Antony Sikorski
Phys 164
Professor Konopacky
11/10/2020

Physics 164 Lab #2: Astronomical Spectroscopy

Authors: Antony Sikorski, Lucas Scheiblich
Contact: asikorsk@ucsd.edu, lscheibl@ucsd.edu
10, November, 2020

Abstract

Spectroscopy is an essential tool that has been utilized throughout various fields of science due to the abundance of information that can be received from observing the study of light's interaction with its surroundings. In the particular case of astronomy, spectroscopy is used to identify the composition of celestial bodies, which allows us to further deepen our understanding of their properties. In this experiment, a USB-2000 spectrometer is used to collect data of discharge lamps, on which methods of formatting and reduction are tested to produce a method to effectively extract a one-dimensional spectrum. These methods consist of: finding the centroids (peaks) of our spectra, comparing them to theoretical wavelengths, using a linear least squares fit and estimating its error, and using this wavelength calibration to produce a final spectra. The KAST spectrograph from the 3 meter Shane telescope at Lick Observatory is then used to collect raw spectral science frames of both the Zw 229-015 Seyfert Type 1 Galaxy and the 3C079 Seyfert Type 2 Galaxy. These science frames are then reduced via the previously developed methods so that a clean wavelength vs intensity graph can be produced to potentially be studied in later experiments, which could theoretically identify the primary chemical composition of the different components of these galaxies.

Section 1: Introduction

In the late 17th century, Isaac Newton coined the term, "spectrum", to represent the rainbow of colors that was produced when a white light was shined upon a prism. Since then, the study of the way light interacts with matter grew exponentially, and has become known as spectroscopy. Later in the 19th century, Gustav Robert Kirschoff invented the first device known as a spectroscope. He invented the device to study the emission spectra of hot objects, and after much observation, came up with three pivotal laws regarding the spectra of different objects. The first law states that a hot, dense source, such as a blackbody, emits a continuous spectra, meaning that it emits a full range of all wavelengths. Secondly, a hot, transparent gas produces an emission line spectrum, where only specific wavelengths of radiation are emitted. Finally, a cool gas in front of a hot light source, produces an absorption line spectrum, where most of the different wavelengths are emitted, with specific ones missing. These laws allowed for pivotal discoveries in astronomy, where stars could start to be classified as their spectra were examined to determine their chemical composition.

This research developed at a rapid pace, and leads us to the state of astronomy today, where the classification of celestial bodies has been significantly refined and improved by modern day technology. Now, we have massive telescopes with apertures spanning 5-10m across, utilizing technology such as two dimensional integral spectrographs to take high resolution surveys of the universe in order to gather information about spectra of multiple sources at once. Even these extremely complex machines utilize the same, basic process. Light from celestial objects travels down into the telescope, where it is beamed through a narrow window to a collimating mirror, which ensures that all of the rays of light are parallel as they travel on their way to the diffraction grating. The diffraction grating disperses light into its different wavelengths by reflecting each wavelength at a slightly different angle. This light is then reflected onto a photodetector, such as a CCD (Charge coupled device), which allows the computer to read the different intensities of the wavelengths.

In this experiment, we use two spectroscopic devices in order to observe common discharge lamps and two galaxies. These two galaxies are the Zw 229-015 Seyfert Type 1 Galaxy and the 3C079 Seyfert Type 2 Galaxy (**Figure 1**). Seyfert galaxies are generally highly active galaxies that emit a large amount of infrared radiation, with their most luminous component being their core. The difference in the galaxy type comes primarily from the speed at which they rotate, with Type 1 galaxies rotating at a significantly higher velocity than Type 2 galaxies. In theory, both of their cores are similar to quasars, with very strong emission lines, which our experimental observations agree with.

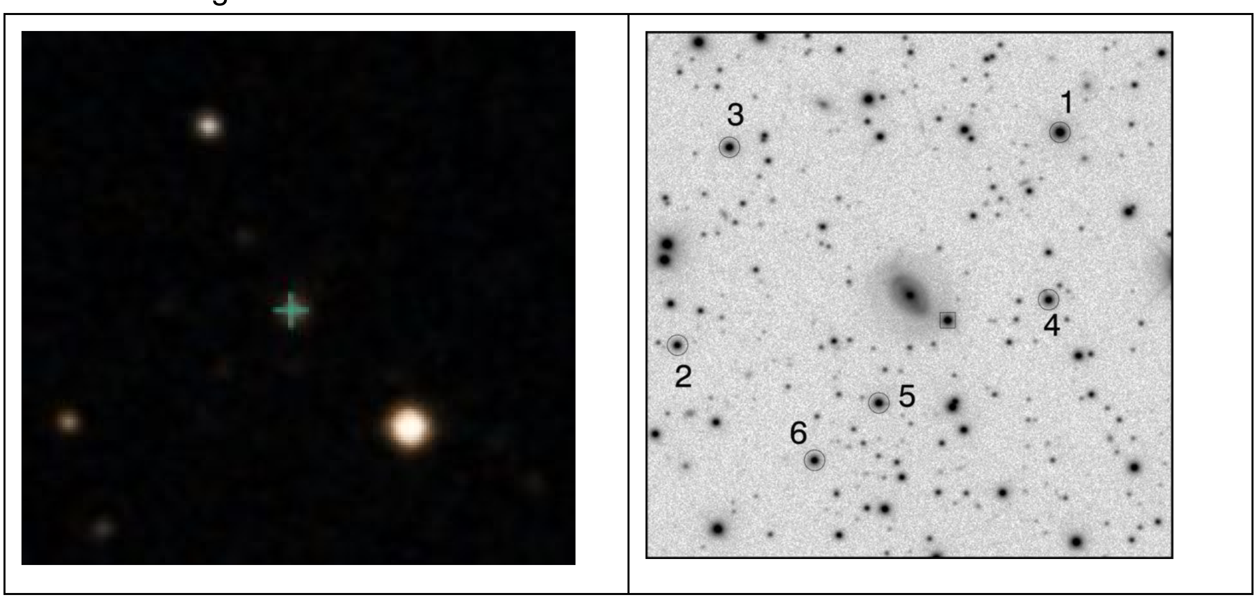


Figure 1: 3CO79 (left) is shown as a bright dot in the bottom right of the image, and the Zw 229-015 (right) in the center of the image.

Section 2: Observations and Data

The equipment used for this lab consists of two spectroscopic devices: the Ocean Optics USB-2000 hand held spectrometer, and the KAST spectrograph at Lick Observatory. The KAST spectrograph is a component to the 3 meter Shane telescope. The data for the USB-2000 was taken Winter 2020, while the KAST data was taken in October of 2013. The USB-2000 was used for observing different elemental discharge lamps, who's files would be used for reference, while the KAST spectrograph was utilized for the taking of two dimensional science frames of our galaxies. The different files used in the experiment are shown below in **Table 1**:

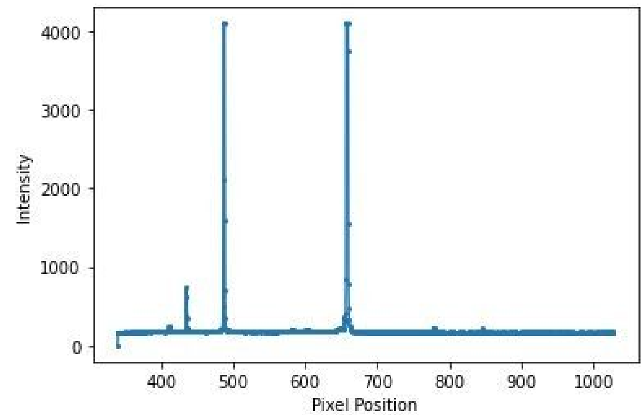
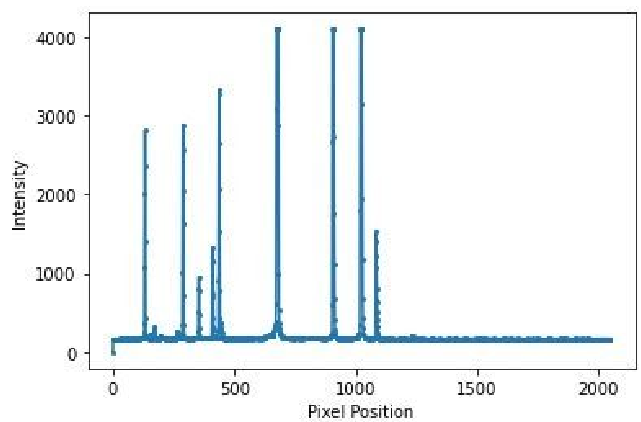
Object	**File**	**Device**	**Slit Separation**	**Year**
Zw-229-015	b151.fits	KAST	4"	2013
3C079	b158.fits	KAST	4"	2013
Hydrogen	Hydrogen01870.txt	USB-2000	NA	2020
Helium	groupc-Helium02190.txt	USB-2000	NA	2020
Mercury	groupc-mercury01590.txt	USB-2000	NA	2020
Neon	groupc-neon01250.txt	USB-2000	NA	2020
Lamp	groupc-lamp01690.txt	USB-2000	NA	2020

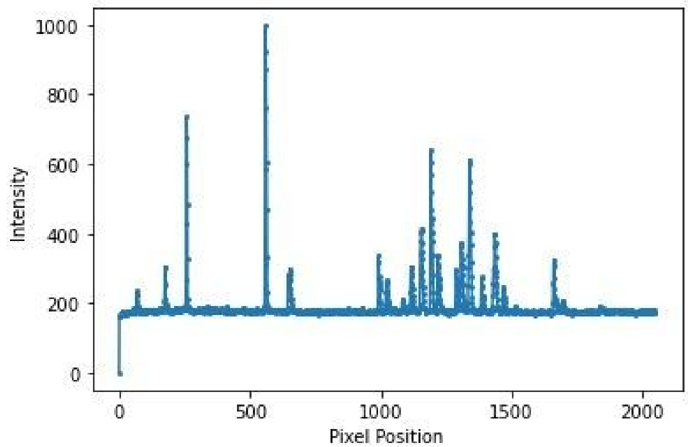
Table 1: Files and Data used in the experiment

The data from the USB-2000 device was used in order to develop data reduction methods that allowed for thorough analysis of the science frames taken by the KAST spectrograph. Since the USB-2000 data was already converted to a 1-dimensional text file, it was simple to plot the spectra as a function of pixel position vs intensity (**Function 2**). The plots of each separate element and object observed by the USB-2000 are shown below.

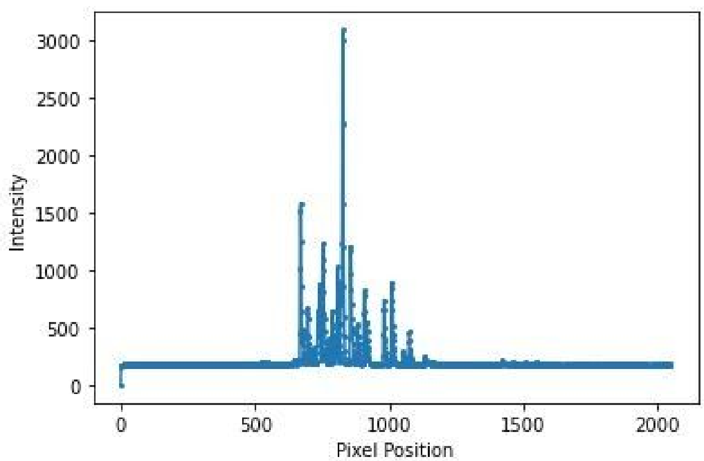
Antony Sikorski
Phys 164
Professor Konopacky
11/10/2020

Figure 2: Non-reduced spectra of each element observed by the USB-2000 plotted as a pixel position vs intensity plot.

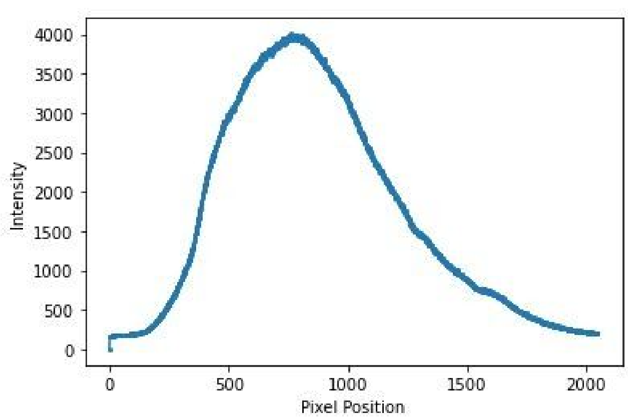
Hydrogen

Helium

Mercury


Intensity
Pixel Position

Neon


Intensity
Pixel Position

Lamp


Intensity
Pixel Position

All of the elements emit an emission line spectra except the lamp, which was excluded from further calculations because it quite obviously emitted a continuous spectra, and would not be a good reference point for our galaxies. The lamp produces this spectrum because it's light comes from a heated coil which acts similarly to an ideal blackbody, which would not be an effective approximation for celestial objects that most likely produce an emission line spectra.

Section 3: Data reduction and Methods

The data reduction for the Ocean-Optics spectrometer was fairly simple due to the fact that we can assume an even excitation across the entire detection surface of the device. This way, there was no need to calibrate the device via dome flats, which will be used to calibrate the KAST spectrograph. The bias is accounted for by setting a lower bound for acceptable data when finding the centroids of the spectra produced by the device. Additionally, as mentioned previously, the lamp was excluded from further calculations because it would not provide an effective approximation for the objects we were studying with KAST.

When using the KAST spectrograph, the data reduction was significantly more involved. The raw science frames that were taken by KAST had serious errors, so it was necessary to use both dome flats and bias frames in order to reduce the data so that the centroids could properly be identified. Initially, each science frame had an overscan region, which was removed by setting an upper bound for the data. Dome flat frames were taken to demonstrate and standardize the variation in intensity of each pixel on the detector, while bias frames with zero exposure accounted for the high probability that there was still some form of information recorded without ever exposing the detector. First, the bias must be subtracted from all of the other frames (science & dome flat). Then, a normalized flat must be created by dividing the flat frame by the median of the flat frame. The science frame was then divided by this normalized flat frame in order to produce a much cleaner image. The function that performs this process via Python is **Function 5**. Unfortunately, in the case of our lab, the reading on the science frames was very weak, and subtracting the bias frame from the science frame often occurred in creating a value very close to zero. In order to account for strange, near-zero errors, the bias flats had to be divided by a number very close to 1, otherwise, the data was nearly unmanageable.

$$Science_{reduced} = (Science - Bias)/((Flat - Bias/median(Flat - Bias))$$

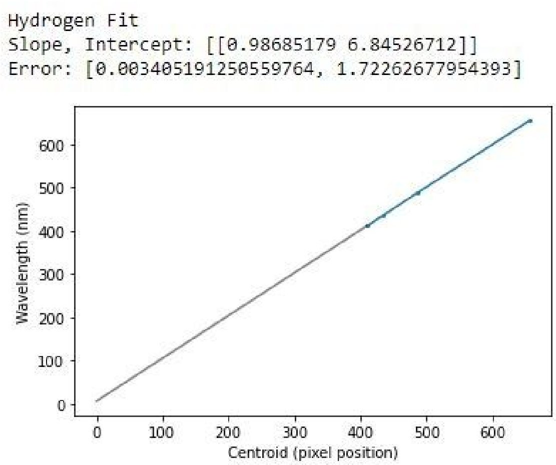
Equation 1: The process for creating a reduced science frame is demonstrated in the above equation, where the frame variables are quite obviously named.

Section 4: Calculations and Modeling

Initially, the data from the USB-2000 spectrometer was analyzed to create methods that would be applicable to the science frames obtained from KAST. To analyze the USB data, we began by identifying the centroids of each element (**Function 1**), which finds peaks in the spectra by analyzing local maxima. Then, wavelengths corresponding to the pixel intensities at which centroids were detected were found by using the NIST database. These were plotted against each other, which produced a relatively linear result, showing a decent correspondence between the chosen wavelengths and the detected centroids. In order to judge the accuracy of our fit, a linear least squares fit (**Function 3**) was used, and the error of our fit was calculated using a least squares fit error function (**Function 4**). The data below shows that our centroid function is quite precise, with our error between the centroids and chosen wavelengths ranging between 0.003% and 20%.

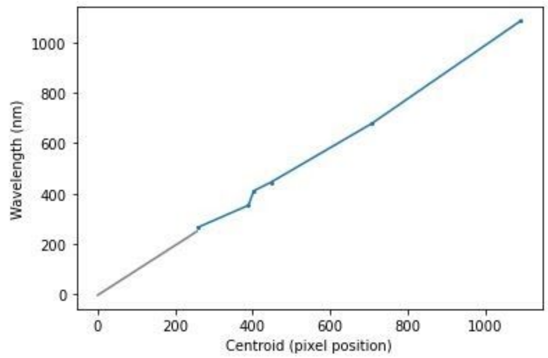
Figure 3: The linear least squares fit is shown on the plotted centroids vs chosen wavelengths, along with the actual best fit function and the error of the fit.

Hydrogen



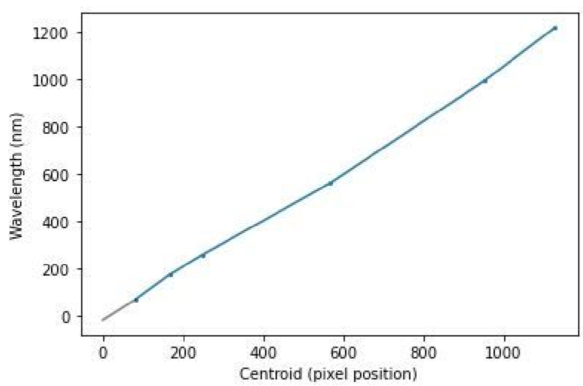
Helium

```
Helium Fit
Slope, Intercept: [[ 0.98804948 -2.46241326]]
Error: [0.030031642999770614, 18.471238251233068]
```



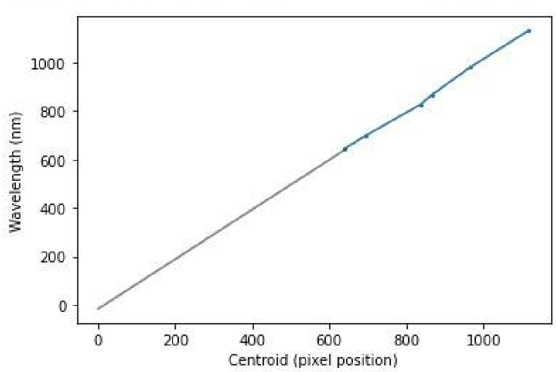
Mercury

```
Mercury Fit
Slope, Intercept: [[  1.07611609 -18.61832225]]
Error: [0.022516000442013254, 14.820268606972371]
```



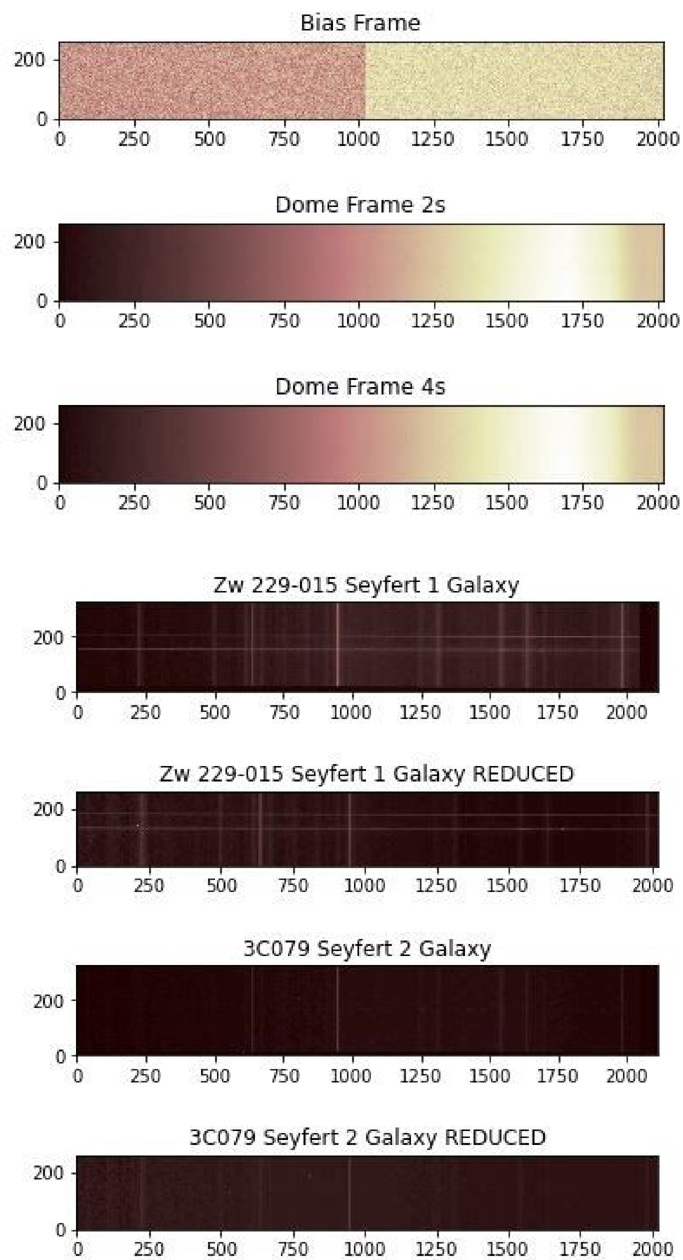
Neon

```
Neon Fit
Slope, Intercept: [[  1.0255357  -17.09323737]]
Error: [0.02323538484490006, 20.177055218310354]
```



After determining that the centroid finding function (**Function 1**) was accurate within an acceptable margin of error, the analysis for the KAST data began. Initially, the data was reduced using **Function 5**, and then plotted in order to view the two dimensional science frames. This is shown below in **Figure 4**.

Figure 4: Bias frame, Dome Frames for varying slit separation, and both raw and reduced science frames of our two galaxies are shown in two dimensional form.



It is not visible to the naked eye, but the two dimensional data plotted in **Figure 4** is actually skewed by a minimal amount. The reduced science frames can be linearized using **Function 6**. This takes a 1 dimensional slice of the data, and allows for a slight skew, which negates additional sources of error in the resulting pixel position vs intensity plot. The arc frames are analyzed the same way so that the resulting one dimensional array of the science frames can be plotted against it. The process for the arc frame is shown in **Figure 5**. This allows for the conversion from a pixel position vs intensity plot to a spectral wavelength vs intensity plot. Our linear least squares function is then applied to this plot. The resulting slope of the linear least squares fit is used to scale the final, calibrated spectral plots in **Figure 6**. The intercepts of the fit function are responsible for the positive offset on the graphs.

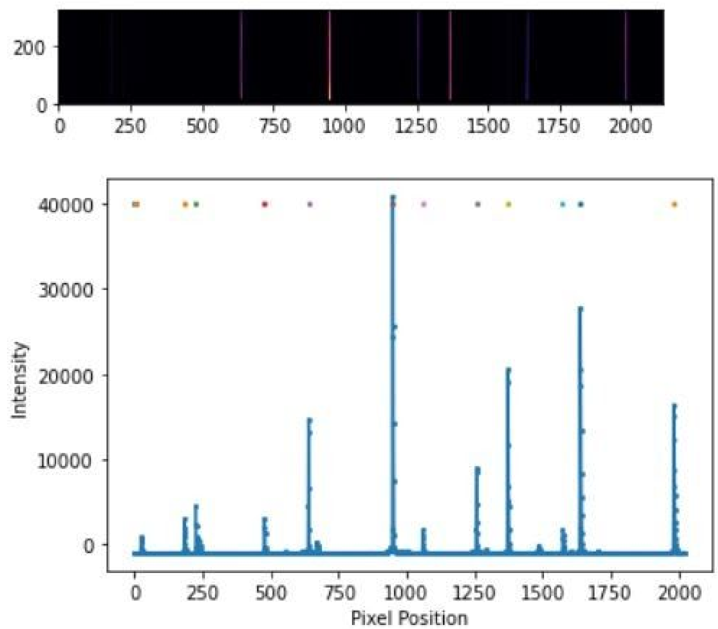
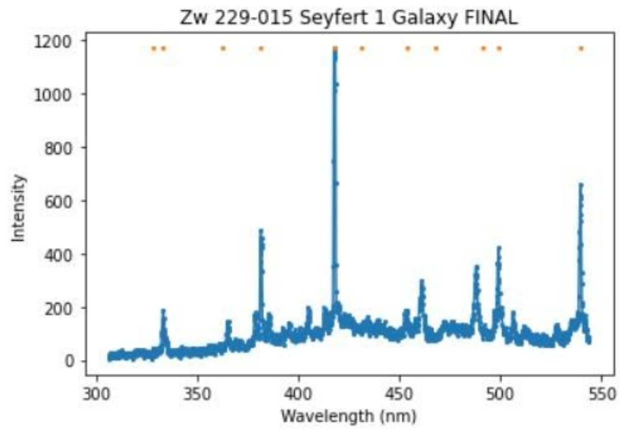


Figure 5: (Above) Two dimensional science frame of the arc lamp spectra. (Below) One dimensional spectra plotted for the arc lamp.



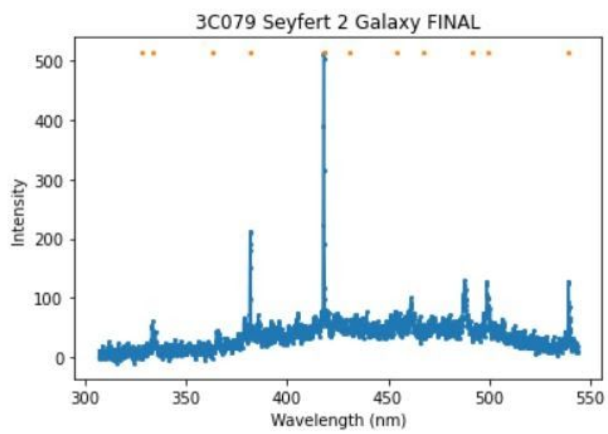


Figure 6: Calibrated spectral plots for both observed galaxies.

Section 5: Discussion

The USB-2000 data was exceptionally effective in developing methods to reduce the data and create a spectra in units of wavelength. The detected centroids plotted against the expected wavelengths that were taken from NIST had a very linear trend, with a linear least squares fit error ranging between approximately 0.003 and 20%. It seems the centroid function (**Function 1**) was accurate when finding the emission centroids, because the numbers show a similar spacing to the ones found from the NIST database. After linearizing (**Function 6**) the KAST data and plotting it against the arc frames, the error ranges from 0.004 to 5%, meaning that our wavelength calibration was very well done. It appears that our final, calibrated spectral plots (**Figure 6**) of the two galaxies are most likely an effective representation of their actual spectra. The shape of the plots is consistent with theoretical expectation for both Seyfert Type 1 and 2 galaxies, because they both give off broad and highly ionized emission lines from their exceptionally luminous cores. In addition, Seyfert Type 1 galaxies are expected to have broader and more intense emission lines than their Type 2 cousins, which is consistent with our observations (**Figure 6**).

If the experiment were to be repeated, there are several areas of improvement that I would suggest. To begin, using the NIST database was an inefficient way to find corresponding wavelengths to test the detected centroids again. The NIST database had a massive range of wavelengths, and had specific ionizations of the elements that did not apply to our experiment. If more time was given, perhaps writing a program specifically dedicated to seeking these corresponding wavelengths for us would have been much more efficient than calibrating by hand. The potential for human error is exponentially greater when we are required to be the ones hand-picking the wavelengths. Next, the dome flats and bias frames provided were difficult to work with, and the actual science frames had such faint intensities that subtracting the bias from the science frame often caused a zero error. To improve this, all of the bias frames and dome flats should be averaged together to get an accurate representation of the natural pixel read noise and their variations in intensity, and the bias frames had to be slightly altered to avoid zero errors.

Finally, this experiment could've been significantly more precise by using additional technology, such as an artificial star; the high power, 589nm lasers that are activated to excite sodium atoms in the higher mesosphere. Using this artificial source as a reference point would've allowed us to detect more sources of error such as real time weather conditions, atmospheric distortion, and other optical aberrations. To conclude, future experiments should consider these multiple suggestions for improvement in order to be able to more accurately create spectral plots of celestial objects. Without a higher level of accuracy implemented, one would be unsure of their resulting calculations regarding the chemical composition of the observed objects.

Section 6: Appendix

All of the in house functions and figures were created by myself and my lab group in a collaborative effort. Our lab group consists of myself, Antony Sikorski, and my lab partner Lucas Scheiblich. No particular function was dominated by one of the members of the lab group, because each one was thoroughly verified and improved by the remaining group members. Code was also sampled and used from most of the Discussions and Coding Exercises, which was permitted by Caleb Cohan.

In House Functions (All written in Python):

Function 1: Centroid Finder

This function finds the centroids of a particular spectra.

```
def centroid_finder(x,y,bias): #Take the x and y values of the spectrum.
    s = []
    I = []
    cmean = []
    cstddev = []
    i = 0
    while i < len(x): #Goes through all of the pixels
        if y[i] > bias: #Skip over any pixel lower than the noise
            check = True
        else:
            check = False
        if check:
            s.append(x[i])
            I.append(y[i])
            if y[i-1] >= y[i] and y[i] < y[i + 1] or y[i + 1] <= bias:
                #If new emission line is found, find standard deviation and mean and append it
                m = np.sum((np.multiply(s,I)) / np.sum(I))
                cmean.append(m)
                #standard deviations
                std = np.sum(np.multiply(I,(s-m)**2)) / np.sum(I)
                cstddev.append(std)
                #Clear the arrays
                s = []
                I = []
        i += 1 #if there are more pixels, check again.
    return [cmean,cstddev] #return the arrays of centroids and standard deviations as a new array
```

Function 2: Spectrum Plot Generator

This function plots spectra from a given list of fits files.

```
def plotspectrum(name):
    filename = name
    data = np.genfromtxt(filename, skip_header = 17, skip_footer=1)
    pos = data[:,0]
    intensity = data[:,1]
    %matplotlib inline
    plt.figure(1)
    plt.plot(pos, intensity, 'o-', markersize = 2)
    plt.xlabel("Pixel Position")
    plt.ylabel("Intensity")
    plt.show()
```

Function 3: Linear Fit Function

This function calculates the linear least squares fit in order to determine the best fit model to our observed data set.

```
def linfit(x,y):
    a = np.array([[np.sum(np.power(x,2)),np.sum(x)],[np.sum(x),x.size]])
```

```
    b = np.array([[np.sum(np.multiply(x,y))],[np.sum(y)]])
    c = np.matmul(np.linalg.inv(a),b)
    return c
```

Function 4: Linear Fit Error and Plot Generator

This function calculates the error of our linear fit, and then generates a plot comparing the best fit and the actual data points.

```
def linfit_error(x,y,m,c):
    N = x.size
    sig2 = 1/(N-2)*(np.sum(np.power(y-(m*x+c),2)))
    sign2=N*sig2/(N*np.sum(np.power(x,2))-np.power(np.sum(x),2))
    sigc2 = sig2*np.sum(np.power(x,2))/(N*np.sum(np.power(x,2))-np.power(np.sum(x),2))
    return [np.sqrt(sign2),np.sqrt(sigc2)]

i = 0
while i<=3:
    print(names[i],"Fit")
    fit = linfit(chosenWL[i],chosenCDS[i])
    fite = linfit_error(chosenWL[i],chosenCDS[i],fit[0],fit[1])
    print("Slope, Intercept:", np.transpose(fit))
    print("Error:", fite)
    plt.figure()
    %matplotlib inline
    plt.plot(chosenWL[i], chosenCDS[i], 'o-', markersize = 2)
    x = np.array(range(int(np.min(chosenWL[i])),))
    plt.plot(x,fit[0]*x+fit[1],'k',alpha = 0.5)
    plt.xlabel("Centroid (pixel position)")
    plt.ylabel("Wavelength (nm)")
    plt.show()
    i+=1
```

Function 5: Science Frame Reduce Function

This function reduces and cleans the data by utilizing both the dome flats and bias frames in order to clean our science frame. Note: Bias is divided by 1.0001 to avoid zero errors.

```
def reduce(science,flat,bias):
    bias = np.array(bias)/1.00001
    flat = np.array(flat)
    science = np.array(science)
    flat_norm = (flat-bias/np.median(flat-bias))
    science_reduced = (science-bias)/flat_norm
    return science_reduced
```

Function 6: Linearize Function and Plot

This function reduces the 2d science frames to cut it down into a 1d slice at a certain position so that we can plot intensity vs pixel position for our two science frames.

```
def linearize(science,max1,max2):
    i = 0
    size = science[0].size
    spectrum = []
    m = (max2-max1)/-size
    while i<size:
        spectrum.append(science[int(max1-m*i),i])

        i+=1
    return np.array(spectrum)

%matplotlib inline
s1 = linearize(Galaxy1[h:279,0:w]-(bias[h:279,0:w]/1.000001),177,166)
plt.figure()
x = np.array(range(0,2020))
plt.plot(x, s1, 'o-', markersize = 2)
plt.xlabel("Pixel Position")
plt.ylabel("Intensity")
plt.show()

s2 = linearize(Galaxy2[h:239,0:w]-(bias[h:239,0:w]/1.000001),177,166)
plt.figure()
x = np.array(range(0,2020))
plt.plot(x, s2, 'o-', markersize = 2)
plt.xlabel("Pixel Position")
plt.ylabel("Intensity")
plt.show()
```