

ASTR:4850 - Spectroscopic Data Reduction

1 Introduction

To introduce you basic spectroscopic data reduction, we will analyze spectroscopic data from the Low-Resolution Imaging Spectrometer (LRIS) mounted on the Cassegrain focus of the 10-meter Keck I telescope on Mauna Kea, Hawaii. Because spectroscopic data reduction is a bit complex, this data reduction lab is split into three parts over three weeks:

- Week 1: Wavelength Calibration
 1. construct a master bias and subtract it from all frames
 2. define the positions of the object aperture and the background aperture
 3. extract raw spectra (count vs. x) from an arc lamp exposure at the two apertures
 4. measure the x -positions of the arc lines in the spectra
 5. fit an analytical function to the wavelength (in Angstrom) vs. x -position relation
 6. apply the best-fit function to convert all x -positions to wavelengths to convert raw spectra (count vs. x) to wavelength-calibrated spectra (count vs. λ)
 7. measure the spectral resolution R of the blue and the red spectrographs.
- Week 2: Source Extraction
 1. measure the non-uniform illumination pattern from exposures of dome projection lamps
 2. normalize the illumination pattern and apply it to standard star and target exposures
 3. extract raw spectra at the object aperture and the background aperture for the flat-fielded standard star and target exposures
 4. apply wavelength calibration solutions from Part 1 to convert raw spectra (count vs. x) to wavelength-calibrated spectra (count vs. λ)
 5. interpolate the background spectra to the same wavelength grid as the object spectra
 6. subtract background spectrum from object spectrum to produce the background-subtracted and wavelength-calibrated spectrum
- Week 3: Flux Calibration
 1. convert the background-subtracted spectra from Part 2 (mean count vs. wavelength) to total count rate (total count/exptime vs. wavelength)
 2. select a coarse wavelength grid that avoids strong absorption lines visible in the spectrum of the standard and measure the total count rate of the standard star at these selected wavelengths

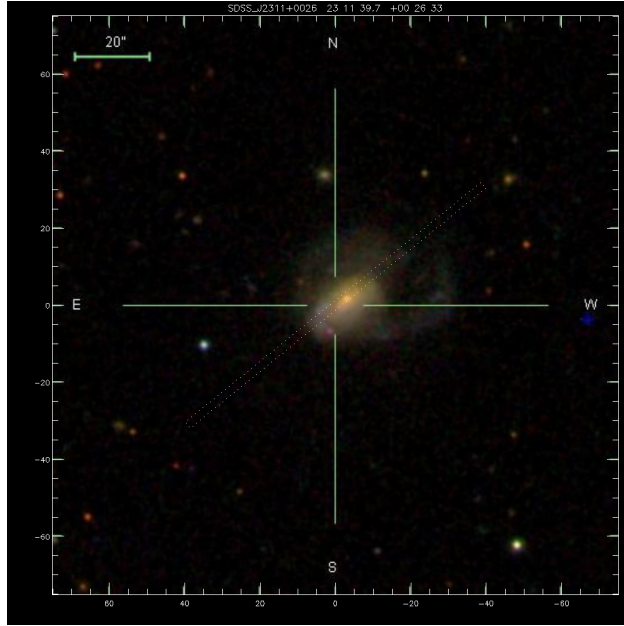


Figure 1: Finder chart of the science field.

3. read the literature spectrum of the standard star and measure the flux densities at the same selected wavelengths calculate the throughput function by dividing the count rate by the flux densities, and interpolate the throughput function to the wavelength grid of LRIS spectrum extracted at the object aperture
4. divide the interpolated throughput function from the standard star and the science target spectra from step 1 to produce the final flux-calibrated spectra (flux density vs. lambda)
5. measure the redshift of the target galaxy by identifying emission lines in the flux-calibrated spectrum.

These are all individual assignments (i.e., not a team project). You'll be asked to submit figures, data files, and/or code for each of the three Parts (10 pts each). You don't need to submit a lab report for each part, but you'll submit one lab report (15 pts) for the entire spectral data reduction exercise. This final lab report will be due in the Final's Week on Wednesday (Note that this course does not have a final exam). I suggest you to draft your lab report as you work on each part because you'll be quite busy in the Final's Week.

2 Observations and Data

2.1 Science Target

The target is a galaxy merger designated as SDSS 2311+0026 simply because it is roughly located at RA = 23h11m and Dec = +0d26m and was selected from the Sloan Digital Sky Survey (SDSS). The data were obtained on 2011 Oct 23 (UT) as part of the radio-selected

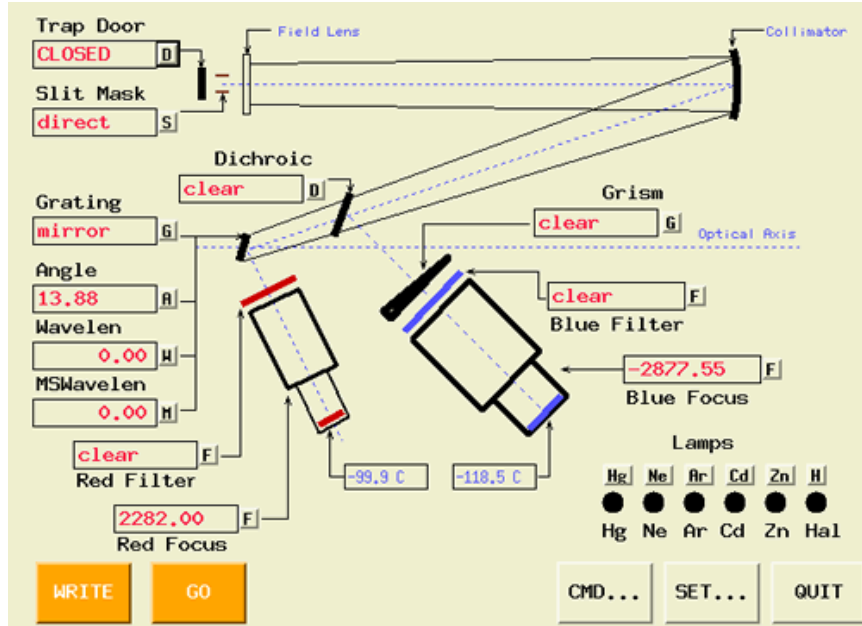


Figure 2: An illustration of the optical layout of Keck LRIS.

binary active galactic nuclei project (2015ApJ...799...72F, 2015ApJ...815L...6F). In Figure 1, the SDSS image shows the science target at the center. The long rectangle shows the intended slit orientation. N is up and E is to the left in this image.

2.2 Instrument Configuration

LRIS is a conventional dual-channel spectrograph that uses a dichroic to split the beam into two separate spectrographs so that blue spectra and red spectra can be taken simultaneously. Its optical layout is illustrated in Figure 2. You can click on this map on the [LRIS homepage](#) to read the specs of the various components.

For this observing run, we had decided on the following instrument setup to achieve a continuous wavelength coverage from 3200 Å to 10,000 Å:

- Dichroic: D560
- Longslit: 1.5-arcsec wide
- Dispersion element in Blue: Grism = 600/4000 (i.e. ,600 l/mm blazed at 4000 Å)
- Dispersion element in Red: Grating = 400/8500 (i.e., 400 l/mm blazed at 8500 Å), tilted to a central wavelength of 8000 Å.

2.3 Data Files

Table 1 explains the content of the FITS files that you have downloaded. The frame numbers refer to the last two digits in the filename. For example, blue frame #11 is b111023_0011.fits, and red frame #28 is r111023_0028.fits.

Table 1: List of LRIS data.

Blue Frames	Red Frames	Object	Comment
9-11	9-11	Bias	zero-exposure frames
12	...	Hg arc lamp	Mercury lamp for blue only
...	14	Ar arc lamp	Argon lamp for red only
25-27	25-27	Dome flat	flat field from an illuminated dome
28	28	BD +28 4211	Standard Star, airmass = 1.09
52	52	SDSS 2311+0026	Science Target, airmass = 1.12

For both the blue and the red spectrographs, the original Keck/LRIS data are stored in multi-HDU FITS files, where the data from each CCD amplifier is stored in a separate FIT extension. For the ease of use, I've assembled the separate arrays into a single array with a Keck routine called `readmhdufits`. In addition, I've transposed and rotated the combined arrays so that wavelengths increases from left to right and the target is positioned in the top half of the slit, like in the illustration below. Finally, the different gains of the four amplifiers in each spectrograph have been multiplied to the subregions of the combined array. This means that the data provided to you have an effective gain of 1 e-/ADU.

3 Procedure for Week 1: Wavelength Calibration

In Week 1 of the Spectroscopic Data Reduction Lab, we aim at (1) understanding the format of the data and (2) calculating the wavelength calibration spectra from the arc lamps.

3.1 Display Spectral Data

Before going into the analysis, it is a good idea to get yourself familiar with the files in the data folder. The file lists tabulated above help you get oriented. You can use DS9 to open the FITS files for the different types of observations. Note that you'll need to adjust the contrast to see the signal. For most cases, I find that the `zscale` function works well – so it gives a good starting point for you to adjust the display limits.

To help visualize the differences among the observations, make a multi-panel figure like Figure 3 to show all types of observations in one figure. In your figure, label each panel with the filename, the observation type (i.e., bias, arc, flat, standard, target), and the exposure time (check the FITS header).

3.2 Bias Subtraction

In the above figure, you probably noticed the discontinuity near the center in each image. It appears in both the blue and the red spectrograph data. These discontinuities correspond to the chip gap between the mosaic'ed CCDs on the camera focal plane, and they are caused by the different bias levels and gains of the CCD electronics. This instrumental artifact can be removed through bias subtraction.

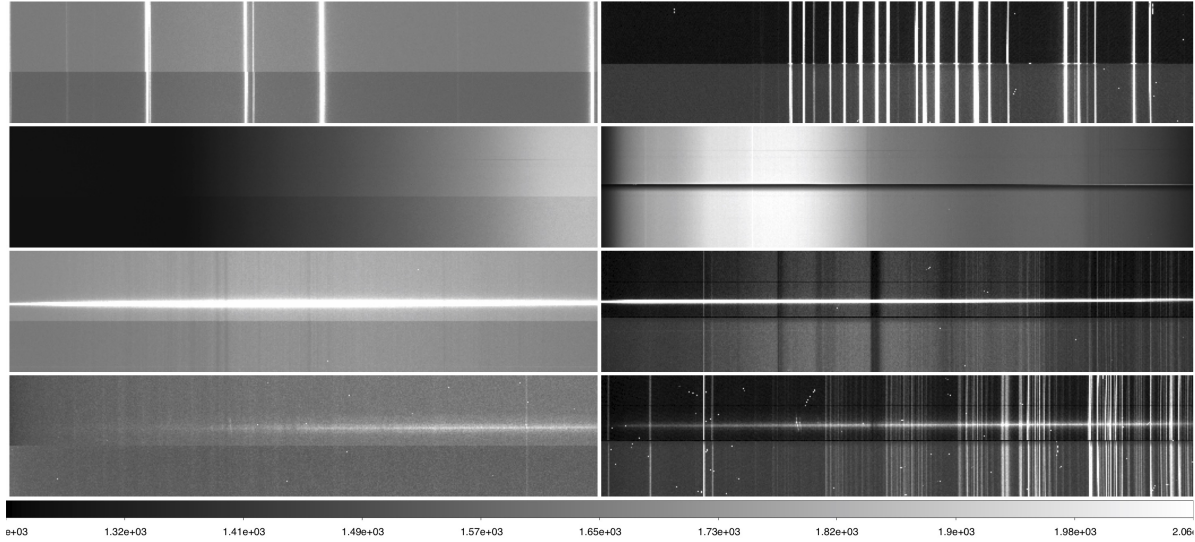


Figure 3: Spectral data from the blue and the red spectrographs are shown on the left and right columns, respectively. From top to bottom are spectral images of the arc lamp, the dome flat, the standard star, and the science target, respectively.

The bias subtraction process is the same as that in the previous imaging labs. But note that there are two cameras in the two spectrographs. For each spectrograph, construct a master bias frame and subtract it from all of the other frames (i.e., arc lamp, flats, standard star, and target). To show the effect of bias subtraction, make the same multi-panel figure using bias-subtracted frames as in the previous section. For a good comparison, use the same display parameters for the two figures.

In the following, we will only use these bias-subtracted frames.

3.3 Wavelength Calibration

The main goal of Part 1 is to obtain the mapping between CCD x-coordinates and wavelengths in Angstrom. To empirically determine this, we will utilize exposures of arc lamps. In the blue, the Mercury (Hg) lamp provides enough emission lines. But Hg doesn't emit enough lines redder than 7000 Å, so we had to use the Argon (Ar) lamp for the red spectrograph. The following discussion focuses on data from the blue spectrograph, but the same procedure is applicable to the Argon frame taken by the red spectrograph.

The wavelengths of Mercury lines are known to us. The [Keck calibration lamps webpage](#) provides the line lists and line identification plots of all the lamps at the facility. For example, the [line list for Hg](#) provides the wavelengths (unit: Å) of Hg lines (in this lab, we'll use the line lists for wavelengths in air instead of in vacuum), and the website also provides the [line identification plot](#).

We now need to use these prior knowledge to determine the relationship between the position along the x-axis of the spectrum images and the wavelength of the radiation. Let's call this relationship the wavelength function (λ vs. x). For an ideal spectrograph, the wavelength function is independent of the CCD y-position because the slit direction is

perfectly aligned with the CCD columns and the grating/grism is exactly perpendicular to the slit. But in reality, there is always some minor dependency on the y-position. As a result, one has to solve the wavelength function for every CCD row. To simplify this process in this lab, we will solve for the wavelength functions only for two extraction positions along the slit – one aligned with the object (object aperture), and another capturing a nearby sky background emission (background aperture). I suggest the following procedure:

- Define spectral extraction apertures with the science exposure
 1. Use DS9 to open the bias-subtracted spectral image of the science target, adjust the contrast so that the target is easily visible.
 2. Draw a rectangle region to enclose most of the flux from the science target, record its central y-position and its y-width (all in Image coordinates).
 3. Draw another rectangle region directly above the science target to set the background aperture. Make sure it doesn't contain any other sources and its y-width is at least 3x larger than that of the object aperture. Record the central y-position and y-width of the background aperture.
 4. Save these DS9 regions and save the DS9 figure.
- Write a Python script to do the rest. To help you get started, an example python code is given at the end of this section.
 1. Load the arc spectral image and extract raw spectra within these two apertures.
 2. Plot the raw spectrum (count vs. x) for the arc spectrum extracted from the object aperture.
 3. Identify the Hg lines by comparing your plot with the line ID plot above, use cursor to measure the rough x-positions of the four strong Hg lines @ 3650.1, 4046.5, 4358.3, 5460.7 Å, record these x-positions in your notes. To achieve better calibration, you want to choose strong lines that cover a wide wavelength range.
 4. Measure the precise x-positions of these lines with a 1D centroid algorithm (which is simply a count-weighted mean). You may want to run the centroid function iteratively to obtain a stable result.
 5. Plot the wavelengths vs. the precise x-positions of the Hg lines and fit a linear wavelength function through the data points: $\lambda = \lambda_0 + D \times (x - x_0)$. Note that the reference pixel, x_0 , can be arbitrarily set to any pixel, so we only fit two parameters: the reference wavelength λ_0 and the dispersion D . Record the set value of 'reference pixel' and the best-fit values of 'reference wavelength', and 'dispersion'. Overlay the best-fit linear wavelength function on the previous plot. Does a linear function adequately describe the data?
 6. Calculate the wavelength differences between the data points and the best-fit linear function at each x-position, and plot this residual as a function of x-position. How can we reduce the residual? That is, how can we improve the wavelength function? What is your estimate of the wavelength calibration error based on the values of these residuals?

7. Apply the best-fit wavelength function to the full x-position array of the raw spectrum to convert x to wavelength.
 8. Plot the wavelength-calibrated arc spectrum as count vs. wavelength. As a sanity check, show vertical lines indicating the wavelengths of all of the arc lines in the line list. If these vertical lines match the peaks of the arc lines, you have got a good wavelength function.
 9. Repeat the procedure for the arc spectrum from the background aperture and compare the differences in the wavelength function.
- Once you're done with the blue spectrograph, adjust the procedure to wavelength calibrate the red spectrograph data. Note that you'll need to use the line list and line ID plot for the Argon lamp.

Below is an example python code that does some of the tasks listed above. Note that it may contain bugs and you'll need to edit most of the numerical values in this example to obtain the correct results.

```

1
2 #####
3 # Spectral Extraction
4 #####
5 # load bias-subtracted arc lamp image
6 arcfile = pyfits.open('redux/b111023_0012.fits')
7 arcimg = arcfile[0].data
8 # this is for the object aperture, you also need to define and extract the
   background aperture
9 y0 = 2100 # center of aperture
10 dy = 36 # half width of the aperture
11 (ny, nx) = shape(arcimg) # dimensions of spectral image
12 # extract a 1-d spectrum by summing across the aperture
13 arc = zeros(nx)
14 for i in range(nx):
15     arc[i] = sum(arcimg[(y0-dy):(y0+dy+1), i])
16 # array of x positions
17 x = 1+arange(nx)
18 # plot extracted spectrum (count vs. x-position)
19 plt.figure(2)
20 plt.clf()
21 plt.xlabel('X Position')
22 plt.ylabel('Counts')
23 plt.plot(x, arc, '-b')
24
25 #####
26 # Wavelength Calibration
27 #####
28 # Hg line wavelength in Ang
29 linew = array([3650.1, 4046.5, 4358.3, 5460.7])
30 # rough line center positions
31 linec = array([ 900, 1500, 2000, 3800])
32 # half-width of centroid window (linec +/- lined)
33 lined = array([ 10, 10, 10, 10])

```

```

34
35 # maximum value in spectrum (for plotting vertical lines)
36 smax = max(arc)
37 # calculate centroids for each line
38 centroid = 0.0*linec
39 for i in range(len(linec)):
40     # array elements inside centroid window
41     k = range(linec[i]-lined[i], linec[i]+lined[i]+1)
42     # calculate centroid position for this line
43     centroid[i] = sum(x[k]*arc[k])/sum(arc[k])
44     # overlay a vertical line on the 1D spectrum
45     plt.plot([centroid[i], centroid[i]], [0, smax], '--g')
46     print 'rough line center, precise line center ', linec[i], centroid[i]
47 # display the plot
48 plt.show()
49
50 # fit for a linear wavelength function
51 # equation: lambda = lam_ref + dispersion * (x - x_ref),
52 # where lam_ref and dispersion are the fitted parameters
53 x_ref = nx/2 # arbitrarily set the reference pixel
54 dispersion, lam_ref, r_value, p_value, std_err = stats.linregress(centroid
    -x_ref, linew)
55 print 'Chosen Reference Pixel = ', x_ref, ' pixel'
56 print 'Best-Fit Wavelength at Reference Pixel = ', lam_ref, ' Ang'
57 print 'Best-Fit Dispersion = ', dispersion, 'Ang/pixel'
58
59 # illustrate the quality of the fit (lambda vs. x)
60 plt.figure(2)
61 plt.clf()
62 plt.xlabel('X Position (Pixel)')
63 plt.ylabel('Wavelength (Ang)')
64 # best-fit linear model
65 y = lam_ref + dispersion*(x-x_ref)
66 plt.plot(x, y, '-b')
67 # show data points used for the fit
68 plt.plot(centroid, linew, 'ko')
69 plt.show()

```

In the code above, the equation for the wavelength calibration is:

$$\lambda = \lambda_{\text{ref}} + D \cdot (x - x_{\text{ref}}) \quad (1)$$

where we set the reference pixel (x_{ref}) as the central pixel in the x-array, and use Scipy's linear least-squares regression method to find the best-fit values for the wavelength dispersion (D) and the wavelength at the reference pixel (λ_{ref}). Note that you can set x_{ref} to be any pixel, even zero.

3.4 Measure Spectral Resolution

A key parameter describing the capability of a spectrograph in resolving spectral features closely separated in wavelength is the spectral resolution R . Defined as $R = \lambda/\text{FWHM}$ where FWHM is the full-width-at-half-maximum in Å of a spectral line at a wavelength

of λ (also in \AA), it is a dimensionless parameter that can be empirically determined by measuring the FWHM of lines in a lamp spectrum or airglow lines from the atmosphere. Spectral resolution not only depends on the grade of the grating but also on the slit width. Choose two unsaturated, relatively isolated, strong emission lines in each of your Hg and Ar arc spectra, and measure their central wavelengths and their FWHMs by either fitting a Gaussian or using simple interpolation method at half of the peak height (note that the baseline level may not be at zero). Now you have the data to calculate R at two wavelengths in both the blue and red spectrographs. Report λ , FWHM, and R of these four measurements in a table and compare your results with the FWHM resolution listed on the [LRIS webpage](#). Note that the website values are for the 1 arcsec slit, so you need to convert the values to the 1.5 arcsec slit used in our observations.

4 Procedure for Week 2: Source Extraction

In Week 2 of the Spectroscopic Data Reduction Lab, we aim at (1) correcting for the non-uniform spatial response and (2) obtaining background-subtracted wavelength-calibrated spectra for the standard star and the science target.

4.1 Spectroscopic Flat Fielding

Unlike imaging data, spectroscopic data have non-uniform responses in both the spatial dimension and the spectral dimension. The two dimensions are almost orthogonal to each other so we can treat them separately. Correcting the non-uniform response in the spatial dimension is called flat fielding, while correcting for the non-uniform response in the spectral dimension is called flux calibration. We deal with the former in this lab, but will deal with flux calibration in Part 3.

In addition, there is yet another non-uniformity in a spectrograph, which is non-uniform response to photons among the millions of CCD pixels. In CCD imaging, the pixel-to-pixel variations are accounted for by the division of the flat field because it is folded into the total response. But in spectroscopy, the pixel-to-pixel variations cannot be corrected by flat fielding in the spatial dimension and flux calibration in the wavelength dimension. Instead, it requires additional flat-field data taken without a slit mask. For simplicity, we skip this additional correction in this lab but we should be aware of this caveat.

4.2 Building a High S/N Flat Field Profile

To empirically determine the non-uniform response across the spatial dimension (i.e., along the slit), we need spectral images of a uniformly illuminated source. This can be achieved, before the night begins, by using a projection lamp to illuminate the dome and take spectral data of the dome using the same instrument setup as in science observations.

Three frames of dome flats for each spectrograph are provided in the data archive that you downloaded in Part 1. You should have subtracted bias from these frames last week. Now inspect the bias-subtracted dome flats with DS9. In the blue dome flats, you should notice that the image is roughly a step function across the vertical (spatial) direction because

the length of the slit is shorter than the field covered by the CCD. For the red dome flats, the spatial dimension roughly matches that of the slit length because only a subregion of the CCDs were read out. This rough step function across the spatial direction shows the non-uniform response across the spatial dimension.

There are large variations across the spectral dimension as well. These are due to a combination of (1) the non-uniform spectral response of the instrument and (2) the spectral shape of the dome projection lamp. Because we don't know the precise spectral shape of the dome lamp (which also varies with time), we cannot use the dome flats to correct the non-uniform spectral response. That is why we have to rely on spectrophotometric standard stars to correct it – a process called flux calibration.

In principle, one could just use the vertical profile along a single CCD column and normalize it to obtain the flat field function, but that would have rather poor S/N, which would introduce additional noise to the science data. To build a high S/N flat field function, it is advised to do the following:

- Median-combine multiple bias-subtracted dome flat images.
- Select a range of CCD columns near the center of the image where there are strong signal (i.e., high ADU counts).
- Stack the spatial profiles within the selected range across the spectral dimension. That is to sum the pixel values along x dimension.
- Normalize the stacked spatial profile at the y-position of the science target (use the center of the object aperture you defined in Part 1).

This normalized profile is your flat-field function. Make a plot showing your flat-field function and submit it.

To correct for non-uniform spatial response in the standard star and science target observations, you'll need to divide the flat-field function out from every column of the bias-subtracted spectral images.

Below is an example python script to build the flat field function and apply it to the bias-subtracted standard star spectral image.

```
1 #####
2 # Build Flat Field
3 #####
4 # load bias-subtracted master flat
5 fitsfile = pyfits.open('redux/masterflat.fits')
6 flating = fitsfile[0].data
7 # define the region along the X-dimension to stack the spatial profiles
8 x1 = # left boundary X-position
9 x2 = # right boundary X-position
10 (ny, nx) = shape(flating) # dimensions of spectral image
11 # stack spatial profiles
12 flat = zeros(ny)
13 for j in range(ny):
14     flat[j] = mean(flating[j, x1:x2])
15 # normalize flat function
16 y0 = # Y position of the science object
```

```

17 nflat = flat/flat[y0]
18 # array of y positions
19 y = 1+arange(ny)
20 #####
21 # Apply Flat Field
22 #####
23 # load bias-subtracted standard star
24 fitsfile = pyfits.open('redux/b111023_0028_b.fits')
25 std_b = fitsfile[0].data
26 std_bf = std_b
27 for i in range(nx):
28     std_bf[:,i] = std_b[:,i]/nflat
29 # write to a new FITS file
30 hdu = fits.PrimaryHDU(std_bf)
31 hdu.writeto('redux/b111023_0028_bf.fits')

```

4.3 Object and Background Spectral Extraction

Once the non-uniformity along the spatial dimension is corrected, we can extract spectra at the object and the background apertures, and subtract the two to recover counts from source only (in the next subsection). The process is similar to the spectral extraction for arc exposures in Part 1. The main differences are: (1) we start from flat-fielded bias-subtracted spectral images that you obtained in the last section, (2) instead of sum, we use mean to extract object spectra and median to extract background spectra, and (3) we now have wavelength functions for both object and background apertures from Part 1. Using median for the background aperture has two advantages: (1) it helps reject cosmic rays and (2) the extracted spectra are now in units of count per pixel (instead of count per aperture) so that the aperture size difference is removed. We use mean instead of median for the object aperture because the spatial profile of an object cannot be flat, unlike the spatial profile of sky background.

Important note: Make sure that the object apertures used to extract blue and red spectra have exactly the same width in arcsec. Because the blue and the red spectrographs have the same plate scale (0.135 arcsec/pix; see [LRIS detectors webpage](#)), the widths in pixel unit are the same as well. This requirement is necessary to properly combine the flux-calibrated spectra from the two spectrographs in Week 3.

Below is an example python script, the arrays `wave_obj`, `count_obj` form the flat-fielded wavelength-calibrated spectra extracted at the object aperture.

```

1 #####
2 # Spectral Extraction
3 #####
4 # load flat-fielded bias-subtracted standard star image
5 # (as indicated by the suffix _bf in the filename)
6 fitsfile = pyfits.open('redux/b111023_0028_bf.fits')
7 stdimg = fitsfile[0].data
8 # define the object aperture
9 y0 = # center of aperture
10 dy = # half width of the aperture
11 (ny, nx) = shape(stdimg) # dimensions of spectral image

```

```

12 # extract raw spectrum
13 # 1. taking mean for object aperture
14 # 2. taking median for background aperture
15 count_obj = zeros(nx)
16 for i in range(nx):
17     count_obj[i] = mean(stdimg[(y0-dy):(y0+dy+1)], i)
18 # array of x positions
19 x = 1+arange(nx)
20 # wavelength function at object aperture
21 xcen =      # x-position of the reference pixel
22 disp =      # dispersion in Ang/pix
23 wcen =      # wavelength in Ang of the reference pixel
24 # convert x to wavelength
25 wave_obj = wcen + disp*(x-xcen)

```

You may want to write a function to do the extraction because you'll need to extract the object and background spectra for the standard star and the science target and for both the blue and the red spectrographs. You'll obtain a total of 8 spectra:

- Standard Star Spectra
 - blue object
 - blue background
 - red object
 - red background
- Science Target Spectra
 - blue object
 - blue background
 - red object
 - red background

Make a plot for the standard star to show the spectra. In a single plot of count per pixel vs. wavelength (in Angstrom), use different line colors to differentiate spectra from the object and background and spectra from the blue and the red spectrographs.

Make the same plot for the science target.

4.4 Background Subtraction

Our last task is to remove signals of the sky background from the object spectra. To do this, we need flat-fielded wavelength-calibrated spectra extracted at two apertures: object and background. In the previous section, by making use mean and median in the extraction, we have already made sure that both object and background spectra are normalized by the size of the aperture along the y-dimension (i.e., spatial dimension). But before we can subtract the two, we need to make sure the spectra are on the same wavelength grid (recall that the wavelength function has a minor dependency on the y-position). A simple way to

achieve that is to interpolate the background spectrum to the same wavelength grid as the object spectrum. Below is a sample python script to do the interpolation and the subsequent subtraction.

Add the background subtracted spectra to your spectral plots in the previous section. Submit the final plots.

```
1 # import scipy module
2 from scipy.interpolate import interp1d
3 # interpolate background spectrum to the wavelength grid of object
  spectrum
4 f_count = interp1d(wave_bkg, count_bkg, fill_value=0, bounds_error=False)
5 count_bkg_matched = f_count(wave_obj)
6 # subtract background spectrum from object spectrum
7 count_obj_sub = count_obj - count_bkg_matched
```

5 Procedure for Week 3: Flux Calibration

In Part 3 of the Spectroscopic Data Reduction Lab, we aim at (1) empirically determining the non-uniform spectral response of the system from the standard star observation, and (2) correcting its effects on the observed spectrum to obtain the final, flux-calibrated spectra.

5.1 Background-Subtracted Spectra

Before you proceed, make sure that your background-subtracted spectra of the science target from Part 2 look like the ones shown in Figure 4 (make sure your wavelength ranges are correct also). If not, probably something went wrong in Part 1 and/or Part 2 and you need to fix the problems so that they don't propagate into Part 3.

5.2 System Throughput

The system throughput is defined as the ratio between the collected count rate and the flux density of the same source and at the same wavelength:

$$\text{throughput} = \frac{\text{count rate}}{\text{flux density}} \quad (2)$$

The system throughput is the product of three throughput functions: those of the atmosphere, the telescope, and the spectrograph. It is a function of wavelength and has a unit of e-/s per erg/s/cm²/Å. It also varies with airmass because of atmospheric extinction. To empirically determine the throughput function, observers typically observe a few standard stars throughout a given night at different airmasses. The most commonly used spectrophotometric standard stars for the northern observatories are listed here (Links to an external site.) along with plots of their spectra and finding charts.

To determine the throughput function, we need to (1) measure the total count rate vs. wavelength and (2) take literature values of flux density vs. wavelength. Once the two are sampled on the same wavelength grid, the throughput is simply a matter of taking the ratio of the two.

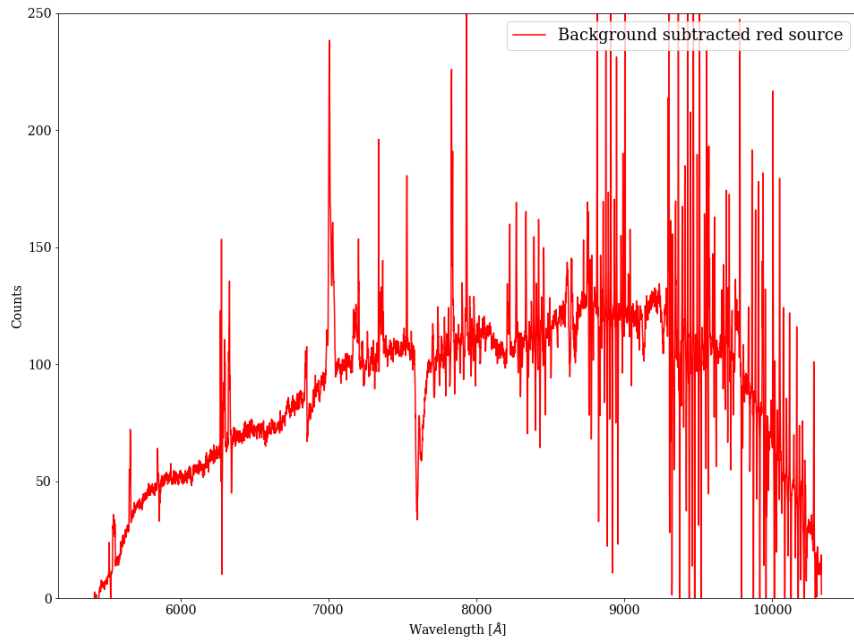
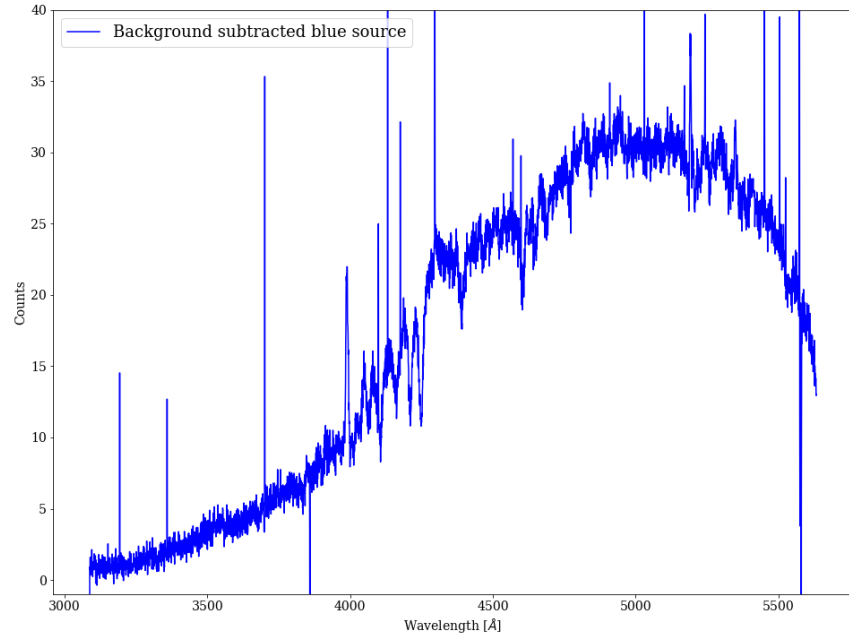


Figure 4: Sky subtracted spectra.

Since the throughput function is solely determined by the atmosphere, the telescope, and the spectrograph, it is independent of the observed source. Therefore, the function calculated from a standard star can be directly applied to the science target, for which literature flux measurements usually don't exist (otherwise, we wouldn't be observing it). Note that for simplicity, we ignore the airmass-dependency of the throughput by assuming that the target and the standard star were observed at the same airmass.

5.3 System Efficiency

Alternatively, throughput can be expressed as an efficiency. The system efficiency is a dimensionless parameter similar to the Quantum Efficiency of a CCD. The efficiency is defined as the ratio between (1) the number of photons detected by the detector per unit time and (2) the number of photons collected by the telescope per unit time. Given that A is the collecting area of the telescope primary mirror and $h\nu = hc/\lambda$ is the photon energy at the wavelength, the number of photons collected by the telescope per unit time is simply flux density multiplied by A and then divided by $h\nu$. Therefore, the efficiency is related to the throughput as in the following equation:

$$\text{efficiency}(\lambda) = \frac{\text{count rate}(\lambda)}{\text{flux density}(\lambda)} \cdot \frac{hc/\lambda}{A} = \text{throughput}(\lambda) \cdot \frac{hc/\lambda}{A} \quad (3)$$

The system efficiencies for various LRIS configurations are given on [the LRIS webpage for the dispersive elements](#). For example, Figure 5 shows the efficiency of LRIS blue with the 600 l/mm grism. It shows that the system is most efficient at $\sim 4200 \text{ \AA}$, where it can detect $\sim 50\%$ of the incoming photons.

Lastly, throughput can also be expressed as the effective collecting area, A_{eff} , which is defined as:

$$A_{\text{eff}}(\lambda) = \frac{\text{count rate}(\lambda)}{\text{flux density}(\lambda)} \cdot hc/\lambda = \text{throughput}(\lambda) \cdot hc/\lambda \quad (4)$$

In summary, throughput, efficiency, effective area are equivalent quantities that characterize the combined wavelength-dependent response of the atmosphere+telescope+instrument system.

5.4 Calculate Total Count Rate Spectra

In Part 2, you obtained background-subtracted, wavelength-calibrated spectra for both the standard star and the science target. These spectra record the average count per CCD row inside the object aperture as a function of wavelength. Because 1 CCD row corresponds to only 0.135 arcsec on the sky (the CCD plate scale listed here), which is much smaller than the seeing FWHM (1-2 arcsec), these spectra only provide partial signal from the source. But, to calculate the throughput, the total signal is needed (this is also why a wider slit is typically used for standard star observations).

Therefore, to compute the throughput function, we first need to multiply these average counts by the number of CCD rows included in the object aperture to obtain the total counts from the source; then, we divide the total counts by the exposure time (which is kept in the FITS header keyword TTIME) to obtain the total count rates.

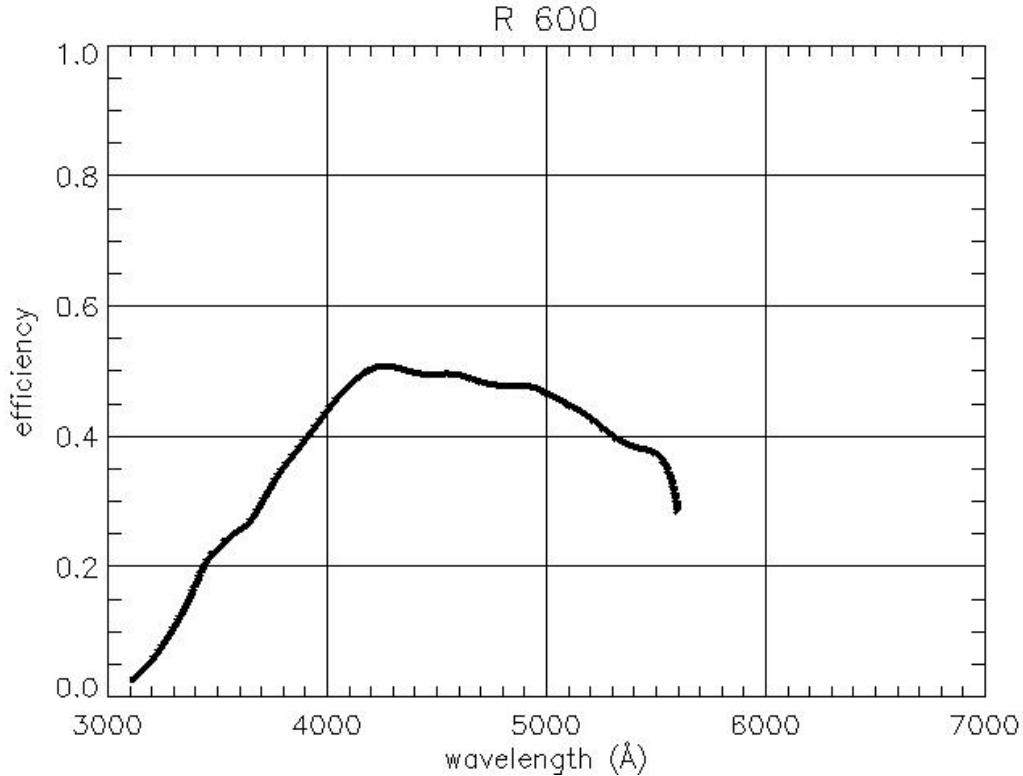


Figure 5: System efficiency of LRIS blue with the 600 l/mm grating.

Plot the total count rates vs. wavelength for both the standard and the science target. Show both the blue and the red spectra. As noted in Part 2, make sure that the widths of the object apertures are the same for the blue and the red spectra.

5.5 Select a Coarse Wavelength Grid

Because the literature spectrum of the standard star is taken by different spectrographs, it is unlikely to have the same spectral resolution as the LRIS spectra. So if we directly divide the LRIS count rate spectrum by the literature spectrum, the mis-matches in spectral resolution will cause sharp artifacts near sharp spectral features such as the absorption lines. To avoid this problem, one could convolve a wavelength-dependent Gaussian kernel to the higher-resolution spectrum to match the lower-resolution spectrum. But that's a bit difficult to implement because the spectral resolution function is not precisely known. Instead, one usually takes a simpler approach.

In the efficiency function shown above, you notice that it smoothly varies with wavelength. This lack of any sharp wavelength-dependency informs us that the throughput function only need to be measured on a rather coarse wavelength grid (say, every 100 Å). So to avoid the sharp artifacts near absorption lines, one could simply choose a coarse wavelength grid that avoids the few absorption lines. For example, you may select the following coarse grid for the blue spectrum:


```
1 wave_coarse = np.arange(3300, 5600, 127.7)
```

But note that this wavelength grid doesn't have to be evenly spaced.

Once you've decided on the coarse wavelength grids for the blue and red spectra, indicate the selected wavelengths in the count rate spectra of the standard star. You can use either dashed vertical lines or overlay colored squares on the spectra as I did in the lecture slide.

Briefly, our plan is to (1) use an interpolation method to obtain the total count rate and the flux density at some widely spaced wavelengths, (2) calculate the throughput values on this coarse wavelength grid, (3) use a Spline interpolation method to evaluate the throughput function on the fine wavelength grid of the LRIS spectra, and (4) divide the LRIS count rate spectra by this interpolated throughput function to obtain flux-calibrated spectra.

5.6 Read Literature Spectrum of the Standard Star

For every spectrophotometric standard, previous papers in the literature have measured its total flux density vs. wavelength. These spectra have been corrected for all of the instrumental and atmospheric effects except the spectral resolution. For your ease-of-use, I provide a FITS file of the literature spectrum of BD+28 4211 is provided as a FITS file (bd284211.fits).

The data of the FITS file give the flux density array in $\text{erg/s/cm}^2/\text{\AA}$. The wavelength array is not provided directly. That is because wavelengths are usually sampled in a linear array, we only need to know the dispersion, the reference pixel, and the wavelength of the reference pixel. These are saved in the FITS header keywords CDEL T1, CRPIX1, and CRVAL1, respectively. They are similar to the WCS keywords, but for the wavelength dimension. With the values of these FITS keywords, you can calculate the wavelength array in Angstrom. In python, you can do something like this:

```
1 hdu = pyfits.open('bd284211.fits')
2 flux_lit = hdu[0].data # flux density array in erg/s/cm2/A
3 x_pos = 1+arange(len(flux_lit))
4 CRVAL1 = hdu[0].header['CRVAL1']
5 CDEL T1 = hdu[0].header['CDEL T1']
6 CRPIX1 = hdu[0].header['CRPIX1']
7 wave_lit = CRVAL1 + CDEL T1 * (x_pos - CRPIX1) # wavelength array in Ang
```

Plot the literature spectrum and use vertical lines to mark the wavelength ranges of the blue spectrograph and the red spectrograph.

5.7 Calculate Throughput Function

The system throughput is simply the ratio between the total count rate and the flux density at the same wavelength. To calculate it on the coarse wavelength grid defined earlier to avoid spurious features, we need to obtain the total count rates and the flux densities at these wavelengths. To do this in python:

```
1 from scipy.interpolate import interp1d
2 f_count_rate = interp1d(wave_std, count_rate_std, fill_value="extrapolate")
3 f_flux = interp1d(wave_lit, flux_lit, fill_value="extrapolate")
4 throughput_coarse = f_count_rate(wave_coarse)/f_flux(wave_coarse)
```

5.8 Apply Throughput Function

Before we can apply the throughput function to the science target, we need to evaluate it to a finer wavelength grid to match that of the LRIS spectra. After that, to obtain the flux-calibrated spectra is simply a matter of division. To obtain a smooth throughput curve, you can use a 3rd-order spline interpolation method. In python:

```
1 f_throughput = interp1d(wave_coarse, throughput_coarse, kind='cubic',
    fill_value="extrapolate")
2 throughput = f_throughput(wave_obj)
3 flux_tgt = count_rate_tgt / throughput
4 flux_std = count_rate_std / throughput
```

Plot the throughput function, show both the measurements on the coarse grid and the interpolated results in one figure.

Plot the flux-calibrated standard star spectra (blue & red together) and compare them with the literature spectrum by overplotting the latter in a different line style or color. Label the curves using legend. As in all of the plots, include the correct units in the titles of axes.

Plot the flux-calibrated science target spectra (blue & red together).

5.9 Measure Redshift and Distance of the Galaxy

The final task is to estimate the redshift of the galaxy and use it to infer its cosmological distance. Identify the following strong emission lines in the flux-calibrated science target spectrum:

```
1 # The first column lists the rest-frame wavelengths in Angstrom
2 3726.00 [O II]
3 3729.00 [O II]
4 4340.47 H gamma
5 4861.33 H beta
6 4363.00 [OIII]
7 4959.00 [OIII]
8 5007.00 [OIII]
9 6548.4 [N II]
10 6562.85 H alpha
11 6583.9 [N II]
12 6717.3 [S II]
13 6731.5 [S II]
```

Shift their observed wavelengths by multiplying their rest-frame wavelengths by (1+redshift) until you find a good match with their observed wavelengths in your spectrum.

Plug in the redshift in [the cosmological calculator](#) to estimate the luminosity distance to the galaxy. Use the default values of Hubble constant and Omegas and assume a flat Universe. How many times farther is this galaxy than M31 (the Andromeda galaxy) to us?

To illustrate that you have measured the correct redshift, show the redshifted wavelengths of the emission lines in the previous plot, e.g., with dashed vertical lines. Label the emission lines at their redshifted wavelengths, and label the redshift and the luminosity distance in the plot.

Congratulations! You are now done with our final lab.