

SVDFIT: An IRAF Task for Eigenvector Sky Subtraction

Douglas J. Mink and Michael J. Kurtz

Harvard-Smithsonian Center for Astrophysics

Abstract. A new method has been developed for estimating and removing the spectrum of the sky from deep spectroscopic observations. This method does not rely on simultaneous measurement of the sky spectrum with the object spectrum. The technique is based on the iterative subtraction of continuum estimates and eigenvector sky models derived from singular value decompositions of sky spectra and sky spectra residuals. IRAF tasks have been developed to implement this technique. Using simulated data derived from small-telescope observations, we demonstrate that the method is effective for faint objects on large telescopes.

1. Introduction

Multi-fiber spectrograph apertures are too small to return simultaneous sky spectra for each fiber, so the sky contribution which is removed from the observed object+sky is not exactly what was observed by that fiber. The current solution is to aim some fibers at blank sky and subtract them from the all of the object+sky spectra, using the closest sky or a mean image sky and fitting it in some way, such as by matching the OI line at 5577 Angstroms.

In a previous paper (Kurtz & Mink 2000), we described a way to model the night sky by decomposing a set of observed night sky spectra into orthogonal eigenvectors using singular value decomposition. A subset of these eigenvectors is then fit to the object+sky spectra and the fit removed, leaving residual object spectra. This is made easier by the fact that for redshift work, any continuum signal is removed anyway, as only the positions and shapes of absorption and emission lines are needed.

This paper describes the software package we developed to carry out the work. Because software written as part of the RVSAO package (Kurtz & Mink 1998) already did much of the work of dealing with spectra, we adapted that and produced a new IRAF package, SVDFIT. The code was written in SPP to take advantage of many existing useful subroutines, though we experimented with various numbers of iterations using CL scripts and simpler SPP tasks.

2. New IRAF Tasks

`svdprep` is an SPP task, based on RVSAO's `sumspec` task, which rebins spectra to a common dispersion, with additional processing at the same time. Specific portions of the spectra, such as the brightest night sky lines, may be re-

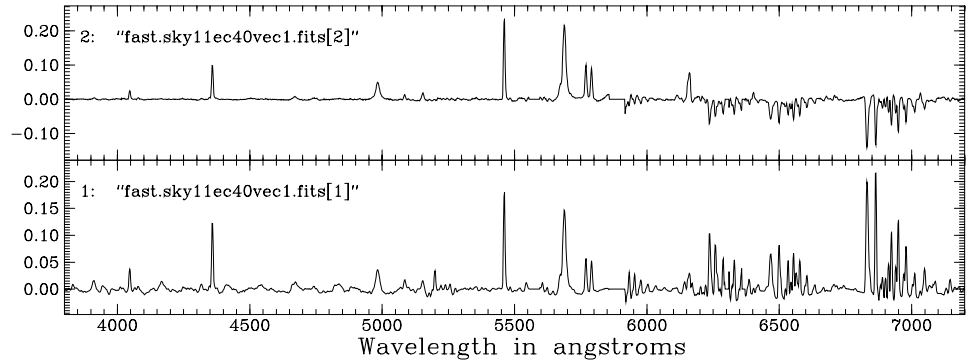


Figure 1. First and second eigenvectors of the night sky after major lines and continuum have been removed.

moved, as may a continuum fit using the same continuum removal subroutines as in RVSAO. Spectra may be normalized to their mean values so they have a common range of values, and stacked into a single two-dimensional file for further processing by `svdvec` or `svdres`. The spectra may also be redshifted to a common radial velocity, though this feature isn't used for sky subtraction.

`svdvec` is a Fortran program, called by IRAF as a foreign task, which decomposes an array of spectra, such as that created by `svdprep`, into eigenvectors using Singular Value Decomposition. An array of eigenvector spectra with the same dispersion is returned.

`svdres` is an SPP task which fits a set of eigenvectors to a spectrum, stacked spectra, or a list of spectra, returning the residuals as spectra in the same format as input. The processing features of `svdprep` are included in `svdres`, so it can be run as a single task on unbinned spectra.

`svdres2` is a modification of `svdres` to fit two sets of eigenvectors to a spectrum, removing an optional continuum in between, returning the final residuals, which may be sky-subtracted spectra. This avoids a lot of I/O and more easily parameterizes the entire two-pass process.

3. Making Eigenvectors

Sky spectra from long exposures on moonless nights were rebinned using the `svdprep` task to make an array of spectra with identical dispersion. Before rebinning each spectrum, the brightest night sky lines (OI at 5577, 6300, and 6363 Angstroms and Na at 5890 Angstroms) were replaced by interpolated continuum, and the entire continuum signal was removed. We assume that any spectral features on these bright lines would have been overwhelmed by the Poisson noise in a long-exposure sky spectrum.

Eigenvectors for these spectra were then computed using `svdvec`. Any singular valued decomposition program could be used. Figure 1 shows the first two eigenvectors.

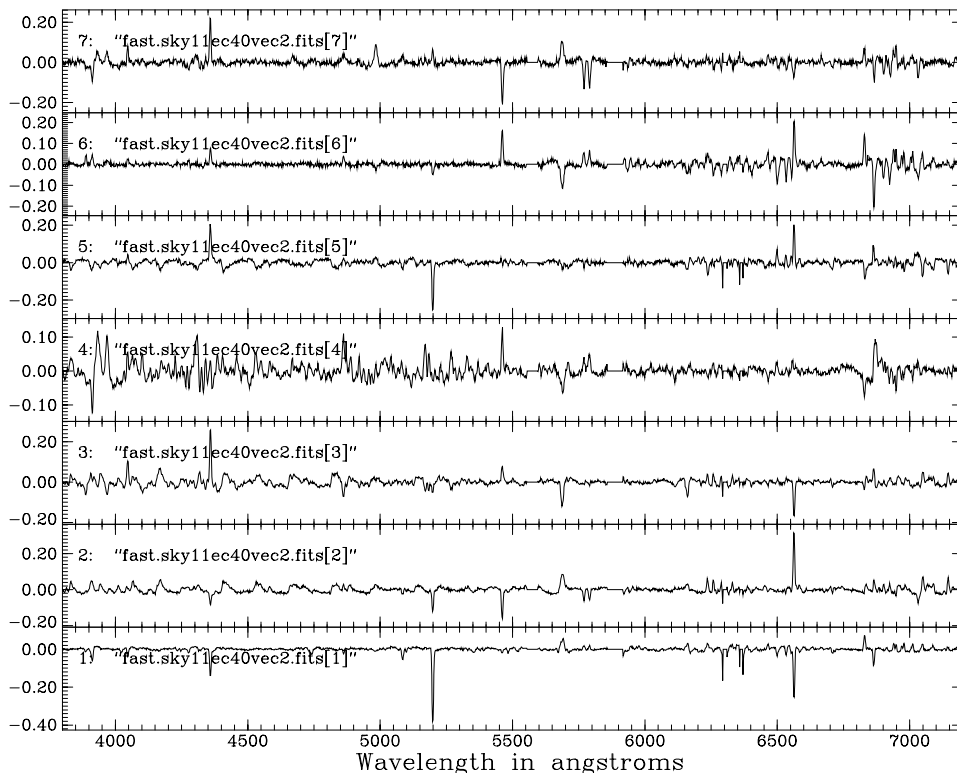


Figure 2. Next seven eigenvectors of the night sky after first two eigenvectors and second continuum have been removed.

Major sky spectrum features are then removed by fitting the first two eigenvectors—3 or 4 might be OK, too—and removing the fit from the array of spectra using `svdres`.

Using `svdprep`, the continuum is removed from the residual spectra. It turns out that many of the eigenvectors in the first pass account for night to night sky continuum variation on a scale smaller by the sky spectrum signal. Removing this spectrum by spectrum leaves only sky and object spectrum features.

We again compute eigenvectors for the array of residual spectra using `svd-vec`. Fitting the first 7–10 of these eigenvectors using `svdres` removes the rest of the sky features from our object+sky spectrum. Figure 2 shows the first seven eigenvectors.

4. Removing Sky Contributions from Spectra

The `svdres2` task fits and removes night sky eigenspectra from observed object+sky spectra. Figure 3 shows the process from bottom to top:

1. Original spectrum.
2. Remove continuum and brightest night sky lines from spectrum and rebin spectrum to match dispersion of eigenspectra.

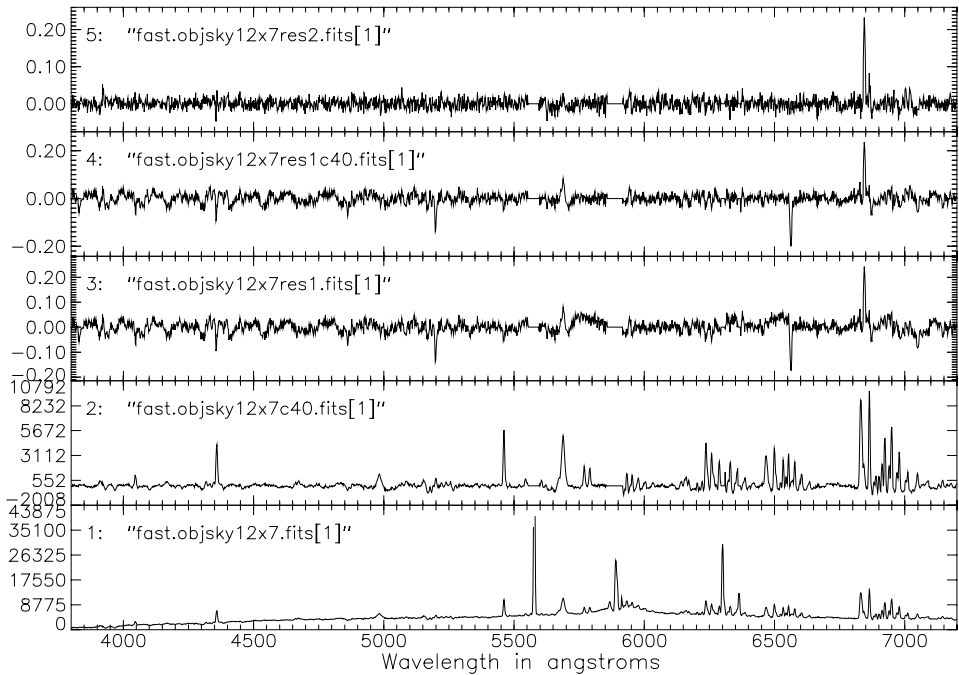


Figure 3. Removal of the night sky contribution to an object spectrum using `svdres2`.

3. Remove major sky spectrum features by fitting first two eigenvectors and remove the fit sky from the spectrum.
4. Remove the continuum from the residual spectrum.
5. Fit the first seven second pass eigenspectra and remove the sky spectrum thus fit to leave a spectrum of the observed object, which may then be run through RVSAO to get a redshift.

5. Obtaining this Software

For the current status of this package, access to source code, additional documentation, and examples, see <http://tdc-www.harvard.edu/iraf/svdfit/>.

References

- Kurtz, Michael J. & Mink, Douglas J. 1998, *PASP*, 110, 934
 Kurtz, Michael J. & Mink, Douglas J. 2000, *ApJ*, 533, L183