Preparing for spectroscopic data reduction in a PyRAF / IRAF-less future

Adam F. Kowalski
University of Colorado, National Solar Observatory, Laboratory for Atmospheric and Space Physics

The APO Data Analysis Working Group surveyed the 3.5m Users in 2019 and found that most users still rely on IRAF/PyRAF for the reduction of their data from the 3.5m and 0.5m telescopes. The discontinued support of IRAF in the future, and the general trend toward a "Python-only" computing infrastructure in astronomy, puts the sustainability of the community's tried and trusted reduction pipelines with IRAF into jeopardy.

Introducing **SpecLab**! This is a Python-only data reduction facility that I have been developing over the past several years incrementally in spare time (on weekends and in the evenings). I started it because of the need to provide students a backup option for the imexam + DS9 convenience that is possible with IRAF.

Speclab has interactive GUI functionalities that are not available in any of the Python software packages yet developed by STScI and Astropy. SpecLab thus has pedagogical functionality with the intention of being used in undergrad and graduate classes to teach data reduction. It can also be used to provide publication-ready reductions via automated scripts.

In this poster, I present a demonstration of SpecLab's *imexam* for quick look assessment of your raw data from any of the instruments on the 3.5m at APO. I also show a preview of *identify* and *standard*.

SpecLab Framework

SpecLab uses PyQtGraph (v0.10) for interactive functionality. It also uses MPLD3 for pop-up plots in your internet browser. Unlike the popular Plotly, MPLD3 is able to convert *matplotlib* plots to HTML and is completely free.

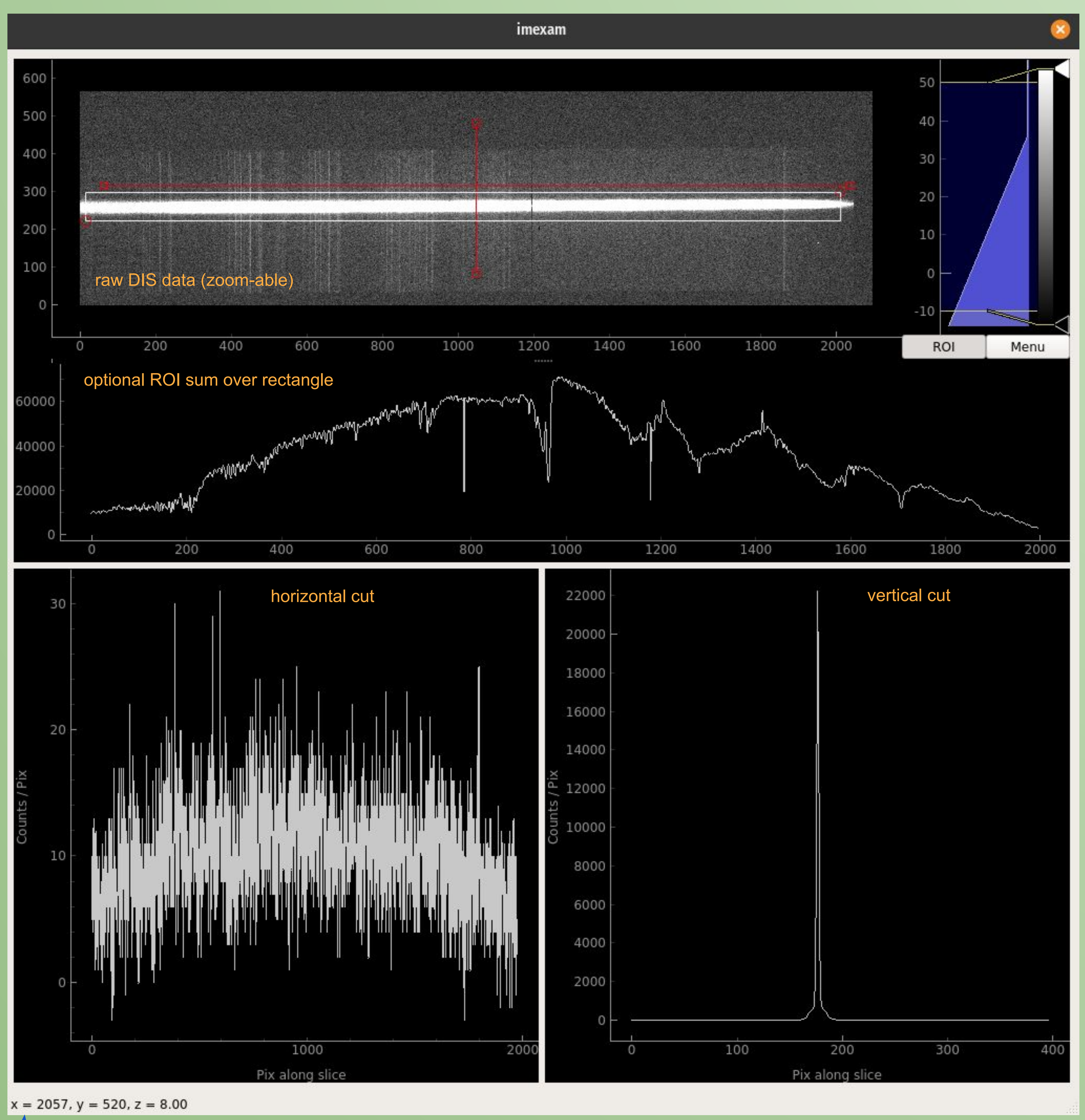
When reducing data, many statistical algorithms are employed. A few in SpecLab are:

- Non-linear least squares fitting (e.g., Gaussian centroiding): we use scipy.optimize.curve_fit or Newton-Raphson iteration.
- Polynomial fitting: we use design matrix linear least squares (e.g., Ivezic et al. 2019).
- Cubic spline fitting with n-pieces (n-"orders"): we use IRAF's algorithm (converted to a Python function).
- 1D spectral extraction: rebinning and fractional pixel extraction, similar to the magnify step in Karen Kinemuchi's ARCES data reduction guide.

imexam for quick look at spectra

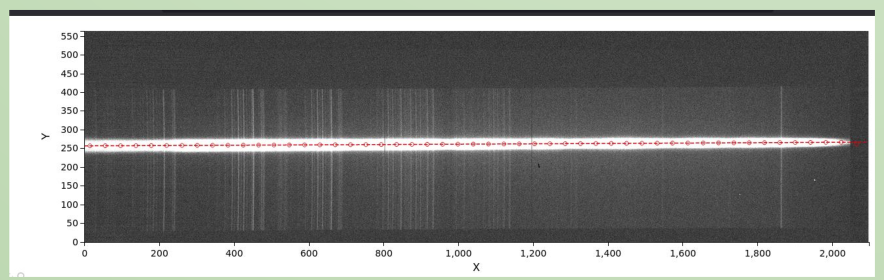
Below is a demonstration of *imexam* on a raw spectrum of the M dwarf DT Vir. This spectrum was obtained with DIS/R300 (CCD windowing was used to decrease readout time).

- all functionalities of SpecLab's *imexam* have been tested on raw spectra from DIS, ARCES, TripleSpec, the Horizontal Spectrograph at the Dunn Solar Telescope, the CMOS on the 24" telescope at the Sommers-Bausch Observatory at CU, and more.
- many useful keystrokes of DS9+IRAF's *imexam* have been incorporated into SpecLab's *imexam*, and it's straightforward for you to include additional keystroke commands. A few example keystrokes are shown below to the right.

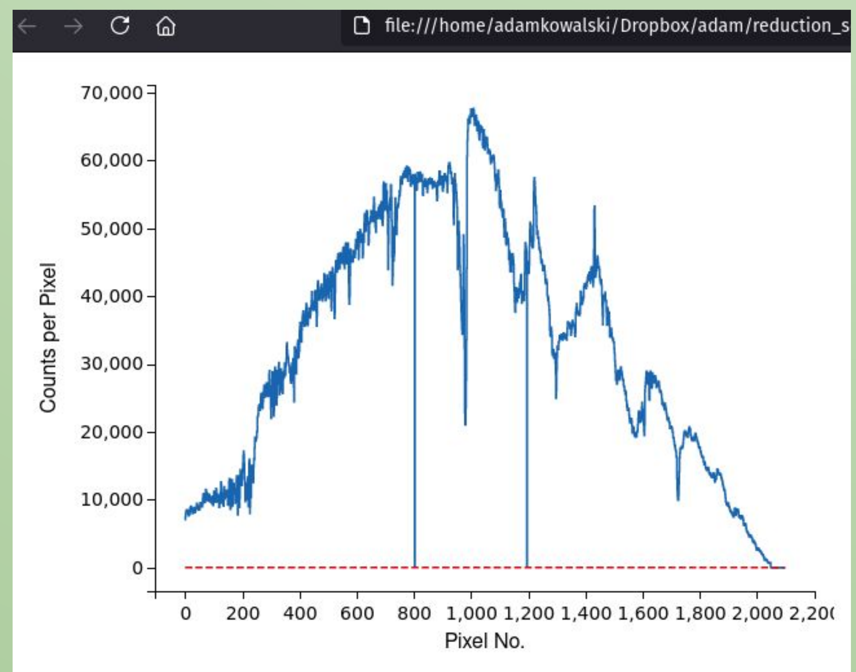


pixel value and coordinates at mouse cursor

The 't' key shows a trace of the spectrum in your web browser



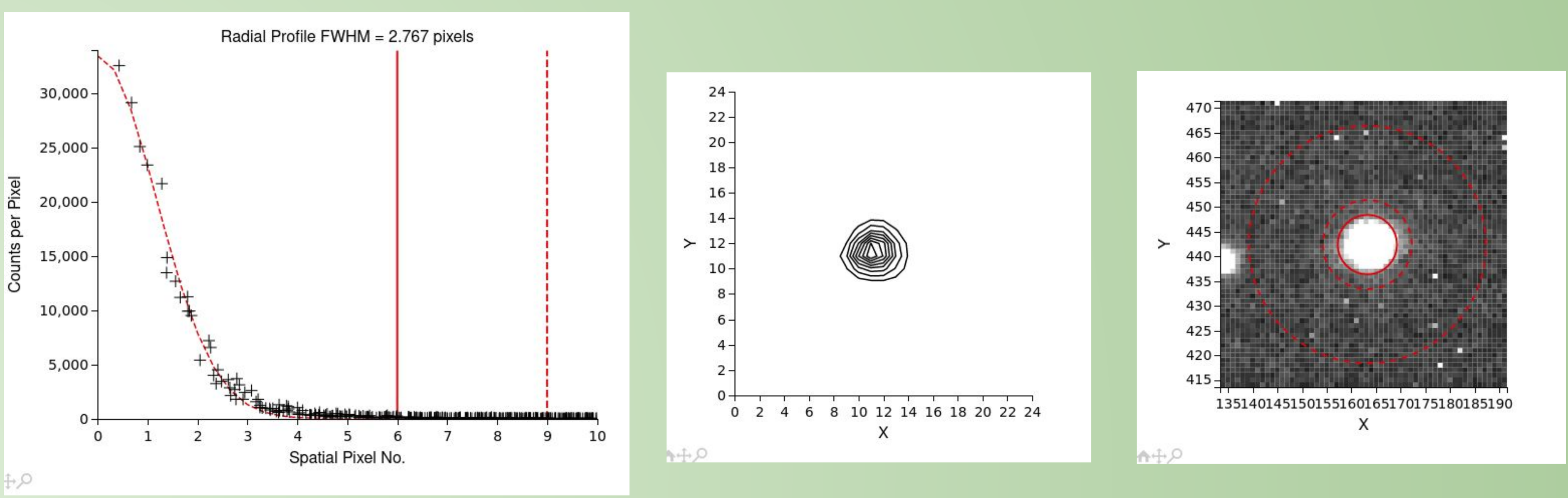
The 'x' key does a trace and extracts a spectrum for quick check of counts



imexam for quick look at imaging data

Below is a demonstration of SpecLab's *imexam* on a raw image of the M dwarf CR Dra. This image was obtained with ARCTIC in the ARCTIC 3500 filter.

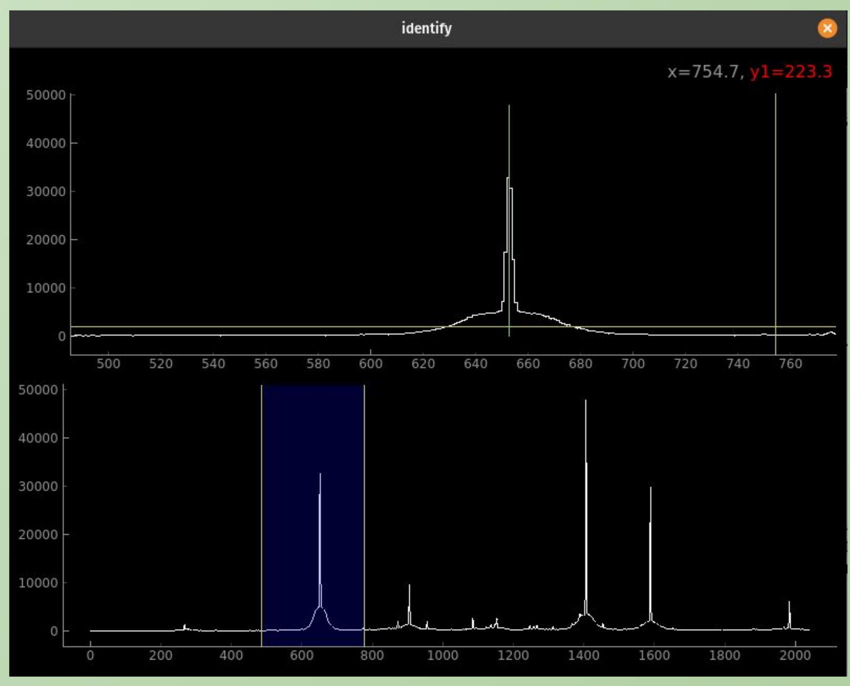
- all functionalities of *imexam* have been tested on raw imaging data from ARCTIC and Flarecam.



Keystrokes generate radial profiles and calculate aperture photometry (left), display contours (middle), and a zoom with apertures (right)

identify and *standard*

Identifying arc emission lines and standard star continuum windows (for flux calibration) by hand is necessary the first time through data reduction from any instrument. Below is a picture of SpecLab's replacement for *identify*. Once a solution is found, SpecLab's *reidentify* can be run to recenter line centroids that have been saved to a file.



SpecLab's *identify* GUI with a HgHeNeAr emission arc lamp spectrum from DIS B1200 (the broad wings are the result of DIS' contamination issues)

Future development

If interested in the software, please contact me at adam.f.kowalski@colorado.edu. Only *imexam* is ready to be distributed at this time. I am planning to link everything through github at a future date.

Generally, it has been difficult to obtain grants to fund a software engineer for this type of work, so I plan to finish it myself as a "hobby". All tools are being integrated into tutorials via Jupyter notebooks for classroom teaching and into fully automated scripts.

The final task is to implement the Marsh 1989 algorithm for optimal extraction of spectra. In doing this, I have found a bug in IRAF's variance-weighting extraction algorithm.

Given recent concerns about Python (Zwart 2020; but see also the reply by Augier et al. 2021), it's unclear if I should switch to Julia for SpecLab. In March 2021, the release of Julia 1.6 fixed its package loading time issues, and furthermore, it has a growing following in astronomy. The backwards compatibility practices of Python dependencies often cause code to break, which has been a problem for SpecLab using newer versions of PyQtGraph.

References

Augier, et al. 2021, *Nature Astronomy* 5, 334
Ivezic, Z. et al. 2019, Statistics, Data Mining, and Machine Learning in Astronomy
Marsh 1989, *PASP* 101, 1032
Zwart 2020, *Nature Astronomy* 4, 819

Acknowledgments

Thanks to Dr. Brett Morris for pointing me to PyQtGraph, and thanks to Dr. Yuta Notsu and Mr. Isaiah Tristan for testing the software.