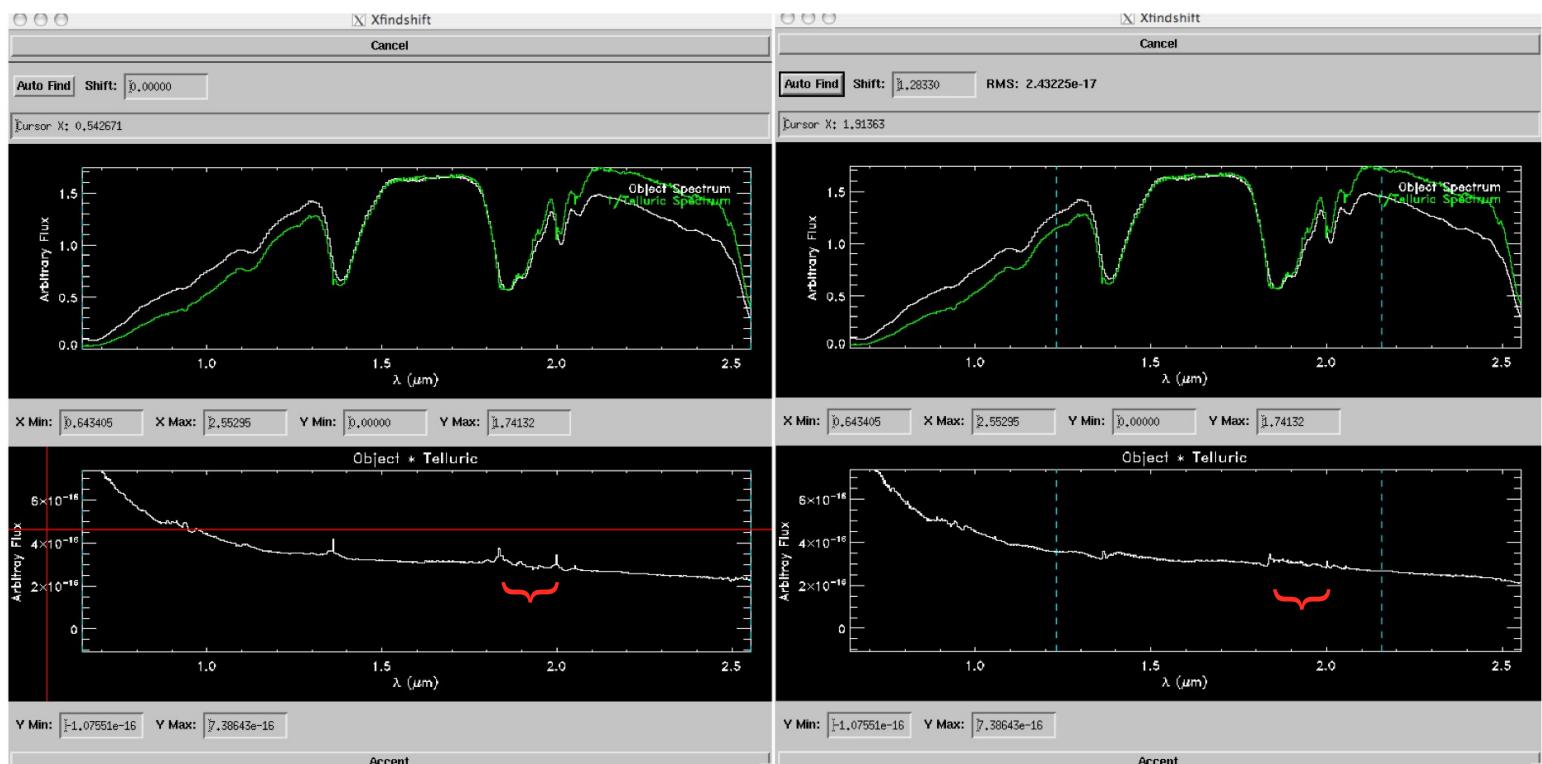
Pa y:



Before and After

In this wavelength region, one has to be very careful as the Paschen lines may border on the telluric features. The idea is to remove the Paschen lines with the least effect on the telluric lines. You don't want to remove those! Here I show in blue the extent of the Pa_y line that you could smooth over. Do NOT correct over the wavelength range of the red arrow, or you will remove the telluric band. Remember, the goal is to make the A0V spectrum resemble the telluric one as much as possible.

Wavelength Shift:



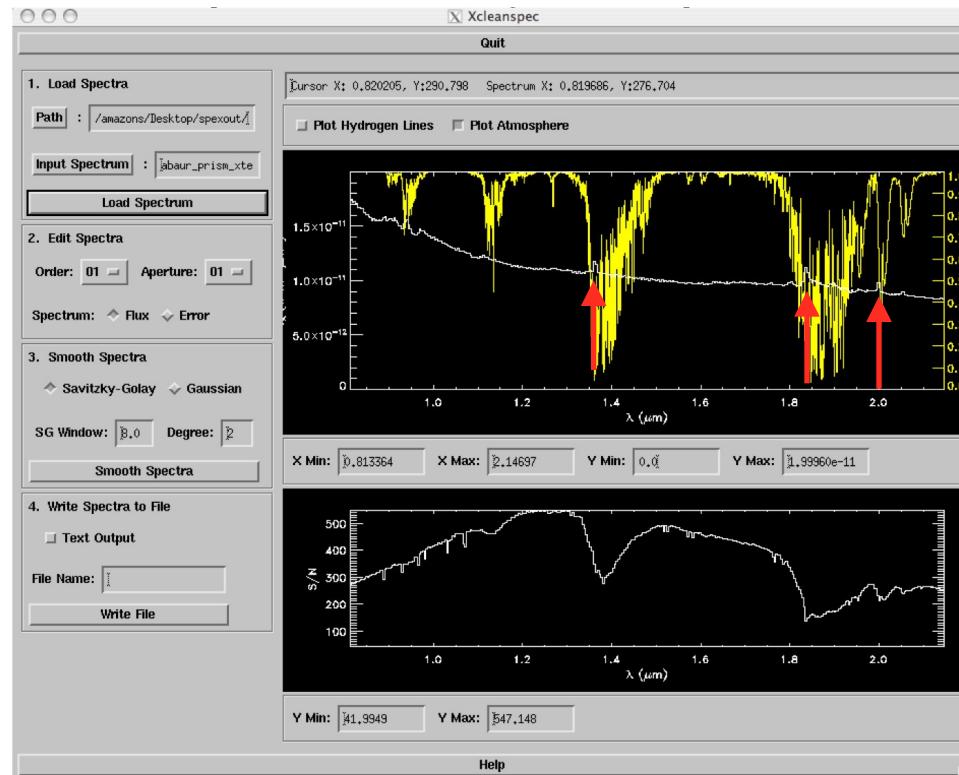
At the left is zero wavelength shift. At right is applying the Auto Find to the region between the dashed lines. Note the slight offset in height near 1.54 microns. This offset can actually be improved by doing the shift manually. Enter a value in the box, hit return, and see if it improves or gets worse. Iterate until you get the best result for the entire spectrum.

XMERGEORDERS

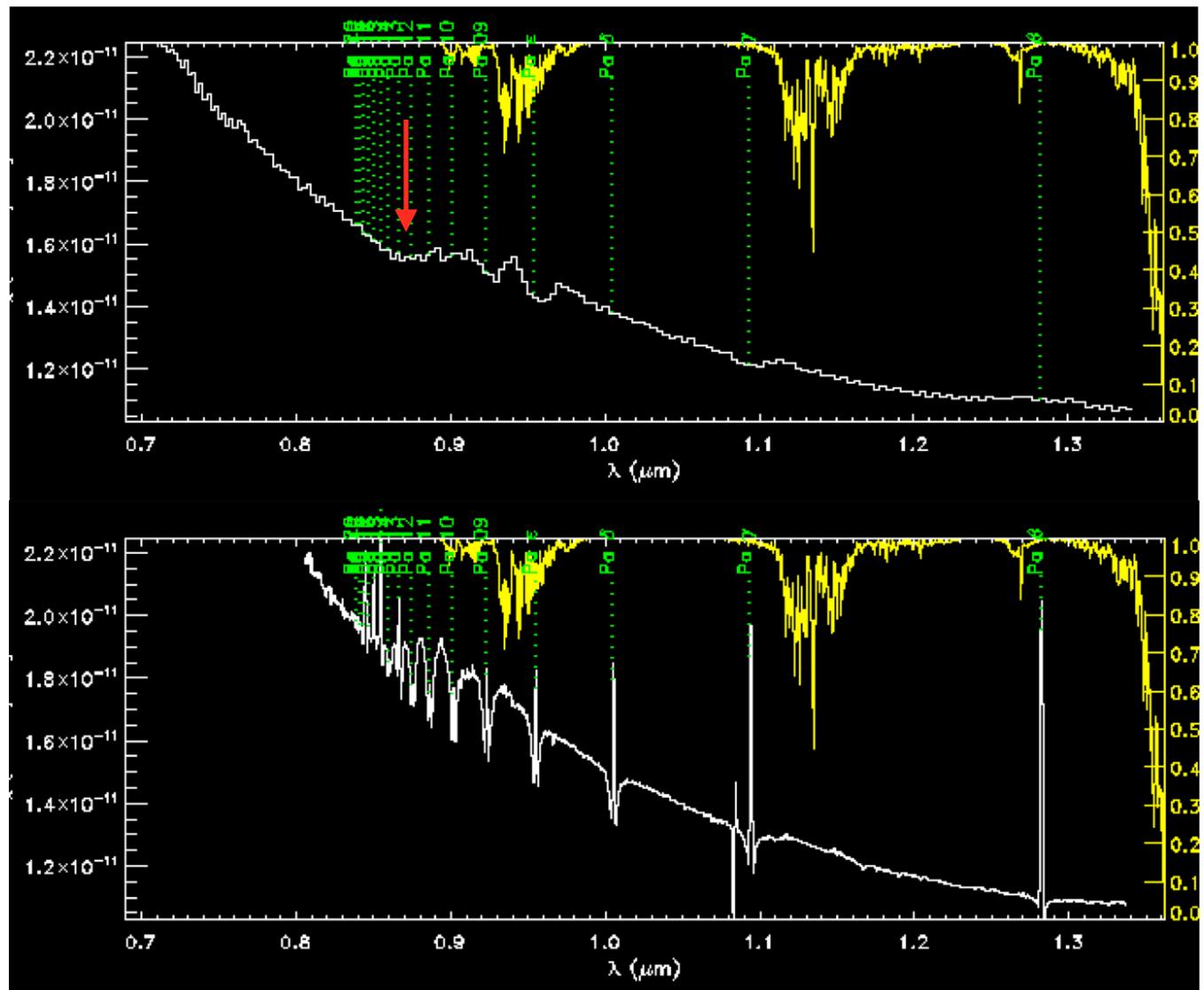
Skip, because there are no orders to merge!

XCLEANSPEC

Here we can see that some of the vertical “spikes” coincide with the strongest telluric absorption features. These can be snipped.



The region between 0.85 and 1.0 μm is sort of a mess. Here we have a confluence of photospheric absorption bands, Paschen and the Ca II triplet both in emission, as well as one of the weaker telluric bands. The “kink” here is real!



Compare the Prism spectrum (above) with the SXD spectrum (below).

In my case, I am looking for the overall spectral shape and level; I am not trying to measure line properties! So at this point I would probably just ignore things and not remove any more data.

XMERGEXD

Skip, because there are no orders to merge!

Brandon's IDL Manual 1.0

I've automated all the procedures for plotting and extraction in IDL. The hardest part is getting all the path names right. If you follow the naming and folder structure that has been / will be laid out, it shouldn't be too difficult. The first part of the manual needs to be repeated for each set of data for the star individually. At the end, all the data sets for a particular star will be merged together.

If some of the programs used here don't work, don't hesitate to contact me. (brandon.mcquilkin@gmail.com). I will try to fix the issues. Or if you need a new feature, I will try to add it.

Fitting with Mathematica

Part of the analysis is figuring out various things about our star (like temperature.) To do that we need to fit a theoretical model to our star's spectra. This can be done with IDL, but it takes a good hour or two to do. It can be done in about 10 minutes using Mathematica. (Plus it is much more straightforward this way.)

Exporting Data for Mathematica

Sadly Mathematica cannot import the data files that we use in IDL. We need to change those formats (fits, sav) into something Mathematica can read (csv.)

To convert your data use m13fitstocsv. If you get an error that this function does not exist make sure that you have the McQuilkin library installed correctly.

Repeat this process for the standard star's data, and all target and prism data sets.

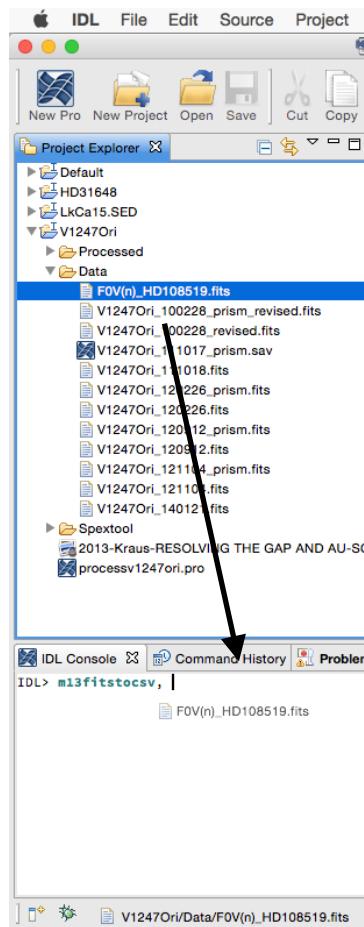
If you have a fits file:

In the console:

1. Type the function name: "m13fitstocsv" then a comma. If the function exists it will turn turquoise.

```
IDL> m13fitstocsv,
```

2. Enter the absolute path to the file in single quotes next. If you have your workspace set up properly, you should just be able to drag the file in project bar into the console to insert the path. If you do not have a fits file, just insert an empty pair of single quotes.



```
IDL> m13fitstocsv, '/Users/Brandon/Documents/School/Sitko/Research/Workspace/  
V12470ri/Data/V12470ri_111018.fits',
```

3. Next enter the export path for the converted data. It should go into a folder for that particular data set under the “Processed” folder. I would drag that folder in, then type the file name.

```
IDL> m13fitstocsv, '/Users/Brandon/Documents/School/Sitko/Research/Workspace/  
V12470ri/Data/V12470ri_111018.fits', '/Users/Brandon/Documents/School/Sitko/  
Research/Workspace/V12470ri/Processed/V12470ri_111018/V12470ri_111018.csv',
```

4. Now we need the data scale. Enter 1.0. If you need the SXD or LXD data scaled, and you know the scaling, enter it here. 99% of the time it will be 1.0.

```
IDL> m13fitstocsv, '/Users/Brandon/Documents/School/Sitko/Research/Workspace/  
V12470ri/Data/V12470ri_111018.fits', '/Users/Brandon/Documents/School/Sitko/  
Research/Workspace/V12470ri/Processed/V12470ri_111018/V12470ri_111018.csv',  
1.0
```

5. Hit return, the program will run and export the data to the desired folder.

If you have a sav file:

In the console:

1. First we need to import the save file to see what we have. To do that we will use the restore function. The restore function is very straight forward. Give it the absolute path to a sav file, and add the “verbose” flag to see what variables are loaded.

```
IDL> restore, '/Users/Brandon/Documents/School/Sitko/Research/Workspace/  
V12470ri/Data/V12470ri_111017_prism.sav', /verbose
```

Once you hit return you should see something like the following:

```
IDL> restore, '/Users/Brandon/Documents/School/Sitko/Research/Workspace/  
V12470ri/Data/V12470ri_111017_prism.sav', /verbose  
% RESTORE: Portable (XDR) SAVE/RESTORE file.  
% RESTORE: Save file written by sitko@Mike-Sitkos-Mac-Pro.local, Thu Feb 16  
18:16:04 2012.  
% RESTORE: IDL version 8.1 (darwin, x86_64).  
% RESTORE: Restored variable: W.  
% RESTORE: Restored variable: LFL.  
% RESTORE: Restored variable: ELFL.
```

So the following variables were loaded. “W”, “LFL”, and “ELFL”. These correspond to the wavelength, the flux in λF_λ .

2. Since we have the data already loaded, we don't need (or have) a path to a fits file. So just enter a pair of empty single quotes.

```
IDL> m13fitstocsv, '' ,
```

3. Next enter the export path for the converted data. It should go into a folder for that particular data set under the “Processed” folder. I would drag that folder in, then type the file name.

```
IDL> m13fitstocsv, '' , '/Users/Brandon/Documents/School/Sitko/Research/  
Workspace/V1247Ori/Processed/V1247Ori_111017/V1247Ori_111017_prism.csv' ,
```

4. Now we need the data scale. If you have SXD/LXD data, type “1.0”. If you have prism data, enter the number to scale the prism data to match the SXD/LXD data.

```
IDL> m13fitstocsv, '' , '/Users/Brandon/Documents/School/Sitko/Research/  
Workspace/V1247Ori/Processed/V1247Ori_111017/V1247Ori_111017_prism.csv' , 1.0 ,
```

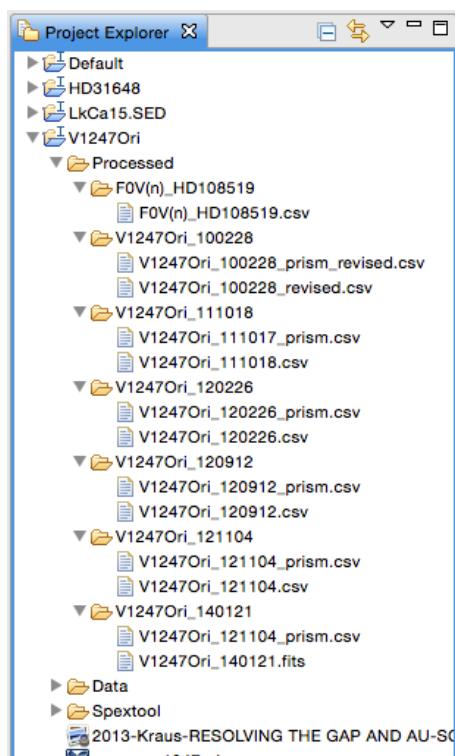
5. Now we need to override the data inputs, since we already loaded the data. To do this we will use the “WOVERRIDE” and “LFLOVERRIDE” flags. To use a flag and set data to it we follow the format:

```
... ,FLAG = variable, ...
```

```
IDL> m13fitstocsv, '' , '/Users/Brandon/Documents/School/Sitko/Research/  
Workspace/V1247Ori/Processed/V1247Ori_111017/V1247Ori_111017_prism.csv' ,  
1.0 ,WOVERRIDE=W ,LFLOVERRIDE=LFL
```

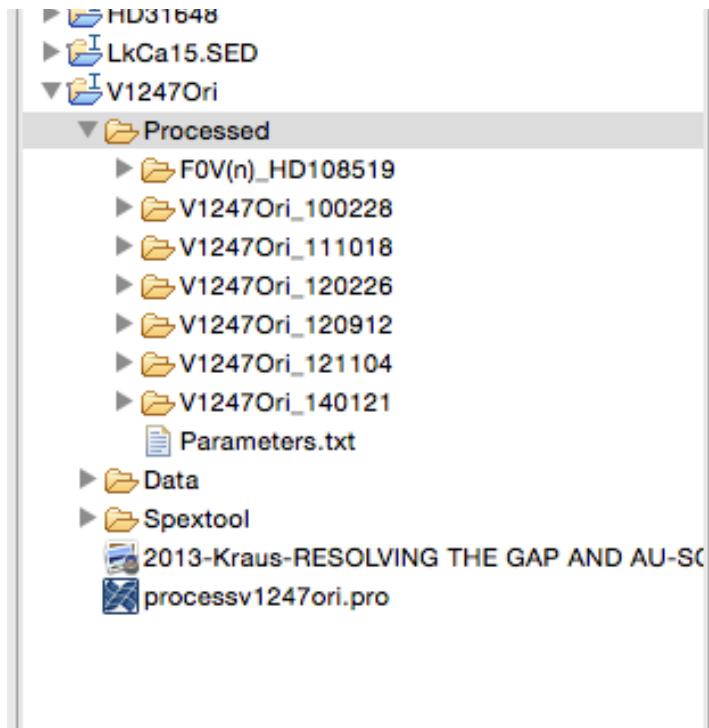
6. Hit return, the program will run and export the data to the desired folder.

The file structure should now look something like this:



Fitting

1. Navigate to the McQuilkin library, and find “M13StarFitter.nb”. Create a copy of this file to each folder that contains the data exported from “M13fitstocsv”. (Each subfolder in the processed folder should get a copy.) Each dataset should get its own copy, that way if something gets messed up, you can easily recalculate all the fits.
2. Also in the “Processed” folder, create a new file called “Parameters.txt”. To do this, right click on the “Processed” folder for your star in the “Projects” pane. And select “New File”. Remember to type the file ending in the box, it’s not added automatically.



Now double click on the “Parameters” file to open it. This file is for writing the values output by Mathematica in one place so that it is easier to find them.

3. Now open the “M13StarFitter.nb” file that you copied in the first step. Navigate to the file in Finder or Windows Explorer, trying to open it through IDL crashes Mathematica. You should see something like this:

StarFitter.nb

Wolfram Mathematica | STUDENT EDITION

Demonstrations | MathWorld | Wolfram Community | Help

Dust Modeling Function

```

dust[x_, t_, dn_, b_] := 
$$\left( \frac{dn}{x^6} \frac{1}{e^{\frac{1.4388 \times 10^4}{t*x}} - 1} \right) * x^b$$


Manipulate[Plot[dust[x, t, dn, b], {x, 0, 7}], {{t, 1300}, 1, 3000, 1},
{{dn, 2.2*10^-9}, 0, 1*10^-8, 1*10^-10}, {{b, .3}, 0, 1, .01}]

```

Import Data

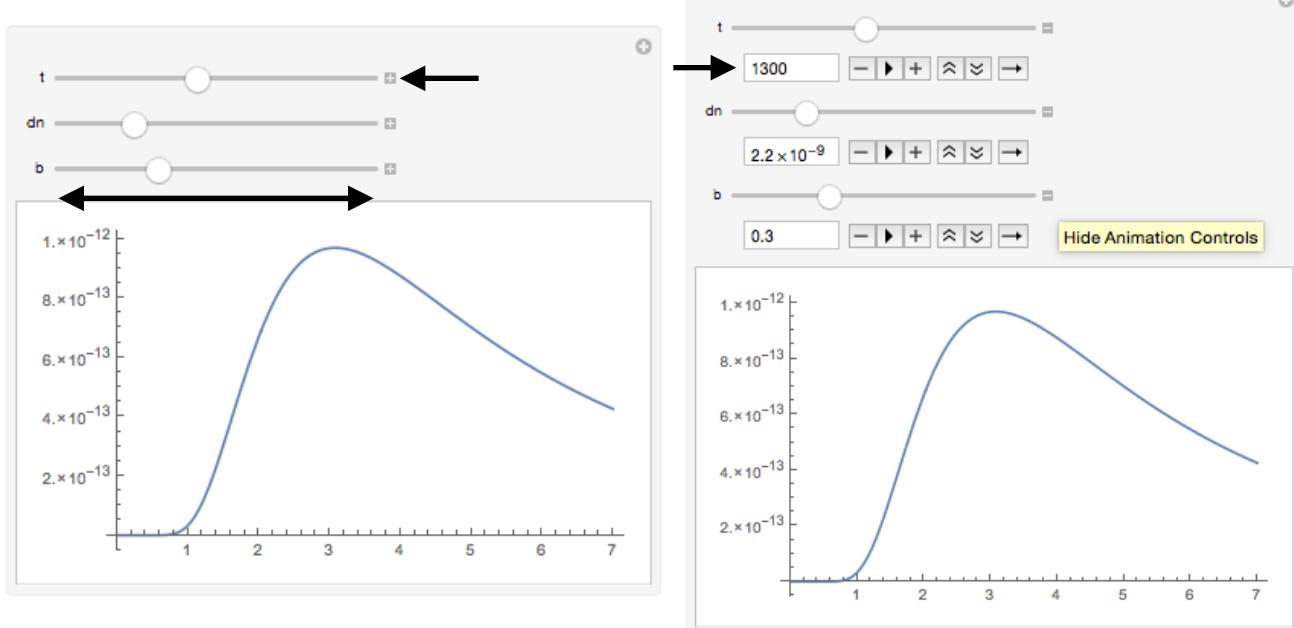
```

listTarget =
Import[
 "/Users/Brandon/Documents/School/Sitko/Research/Workspace/V1247Ori/
 Processed/V1247Ori_100228/V1247Ori_100228_revised.fits.csv"]

```

Note: To run code in Mathematica, select the cell with the code, and press $\text{Shift} + \text{Enter}$.

4. Run the first and second cell, in the “Dust Modeling” section, in order. This creates the model function we will be using to model the dust to help fit the standard star to the target data.
5. The second cell will create a manipulation box. This allows you to manipulate different parameters of the function, and see what the changes do in real time. You can adjust the parameters with the sliders, or by pressing the “+” button, show a text box where you can enter in a number. Use the text box for finer changes in values.

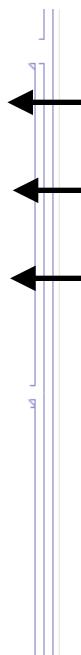
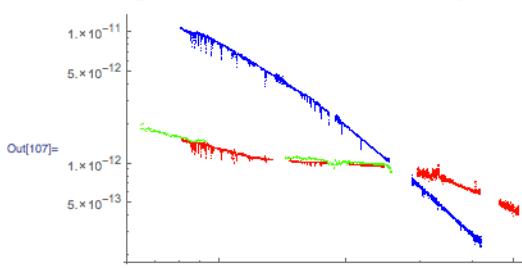


6. Scroll down to the “Import Data” section. Delete the file path that exists (including the quotes.) The select from the menu: Insert → File Path... Then browse to the CSV file for the target, prism, and standard star data you are analyzing. (The target and prism data should be in the folder with StarFitter.nb if you have been following along. The standard data should be in the “Data” folder.) Then run all the cells in that section in order.

Import Data

```
In[104]:= listTarget =
  Import[
    "/Users/Brandon/Documents/School/Sitko/Research/Workspace/V1247Ori/Processed/
     V1247Ori_100228/V1247Ori_100228_revised.csv"];
listTargetPrism =
  Import[
    "/Users/Brandon/Documents/School/Sitko/Research/Workspace/V1247Ori/Processed/
     V1247Ori_100228/V1247Ori_100228_prism_revised.csv"];
listStandard =
  Import[
    "/Users/Brandon/Documents/School/Sitko/Research/Workspace/V1247Ori/Processed/
     FOV(n)_HD108519/FOV(n)_HD108519.csv"];

ListLogLogPlot[{listTarget, listStandard, listTargetPrism},
  PlotStyle -> {Red, Blue, Green}, PlotRange -> All]
```



The target data will be red, the prism will be green, the standard will be blue.

7. Next we will create the standard model function (“Standard Model” section). To do this we need to clean up, and smooth out the data.

- a. So the first few lines remove data points. You’ll want about 300-600 data points.:)

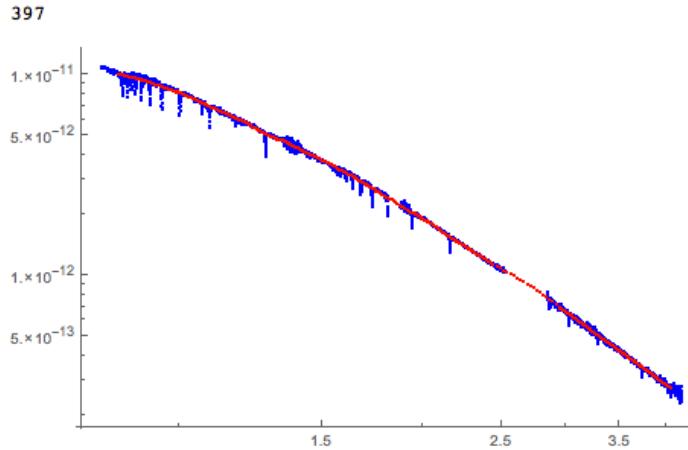
```
sdata = Drop[sdata, {1, Length[sdata], 2}];
```

Add or remove copies of this line and run the code until the number above the plot is around 300-600. (The number output above the plot is the number of data points that are left after some are removed.)

- b. The next line of code takes a moving average over the data to smooth it out. A value of 20 should be good. We just want the general form of the line of the original data.

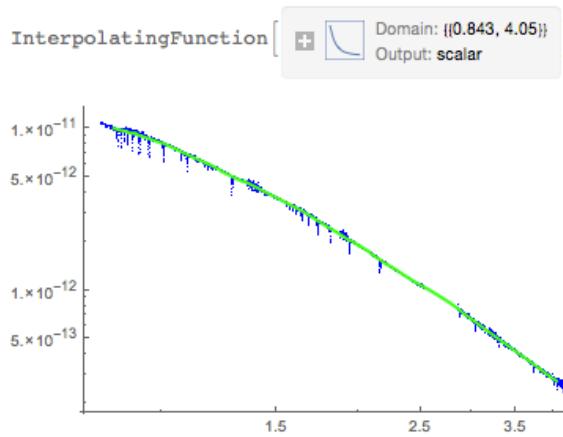
```
sdata = MovingAverage[sdata, 20];
```

If there are any spikes in the cleaned data (red), increase this value. If the form of the cleaned data gets distorted from the form of the original data (blue) decrease the value.



The number is the number of data points left after the data has been cleaned. The blue dots are the original data, and the red dots are the cleaned data.

- c. Now we are going to create an interpolating function for the data. An interpolation function just fills in the gaps between data points with spline curves so that the data can be used like a regular function. Just run the next cell to create the function, and plot it. The output should look like:



8. Next we are going to create the target model. The only thing we need to do here is smooth the data like we did for the target model.

- a. So the first few lines remove data points. You'll want about 300-600 data points.:

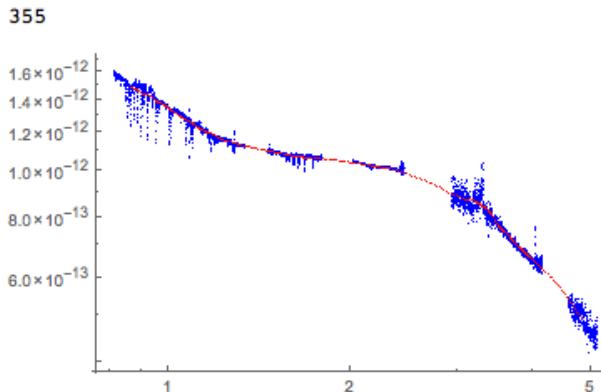
```
tdata = Drop[tdata, {1, Length[tdata], 2}];
```

Add or remove copies of this line and run the code until the number above the plot is around 300-600. (The number output above the plot is the number of data points that are left after some are removed.)

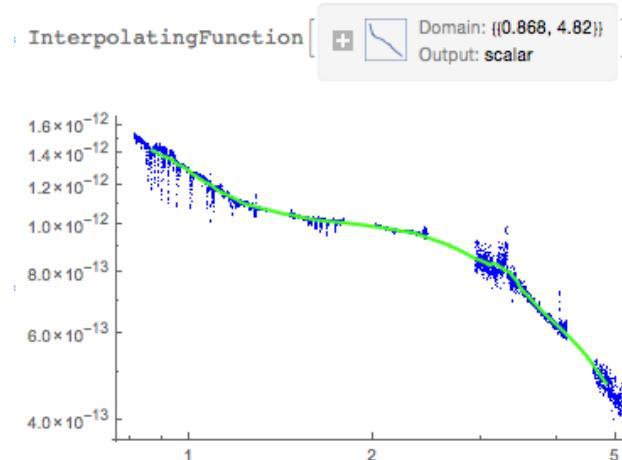
- b. The next line of code takes a moving average over the data to smooth it out. A value of 30 should be good. We just want the general form of the line of the original data.

```
tdata = MovingAverage[tdata, 30];
```

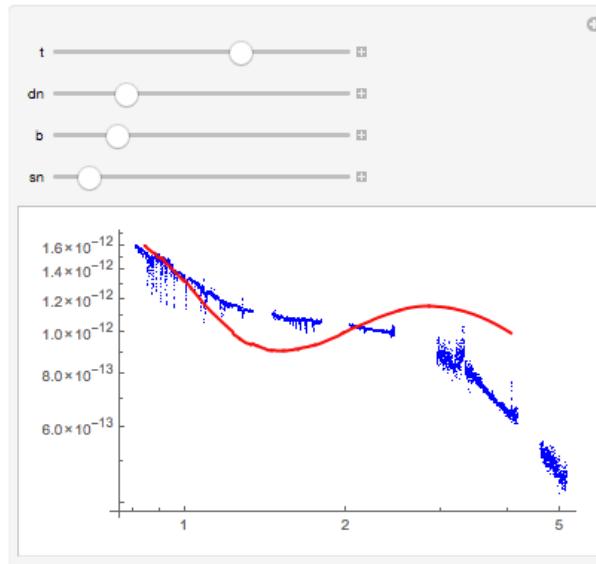
If there are any spikes in the cleaned data (red), increase this value. If the form of the cleaned data gets distorted from the form of the original data (blue) decrease the value.



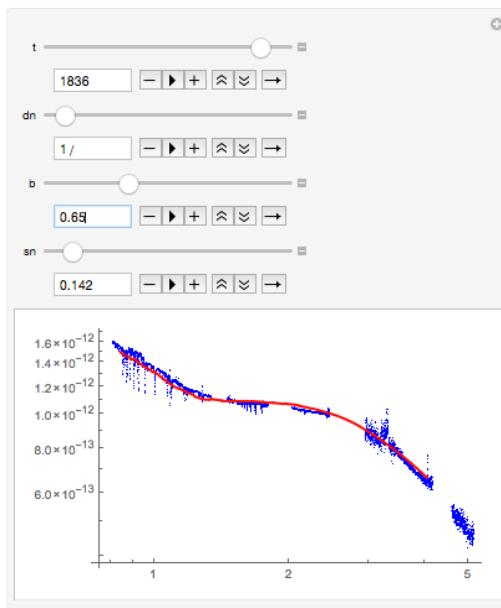
- c. Now we are going to create an interpolating function for the data. The output should look like:



9. Next we need to scale the target data so that it's intensity matches the intensity of the prism data. Run the cells in the "Scale the target data" section, and use the slider on the manipulate function to scale the target data, until it is as close to the prism data as possible. When the best scale value has been determined, enter it on the line below the manipulate box, and run the cell.
10. Now we will create the proper function so that we can fit the data. Run the first and second cells in the "Create the Function to Fit" section. The second cell will output a Manipulate box:



The goal is to make the red line (the standard star + dust model) match the blue data (target data). Try to get it as close as possible. You will most likely have to enter values manually to get fine enough adjustments. When you are finished you should have something like this:



11. The last thing we will do is fit the data using a least squares fit. This will get us the best possible values for the model.

- First we drop some of the first and last data points of the target data. This prevents an error when the `FindFit[]` function goes beyond the domain of the interpolating function.

```
dropAmount = 26;
data = Drop[Drop[tdata, dropAmount], -dropAmount]
```

Each time you run the fit, increase this number if you get a domain error. If you don't, try decreasing it until you do. Iterate until you find the lowest value where the error doesn't occur.

- Next we will fit the data. Enter in the values you found to be close from the `Manipulate` function in the last step as starting values of the `FindFit[]`. (Highlighted in red here.)

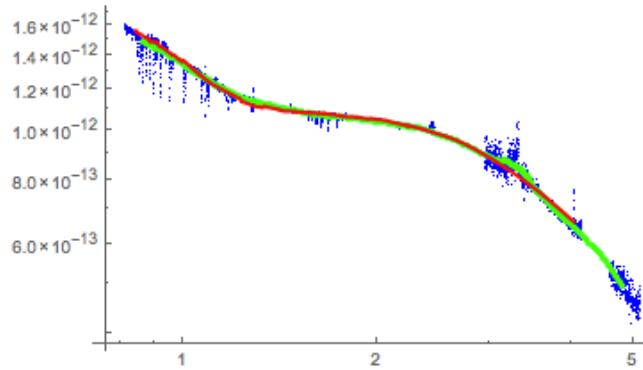
```

fit = FindFit[data,
  model[x, t, dn, b,
    sn], {{t, 1836}, {dn, 4.0*10^-10}, {b, 0.65`}, {sn, 0.142`}}, x]

```

The best fit values for the fit will be shown, and a plot of all the data and the fit will also be shown:

$\{t \rightarrow 1835.21, dn \rightarrow 3.70962 \times 10^{-10}, b \rightarrow 0.701215, sn \rightarrow 0.149287\}$



12. To really get good fit values, I would enter the numbers found back into the manipulate function, adjust them a little more to try to get a better fit, then enter them back into the `FindFit[]` function. Once you do this three or four times, you should have really good values for the find fit.

13. Now you should copy the fit values into the “Parameters.txt” file. I would write them in like this:

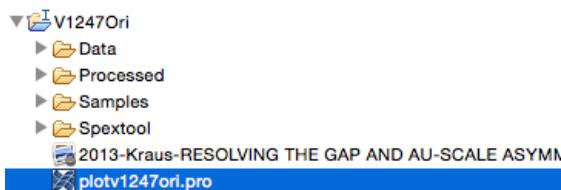
```

111018 scaled to 111017 prism: t = 1831.03, dn = 4.00434e-10, b = 0.712414, sn
= 0.167681 scale = 1.058

```

Plotting in IDL

Before you start plotting in IDL it is a good idea to create a file under your star's folder named “plot(star name).pro”.



And all the code you write following the guide below, write in this file. This is so that you, or anyone else with IDL can recreate the same plots you have made. Also you can run every single plot at once with a click of a button, you don't have to type every line into the console again, and again, if you made a mistake.

Open the plotting file and type the following:

```
pro (file name)  
  
end
```

For (file name) type the file name without “.pro” on the end.

All the code you write should be between these two lines. That will allow IDL to run it.

Next create a variable called “root”. And set it equal to the path to your IDL workspace:

```
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'
```

So you should have something like this.

```
pro (file name)  
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'  
  
;The rest of your code goes here.  
  
end
```

Also, anytime you see a “\$”, it means that the current line continues onto the next line. It is for formatting the code for better legibility. They can be omitted, or inserted anywhere.

Model Creator

“m13modelcreator.pro” creates a model of the target star, using the standard star, plus the other parameters we found using Mathematica. The function is documented, so you can view its documentation via IDL, or by opening the file in IDL. Below is a portion of the documentation:

CALLING SEQUENCE:

```
m13modelcreator, targetData, standardData, saveFolder, targetName, standardName,  
targetObservationDate, t, dustnorm, beta, starnorm, TargetScale = targetScale,  
OMITDATE = omitDate, BOXCARONLY = boxcarOnly, GAUSSIANONLY = gaussianOnly,  
DETAIPLPOTONLY = detailPlotOnly, FULLPLOTONLY = fullPlotOnly
```

INPUTS:

targetData: The fits file that contains the spectral data for the target star. Input is the full file path.
standardData: The fits file that contains the spectral data for the standard star. Input is the full file path.
saveFolder: The folder to place all the model save files and plots. Input is the full file path.
targetName: The name of the target star as a string.
standardName: The name of the standard star as a string.
targetObservationDate: The date the target star was observed as a string.
t: The temperature of the dust needed to get the model to match the target star spectrum.
dustNorm: The dust norm needed to get the model to match the target star spectrum.
beta: The beta needed to get the model to match the target star spectrum.
starNorm: The star norm needed to get the model to match the target star spectrum.
targetScale: The amount to scale the target data by. This scaling is usually the scaling required to get the input target data to match the flux of its prism data companion.

KEYWORD PARAMETERS:

OMITDATE: Set this keyword to not include the targetObservationDate in the plot.
BOXCARONLY: Set this keyword to only use boxcar smoothing.
GAUSSIANONLY: Set this keyword to only use gaussian smoothing.
DETAIPLPOTONLY: Set this keyword to only create the detail plots.
FULLPLOTONLY: Set this keyword to only create the full plots.

Here is an example of it being called to plot data for V1247Ori:

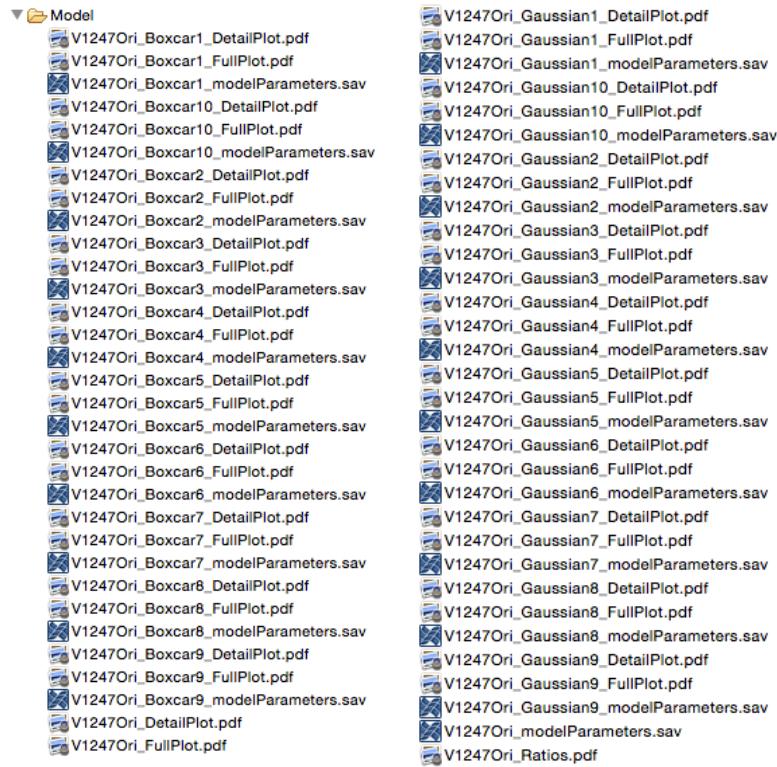
```
m13modelcreator, root + 'V1247Ori/Data/V1247Ori_111018.fits', root + 'V1247Ori/  
Data/F0V(n)_HD108519.fits', root + 'V1247Ori/Processed/V1247Ori_111018/Model',  
'V1247Ori', 'HD108519 F0V N Standard', '111018', 1831.03, 4.00434e-10, 0.712414,  
0.167681, 1.1
```

Here “root” is defined to be the path to IDL’s workspace:

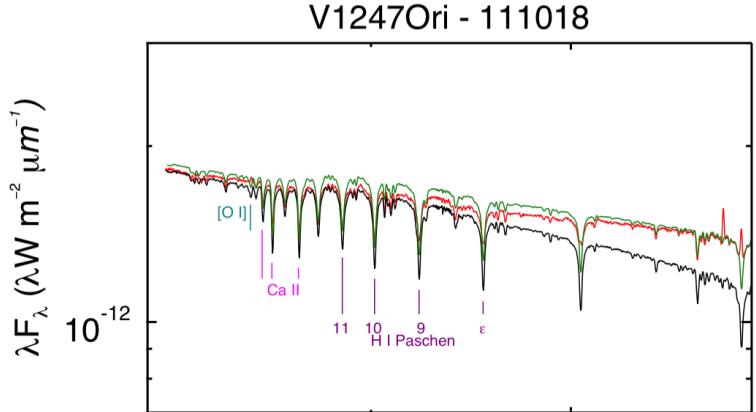
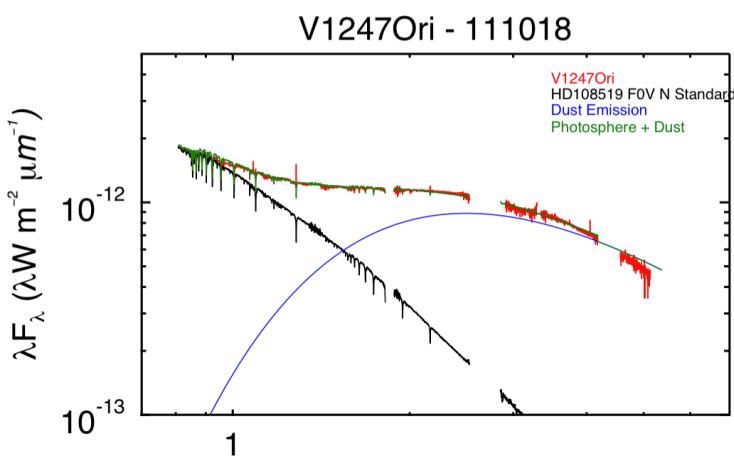
```
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'
```

When M13ModelCreator is run, a bunch of windows will flash on and off the screen. This is normal, it means that the program is working.

The model creator will create a bunch of different files for different parameters (Which are denoted in the file name.):

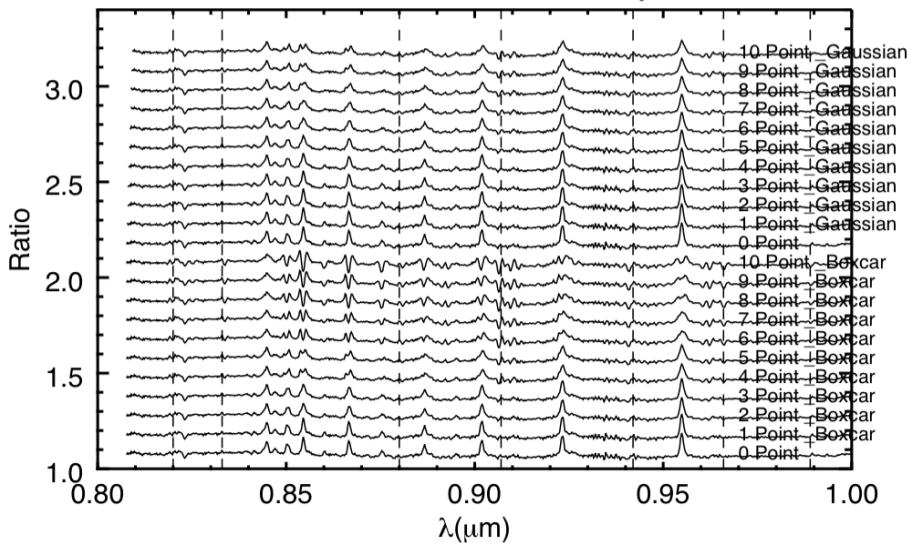


M13ModelCreator will create a detail and a full plot for a bunch of different smoothings. Ten sets of plots for a BoxCar smoothing with values 1-10, and ten sets of plots for Gaussian smoothing with values 1-10. There will also be a set for no smoothing.



The final window that should appear should have the following plot:

V1247Ori 111018 - Comparison



This plot compares the different smoothing types and strengths that have been applied to the data. The spectra that looks the best will be used for the next step. The best spectra will be the one that still has strong spikes, but a lot of the smaller bumps will have been smoothed out. Usually a 4-5 point Gaussian is the best. Boxcar is sub-par, it ends up inverting a lot of the lines, and tends to make things worse. If you have to pick a Boxcar, I would go with a 1-3 Point.

Prism Data

If you have to model off of prism data, you are most likely going to have to set the /BOXCARONLY flag. Prism data has too few data points for Gaussian smoothing to work.

Band Flux

“m13bandflux.pro” calculates the average flux if the different frequency bands. Currently it calculates the average flux in the JHKL bands.

CALLING SEQUENCE:

```
m13bandflux, inputFile, outputFolder
```

INPUTS:

inputFile: The sav file output by M13ModelCreator with the best smoothing value.
outputFolder: The folder to output all the plots and data files to.

Here is an example of it being called to plot data for V1247Ori, with the Gaussian 5 point smoothing.:.

```
m13bandflux, root + 'V1247Ori/Processed/V1247Ori_111018/Model/  
V1247Ori_Gaussian5_modelParameters.sav', root + 'V1247Ori/Processed/  
V1247Ori_111018/Lines'
```

Here “root” is defined to be the path to IDL’s workspace:

```
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'
```

It will create a single sav file that will contain all the necessary information.

Auto Lines

“m13autolines.pro” compares the individual spectra lines for the target star, and the model star. (The model created by M13ModelCreator.) It creates a plot for each line, plotting the two spectral lines, their difference, and plots the Chi-Squared difference for each. It is very simple to run, just plug in the sav file for the best smoothing output by M13ModelCreator, and enter the folder to save all the plots to.

CALLING SEQUENCE:

```
m13autolines, inputFile, outputFolder
```

INPUTS:

inputFile: The sav file output by M13ModelCreator with the best smoothing value.
outputFolder: The folder to output all the plots and data files to.

Here is an example of it being called to plot data for V1247Ori, with the Gaussian 5 point smoothing.:

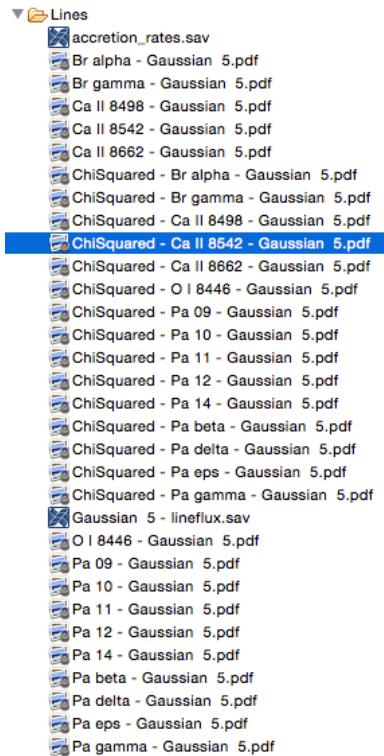
```
m13autolines, root + 'V1247Ori/Processed/V1247Ori_111018/Model/  
V1247Ori_Gaussian5_modelParameters.sav', root + 'V1247Ori/Processed/  
V1247Ori_111018/Lines'
```

Here “root” is defined to be the path to IDL’s workspace:

```
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'
```

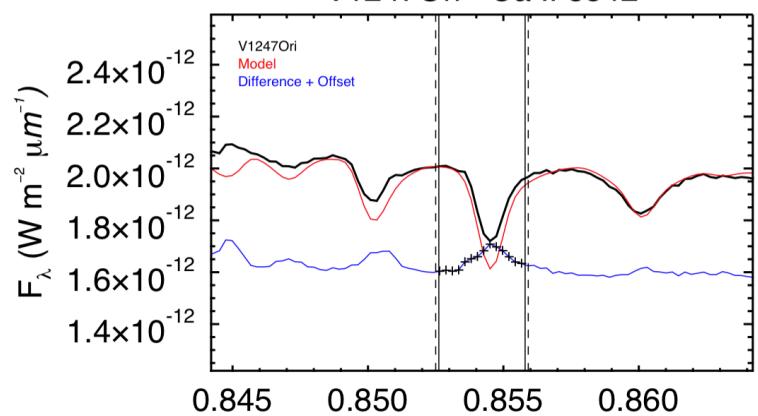
Again once run, a bunch of windows will flash on and off screen. Just wait for it to finish.

You should end up with a folder full of files like this:

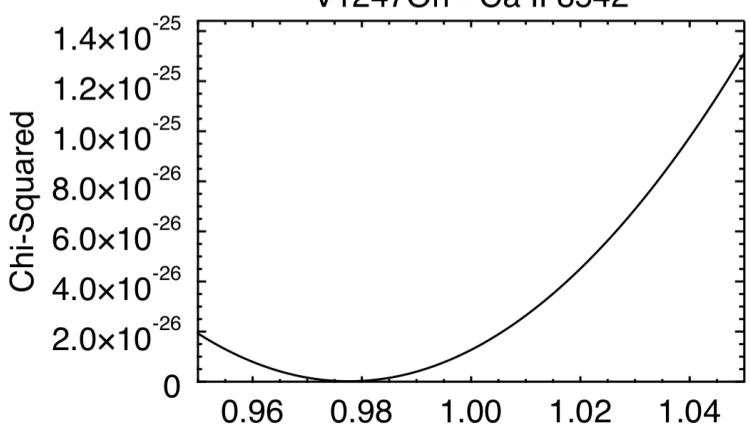


Each line should have a pair of plots:

V1247Ori - Ca II 8542



V1247Ori - Ca II 8542



Accretion Rates

“m13accretionrates.pro” calculates the accretion rate of dust into the star based off of the star’s spectra, distance from Earth, mass, and radius. No plots are generated with this file, just a sav file with the accretion rate.

CALLING SEQUENCE:

`m13accretionrate`, lineFluxFile, saveFolder, distance, starMass, starRadius

INPUTS:

lineFluxFile: The lineflux sav file output by M13AutoLines

saveFolder: The folder to output the file with the accretion rates to.

distance: The distance from Earth to the star in parsecs.

starMass: The mass of the star in solar masses.

starRadius: The radius of the star in solar radai.

Here is an example of it being called to plot data for V1247Ori.

```
m13accretionrate, root + 'V1247Ori/Processed/V1247Ori_111018/Lines/Gaussian 5 -  
lineflux.sav', root + 'V1247Ori/Processed/V1247Ori_111018/Lines', 385, 1.86, 2.3
```

Once you have run it, you should end up with a file named “(starname)_accretion_rates.sav”.

Plot(star name).pro

The plot file should now have three lines of code in it:

```
pro plotV12470ri
```

```
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_111018.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root +  
'V12470ri/Processed/V12470ri_111018/Model', 'V12470ri', 'HD108519 FOV N Standard', '111018', 1831.03, 4.00434e-10,  
0.712414, 0.167681, 1.1  
m13bandflux, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root +  
'V12470ri/Processed/V12470ri_111018/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root +  
'V12470ri/Processed/V12470ri_111018/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/  
Processed/V12470ri_111018/Lines', 385, 1.86, 2.3  
  
end
```

You now need to repeat these steps for all the data files you have. You should just be able to copy and paste these three line, and slightly modify a few parameters. Eventually you will end up with this:

```
pro plotV12470ri  
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_100228_revised.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_100228/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '100228', 1835.21, 3.70962e-10, 0.701215, 0.149287, 1.04734  
m13bandflux, root + 'V12470ri/Processed/V12470ri_100228/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_100228/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines', 385, 1.86, 2.3  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_111018.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_111018/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '111018', 1831.03, 4.00434e-10, 0.712414, 0.167681, 1.1  
m13bandflux, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines', 385, 1.86, 2.3  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_120226.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120226/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '120226', 1850.46, 3.99859e-10, 0.673524, 0.146005, 1.16778  
m13bandflux, root + 'V12470ri/Processed/V12470ri_120226/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_120226/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines', 385, 1.86, 2.3  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_120912.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120912/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '120912', 1881.79, 4.00317e-10, 0.691722, 0.149589, 1.01318  
m13bandflux, root + 'V12470ri/Processed/V12470ri_120912/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_120912/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines', 385, 1.86, 2.3  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_121104.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_121104/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '121104', 1855.96, 3.66121e-10, 0.797781, 0.155318, 1.07100  
m13bandflux, root + 'V12470ri/Processed/V12470ri_121104/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_121104/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines', 385, 1.86, 2.3  
  
m13modelcreator, root + 'V12470ri/Data/V12470ri_140121.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_140121/Model', $  
'V12470ri', 'HD108519 FOV N Standard', '140121', 1846.98, 4.02646e-10, 0.527497, 0.147659, 0.925720  
m13bandflux, root + 'V12470ri/Processed/V12470ri_140121/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines'  
m13autolines, root + 'V12470ri/Processed/V12470ri_140121/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines'  
m13accretionrate, root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines', 385, 1.86, 2.3  
  
end
```

Line Comparison

“m13linecompare.pro” compares the spectra from each dataset. It plots them in two different ways, one unscaled, so that you can see the relative intensities of the spectra around the emission/absorption lines. The other scaled, so that you can compare the relative intensities of the emission/absorption lines. There is also an option to include the standard star in the plots, so generally you will have four plots per line.

CALLING SEQUENCE:

```
m13linecompare, filesArray, colorsArray, outputFolder, STANDARDSTAR =  
standardStar, STANDARDNAME = standardName
```

INPUTS:

filesArray: The array of spectra files to compare. These should be sav files output by M13ModelCreator. Use the same file used in M13AutoLines.

colorsArray: The colors to use to plot each dataset. There should be one color per dataset. These should be the named color string constants provided by IDL.

outputFolder: The folder to place all the comparison plots. Input is the full file path.

KEYWORD PARAMETERS:

STANDARDSTAR: Set this keyword to include the standard star in the plot. The value should be set to the full file path of the fits file for the standard star.

STANDARDNAME: The name of the standard star.

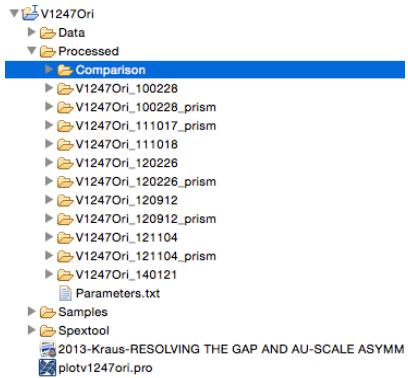
Here is an example of it being called to plot all the data files for V1247Ori (Without the standard star).

```
m13linecompare, [root + 'V1247Ori/Processed/V1247Ori_100228/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_111018/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120226/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120912/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_121104/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_140121/Lines/Gaussian 5 - lineflux.sav'], $  
['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue'], $  
root + 'V1247Ori/Processed/Comparison'
```

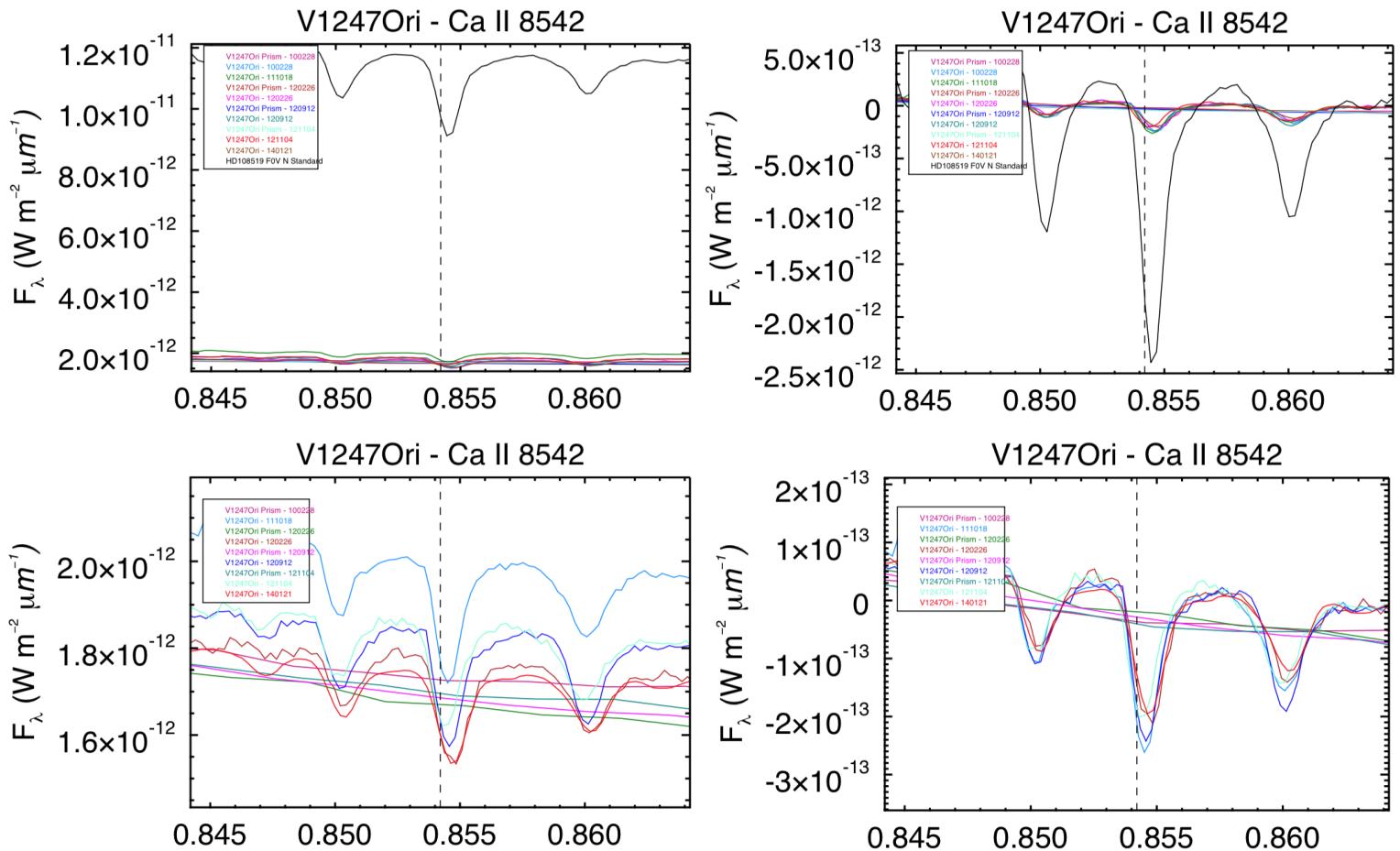
Here is an example of it being called to plot all the data files for V1247Ori (With the standard star).

```
m13linecompare, [root + 'V1247Ori/Processed/V1247Ori_100228/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_111018/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120226/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120912/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_121104/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_140121/Lines/Gaussian 5 - lineflux.sav'], $  
['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue'], $  
root + 'V1247Ori/Processed/Comparison', $  
STANDARDSTAR = root + 'V1247Ori/Data/F0V(n)_HD108519.fits', STANDARDNAME = 'HD108519 F0V N Standard'
```

The first input is an array. In IDL an array is a list of values, separated by commas, within a set of brackets (“[]”). The next input is an array of all the colors to use. And the final input is the output folder. The output folder should be a folder called “Comparison” in the processed folder.



I would run the program with and without the standard star. After you do that you should end up with four plots for each line in the comparison folder. They should look something like this:



Plot(star name).pro

Your plotting file should now look like this:

```
pro plotV1247ori

root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'

m13modelcreator, root + 'V12470ri/Data/V12470ri_100228_revised.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_100228/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '100228', 1835.21, 3.70962e-10, 0.701215, 0.149287, 1.04734
m13bandflux, root + 'V12470ri/Processed/V12470ri_100228/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_100228/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_111018.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_111018/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '111018', 1831.03, 4.00434e-10, 0.712414, 0.167681, 1.1
m13bandflux, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120226.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120226/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '120226', 1850.46, 3.99859e-10, 0.673524, 0.146005, 1.16778
m13bandflux, root + 'V12470ri/Processed/V12470ri_120226/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_120226/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120912.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120912/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '120912', 1881.79, 4.00317e-10, 0.691722, 0.149589, 1.01318
m13bandflux, root + 'V12470ri/Processed/V12470ri_120912/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_120912/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_121104.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_121104/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '121104', 1855.96, 3.66121e-10, 0.797781, 0.155318, 1.07100
m13bandflux, root + 'V12470ri/Processed/V12470ri_121104/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_121104/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_140121.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_140121/Model', $ 
  'V12470ri', 'HD108519 F0V N Standard', '140121', 1846.98, 4.02646e-10, 0.527497, 0.147659, 0.925720
m13bandflux, root + 'V12470ri/Processed/V12470ri_140121/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines'
m13autolines, root + 'V12470ri/Processed/V12470ri_140121/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines', 385, 1.86, 2.3

m13linecompare, [root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav'], $ 
  ['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue'], $ 
  root + 'V12470ri/Processed/Comparison'

m13linecompare, [root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', $ 
  root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav'], $ 
  ['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue'], $ 
  root + 'V12470ri/Processed/Comparison', $ 
  STANDARDSTAR = root + 'V12470ri/Data/F0V(n)_HD108519.fits', STANDARDNAME = 'HD108519 F0V N Standard'

end
```

Line Comparison Charts

"m13linecomparechart.pro" create bar charts to make it easy to compare the flux of the different emission lines over time. The inputs are generally the same as M13LineCompare.

CALLING SEQUENCE:

```
m13linecomparechart, filesArray, julianDatesArray, outputFolder
```

INPUTS:

filesArray: The array of spectra files to compare. These should be sav files output by M13ModelCreator. Use the same file used in M13LineCompare.

julianDatesArray: The julian date of the observation the data is for. Use Julday(month, day, year) to create the dates.

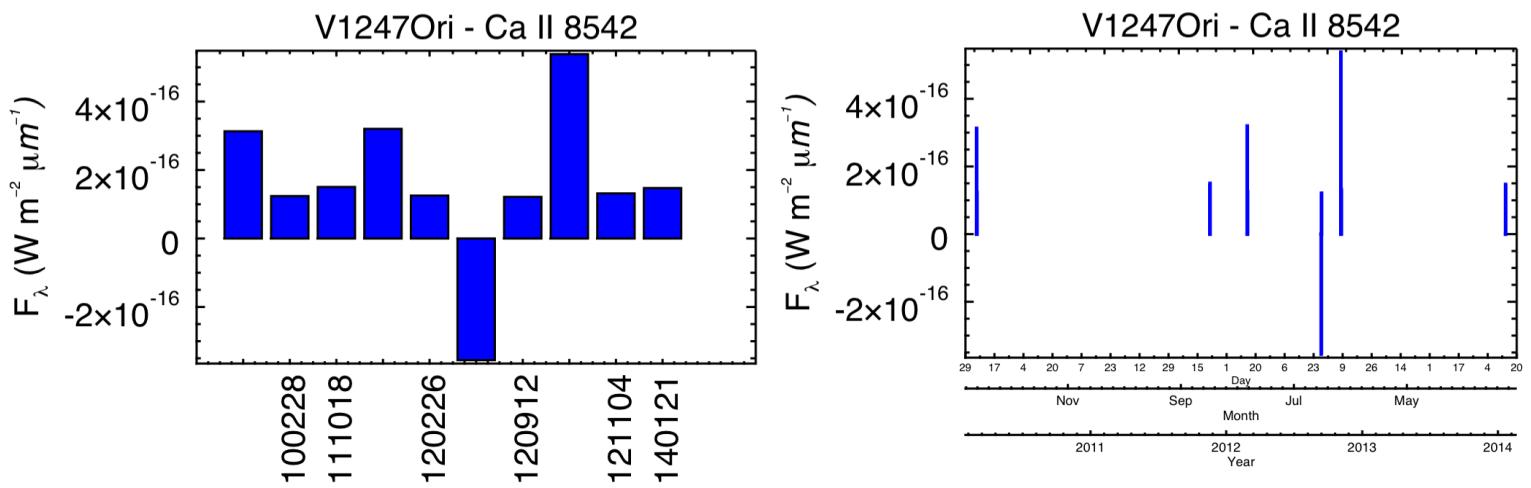
outputFolder: The folder to place all the comparison plots. Input is the full file path.

Here is an example of it being called to compare all the data files for V1247Ori.

```
m13linecomparechart, [root + 'V1247Ori/Processed/V1247Ori_100228/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_111018/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120226/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120912/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_121104/Lines/Gaussian 5 - lineflux.sav', $  
root + 'V1247Ori/Processed/V1247Ori_140121/Lines/Gaussian 5 - lineflux.sav'], $  
[JulDay(2, 28, 2010), Julday(2, 29, 2010), julday(11, 18, 2011), julday(2, 26, 2012), julday(2, 27, 2012),  
julday(9, 12, 2012), julday(9, 13, 2012), julday(11, 4, 2012), julday(11, 5, 2012), julday(1, 21, 2014)], $  
root + 'V1247Ori/Processed/Comparison'
```

The first input is the same array of files used for M13LineCompare. The second input is an array of julian dates, corresponding to the dates the data was collected. The last input is the output folder. It should be the same "Comparison" folder used for M13LineCompare.

After running the program, you should end up with a pair of plots for each emission line that look like this:



Plot(star name).pro

Your plotting file should now look like this:

```
pro plotV1247Ori  
root = '/Users/Brandon/Documents/School/Sitko/Research/Workspace/'  
  
m13modelcreator, root + 'V1247Ori/Data/V1247Ori_100228_prism_revised.fits', root + 'V1247Ori/Data/F0V(n)_HD108519.fits', root + 'V1247Ori/Processed/V1247Ori_100228_prism/Model', $  
'V1247Ori Prism', 'HD108519 F0V N Standard', '100228', 1859.05, 3.78484e-10, 0.543557, 0.150, 1.0, /BOXCARONLY
```

```

m13autolines, root + 'V12470ri/Processed/V12470ri_100228_prism/Model/V12470ri_Prism_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228_prism/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_100228_prism/Lines/No Smoothing - lineflux.sav', root + 'V12470ri/Processed/V12470ri_100228_prism/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_100228_revised.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_100228/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '100228', 1835.21, 3.70962e-10, 0.701215, 0.149287, 1.04734
m13autolines, root + 'V12470ri/Processed/V12470ri_100228/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_100228/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_111018.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_111018/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '111018', 1831.03, 4.00434e-10, 0.712414, 0.167681, 1.1
m13autolines, root + 'V12470ri/Processed/V12470ri_111018/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_111018/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120226.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120226/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '120226', 1850.46, 3.99859e-10, 0.673524, 0.146005, 1.16778
m13autolines, root + 'V12470ri/Processed/V12470ri_120226/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120226/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120226_prism.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120226_prism/Model', $ 
'V12470ri Prism', 'HD108519 F0V N Standard', '120226', 1848.86, 3.89389e-10, 0.727611, 0.143096, 1.0, /BOXCARONLY
m13autolines, root + 'V12470ri/Processed/V12470ri_120226_prism/Model/V12470ri_Prism_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120226_prism/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120226_prism/Lines/No Smoothing - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120226_prism/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120912.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120912/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '120912', 1881.79, 4.00317e-10, 0.691722, 0.149589, 0.1318
m13autolines, root + 'V12470ri/Processed/V12470ri_120912/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120912/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_120912_prism.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_120912_prism/Model', $ 
'V12470ri Prism', 'HD108519 F0V N Standard', '120912', 1862.16, 4.35605e-10, 0.602995, 0.144997, 1.0, /BOXCARONLY
m13autolines, root + 'V12470ri/Processed/V12470ri_120912_prism/Model/V12470ri_Prism_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_120912_prism/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_120912_prism/Lines/No Smoothing - lineflux.sav', root + 'V12470ri/Processed/V12470ri_120912_prism/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_121104.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_121104/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '121104', 1855.96, 3.66121e-10, 0.797781, 0.155318, 1.07100
m13autolines, root + 'V12470ri/Processed/V12470ri_121104/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_121104/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_121104_prism.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_121104_prism/Model', $ 
'V12470ri Prism', 'HD108519 F0V N Standard', '121104', 1858.5, 4.00788e-10, 0.677031, 0.146969, 1.0, /BOXCARONLY
m13autolines, root + 'V12470ri/Processed/V12470ri_121104_prism/Model/V12470ri_Prism_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_121104_prism/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_121104_prism/Lines/No Smoothing - lineflux.sav', root + 'V12470ri/Processed/V12470ri_121104_prism/Lines', 385, 1.86, 2.3

m13modelcreator, root + 'V12470ri/Data/V12470ri_140121.fits', root + 'V12470ri/Data/F0V(n)_HD108519.fits', root + 'V12470ri/Processed/V12470ri_140121/Model', $ 
'V12470ri', 'HD108519 F0V N Standard', '140121', 1846.98, 4.02646e-10, 0.527497, 0.147659, 0.925720
m13autolines, root + 'V12470ri/Processed/V12470ri_140121/Model/V12470ri_Gaussian5_modelParameters.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines'
m13accretionrate, root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', root + 'V12470ri/Processed/V12470ri_140121/Lines', 385, 1.86, 2.3

m13linecompare, [root + 'V12470ri/Processed/V12470ri_100228_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', $ 
['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue', 'teal', 'aquamarine', 'red', 'saddle_brown'], root + 'V12470ri/Processed/Comparison'

m13linecompare, [root + 'V12470ri/Processed/V12470ri_100228_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', $ 
['medium_violet_red', 'dodger_blue', 'green', 'firebrick', 'magenta', 'blue', 'teal', 'aquamarine', 'red', 'saddle_brown'], root + 'V12470ri/Processed/Comparison', $ 
STANDARDSTAR = root + 'V12470ri/Data/F0V(n)_HD108519.fits', STANDARDNAME = 'HD108519 F0V N Standard'

m13linecomparechart, [root + 'V12470ri/Processed/V12470ri_100228_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_100228/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_111018/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120226/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_120912/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104_prism/Lines/No Smoothing - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_121104/Lines/Gaussian 5 - lineflux.sav', $ 
root + 'V12470ri/Processed/V12470ri_140121/Lines/Gaussian 5 - lineflux.sav', $ 
[julday(2, 28, 2010), julday(2, 29, 2010), julday(11, 18, 2011), julday(2, 26, 2012), julday(2, 27, 2012), $ 
julday(9, 12, 2012), julday(9, 13, 2012), julday(11, 4, 2012), julday(11, 5, 2012), julday(1, 21, 2014)], $ 
root + 'V12470ri/Processed/Comparison'

end

```

Accretion Rate Charts

The last thing to do is plot a comparison of all the accretion rates. This program works the same as M13LineCompareChart.

CALLING SEQUENCE:

```
m13accretioncompare, filesArray, julianDatesArray, outputFolder
```

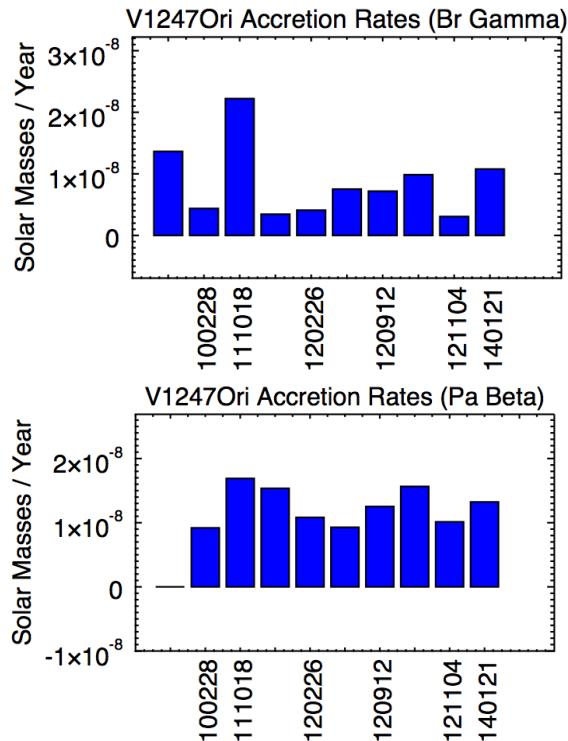
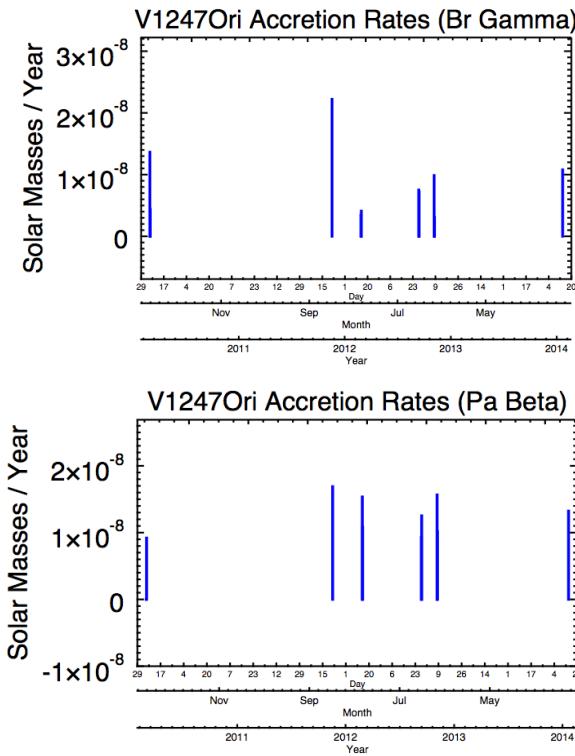
INPUTS:

- filesArray:** The array of accretion rate files to compare. These should be sav files output by M13AccretionRate.
julianDatesArray: The julian date of the observation the data is for. Use Julday(month, day, year) to create the dates.
outputFolder: The folder to place all the comparison plots. Input is the full file path.

Here is an example of it being called to compare all the data files for V1247Ori.

```
m13accretioncompare, [root + 'V1247Ori/Processed/V1247Ori_100228_prism/Lines/V1247Ori Prism_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_100228/Lines/V1247Ori_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_111018/Lines/V1247Ori_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120226_prism/Lines/V1247Ori Prism_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120226/Lines/V1247Ori_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120912_prism/Lines/V1247Ori Prism_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_120912/Lines/V1247Ori_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_121104_prism/Lines/V1247Ori Prism_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_121104/Lines/V1247Ori_accretion_rates.sav', $  
root + 'V1247Ori/Processed/V1247Ori_140121/Lines/V1247Ori_accretion_rates.sav'], $  
[julday(2, 28, 2010), julday(11, 18, 2011), julday(2, 26, 2012), $  
julday(9, 12, 2012), julday(11, 4, 2012), julday(1, 21, 2014)], $  
root + 'V1247Ori/Processed/Comparison'
```

After running the program, you should end up with four plots, two for each calculation type:



Appendix

SpexTool Cursor Commands

On almost any plot/graph in SpexTool, cursor commands can be used. Cursor commands allow you to zoom, select, scale, or perform other actions on the plot, or the data. Cursor commands are quite simple, just press the key on the keyboard for the command you wish to perform, then left click on the plot to perform the action. (Some actions require two clicks.)

Listed below are the commands used in this manual:

s - Select Tool: The select tool selects a range of values on a plot. To use it, hit “s” on the keyboard. Then select the leftmost edge of the range by left clicking on the plot. Then select the rightmost edge of the range by left clicking on the plot. You should end up with two vertical lines on the plot denoting the range:

Note: For most things requiring the select tool, be careful about the selected range. Do not select too close to the edges of the data. Also do not select data that has a lot of large spikes, like the very left end of the plot above. (Most other tools work like this)

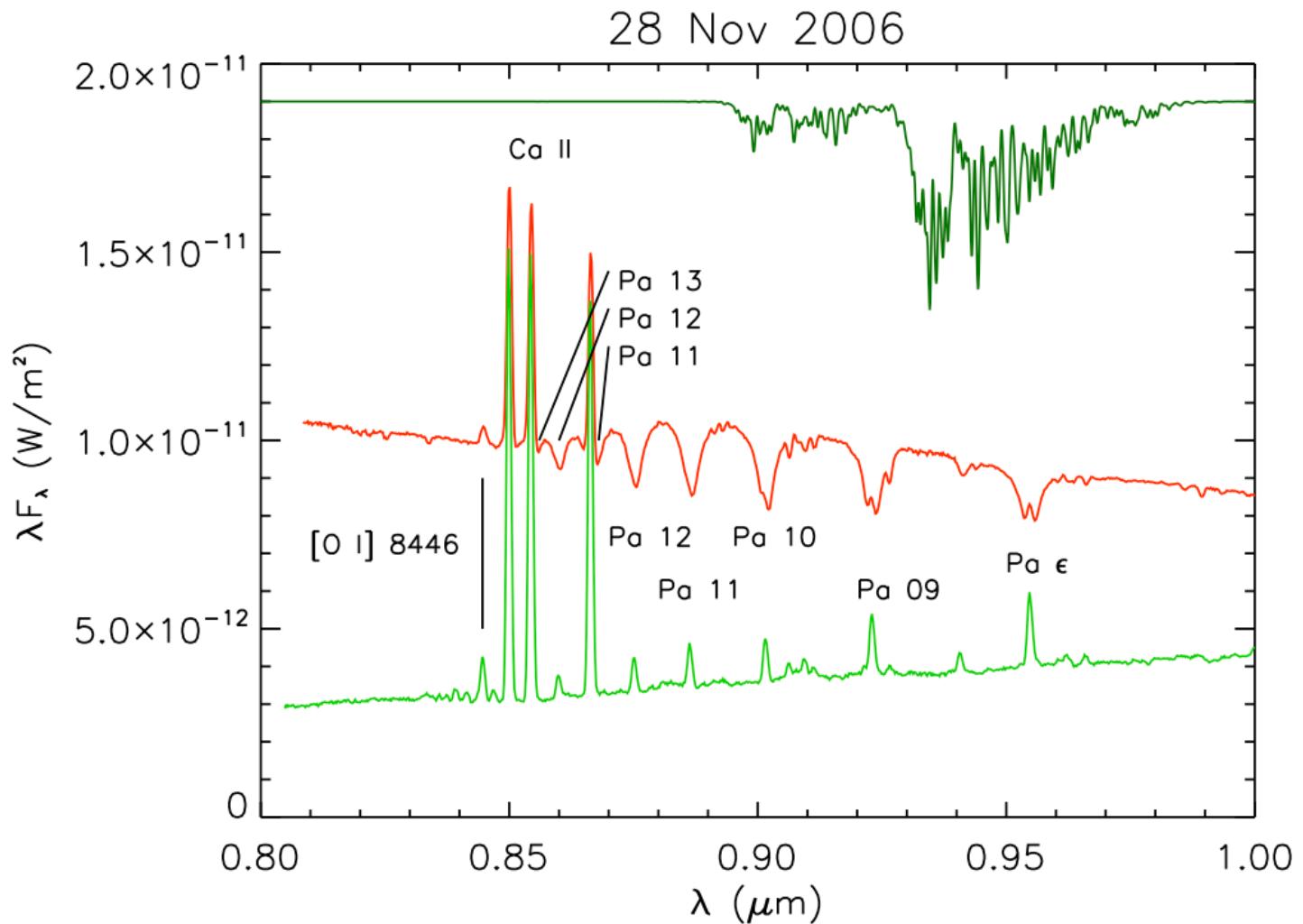
z - Zoom Tool: The zoom tool zooms into a rectangle on the given plot. To use it, hit “z”, then select the top left corner of the rectangle by left clicking on the plot. Then select the bottom right corner of the rectangle by left clicking on the plot. The plot should now be zoomed. You can now use other tools like normal.

Note: To leave a zoom mode, press “c”.

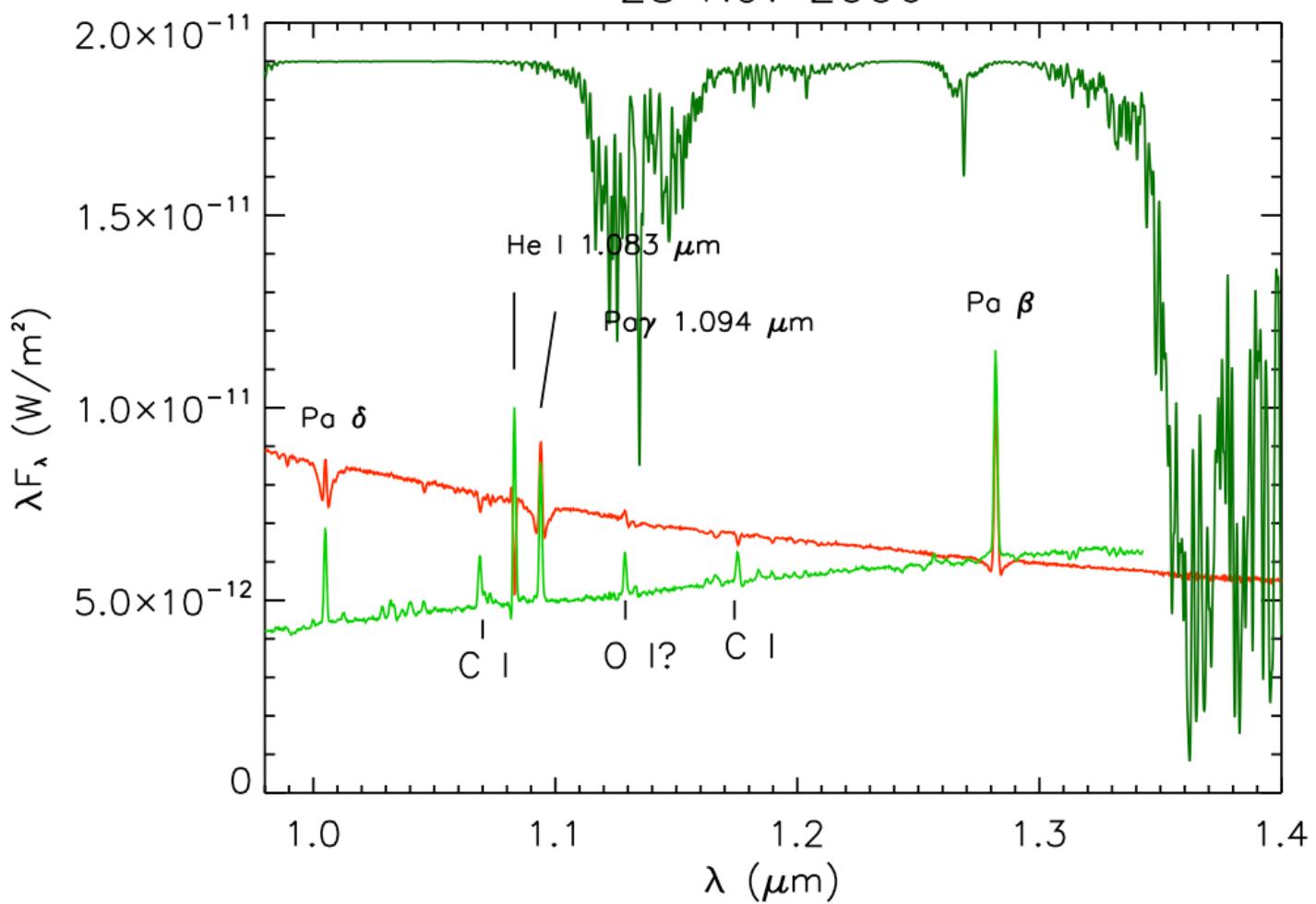
c - To clear a normalization, zoom, or line selection mode.

Line Identifications

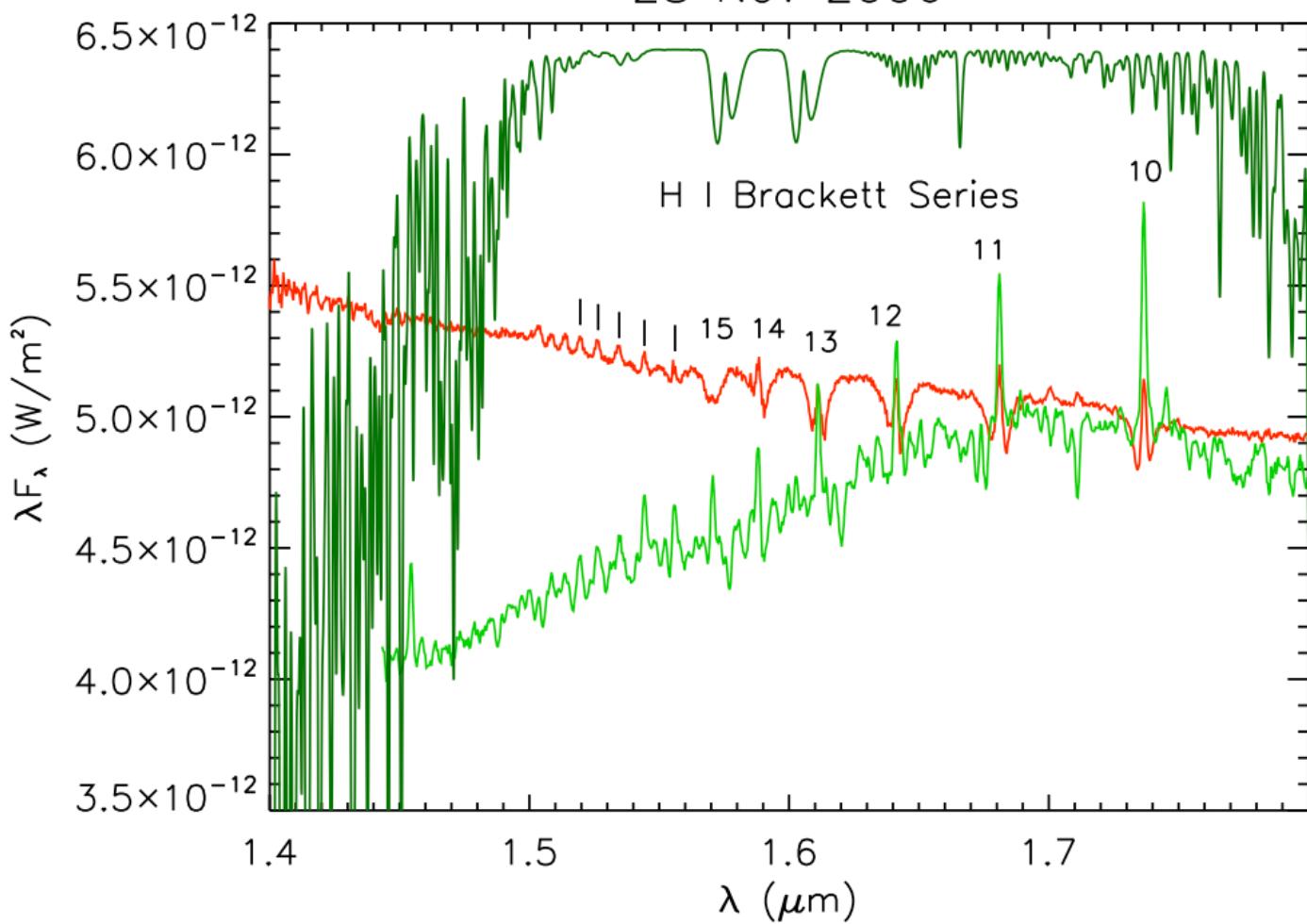
The following plots are an attempt at line identification in PMS stars. In each panel I show a section of SpeX data for two stars, along with the telluric spectrum that Spextool uses. This is meant to be an aid to knowing what features are real or not, but it pre-supposes that no weak stellar lines are included in the latter.



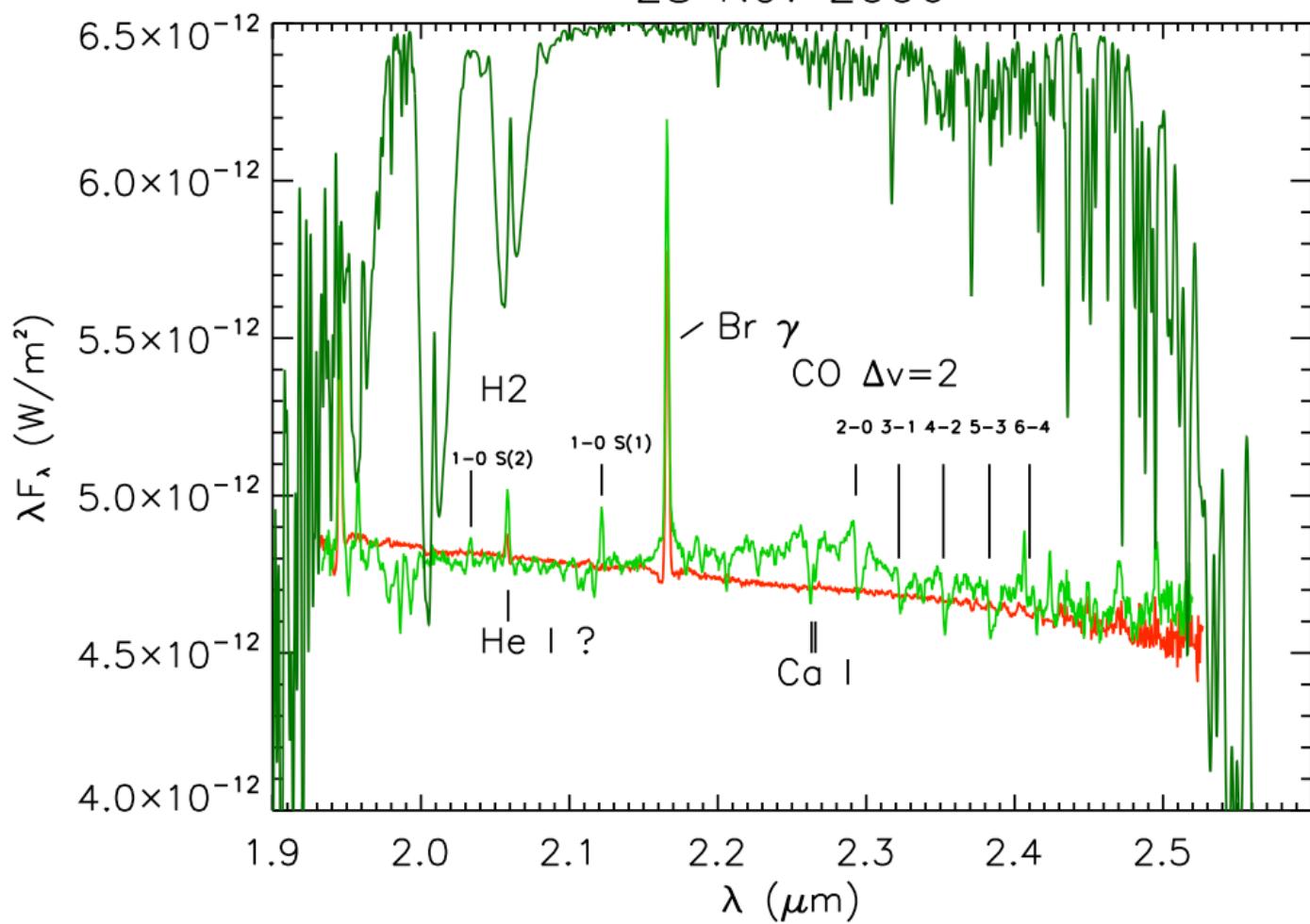
28 Nov 2006



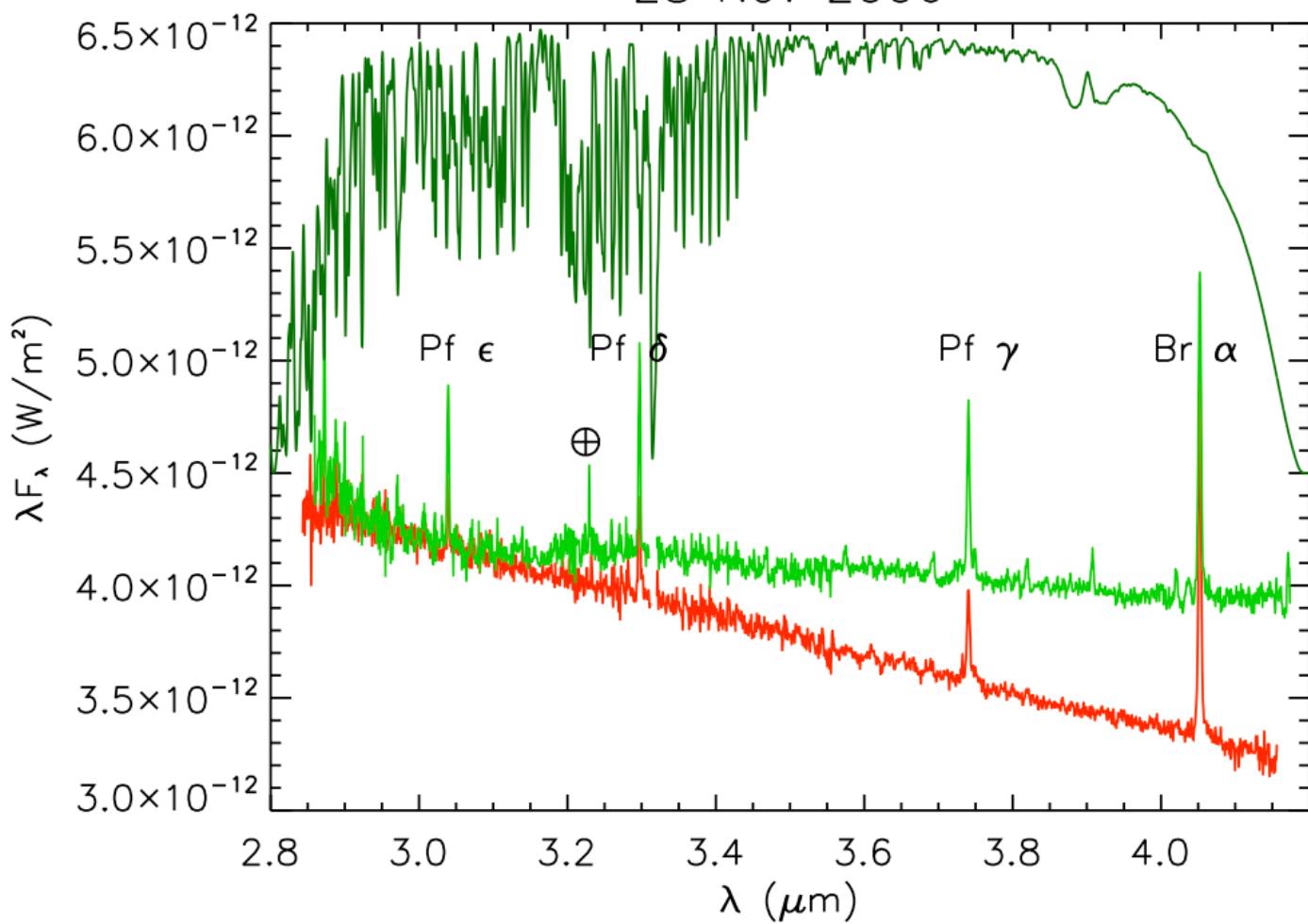
28 Nov 2006



28 Nov 2006



28 Nov 2006



28 Nov 2006

