DSCP Final Project

Yu-Hsuan Liu 劉宇軒
ID:40875038H

1 Preface

Hi, I am Yu-Hsuan Liu from the department of physics. I have been doing research as an undergraduate student in ASIAA since February 2022. My research specialties are stellar evolution and supernova spectroscopy. This project will show how to use the techniques I have learned from this course to analyze observational data. However, I will skip many details about science because it is not the primary goal of this project.

I will briefly show the necessary materials to study the spectroscopic properties of supernovae. The following .txt file is an example of what a typical supernova spectrum data will look like. There are two columns of data in ASCII format. The first one is wavelength, and the second is measured flux. Also, the unit of the measured flux is confined to 10^{-15} $erg/cm^2 \cdot s \cdot \mathring{A}$.

```
5320.00   0.00170182
5321.00   0.00206928
5322.00   0.00211095
5323.00   0.00167062
5324.00   0.00113505
5325.00  0.000743681
5326.00  0.000680448
5327.00   0.00113356
5328.00   0.00146717
5329.00   0.00141629
5330.00   0.00106693
5331.00  0.000989013
5332.00  0.000607791
5333.00  0.000490684
```

Figure 1: two columns of data in ASCII format (wavelength, flux)

As you can see, there is a space between the two columns. This is the format widely used by many theory spectroscopists. It is also compatible with many codes to analyze such spectroscopic data. If the data is not in this format, my computer can not do further calculations. However, astronomers who spend many hours operating telescopes to observe these spectra tend to save these data in another format as follows:

```
"data":[
        [
                "5320.00",
                "1.70182e-18"
        ],
        [
                "5321.00",
                "2.06928e-18"
        ],
        [
                "5322.00",
                "2.11095e-18"
        ],
        [
                "5323.00",
                "1.67062e-18"
```

Figure 2: A different data format saving wavelengths and flux in one list element

So, I need to make minor adjustments to these data to have better compatibility. I am going to convert all the data that has a format similar to Fig. 2 to a two-column ASCII table (Fig. 1).

2 Example Codes 1: plot.py

The first code code converts all the data in a format with multiple "lists in the list" into a two-column ASCII table. I save the original data in "Data_from_Catalog.py", and the final output of a two-column ASCII table is "ESSENCEm158.txt". The .txt file will be created after running this code. You will also see a plot of a spectrum of the supernova ESSENCEm158.

```python
#load the modules and needed files
import matplotlib.pyplot as plt
import Data_from_Catalog

#Create a new .txt file to write in data (target: 2 columns ASCII )
wrt=open(r'ESSENCEm158.txt','w')

#import the observational data, the format is similar to  Fig.2
data=Data_from_Catalog.f()

#For data visualization
wl=[]
flux=[]

#Read data and rewrite in a new file (list in list --> 2 column ASCII)
#all "append" in this section will be useful for data visualization
for i in data:
    for j in i:
        if float(j)>1000: #basically it's wavelength, put it in the first column
            wl.append(float(j))
            wrt.write(j)
            wrt.write(' ') #space
        else:
            flux.append(float(j))
            F=str(float(j)*1.e15)  #calibration to 1.e-15 for research purpose
            wrt.write(F)
            wrt.write('\n')
wrt.close()

#Data Visualization
plt.plot(wl,flux)
plt.xlim(5500,7000)
plt.title("ESSENCEm158")
plt.xlabel(r"rest wavelength ($\AA$)")
plt.ylabel(r"relative flux")
plt.show()
```

The plot will look like:

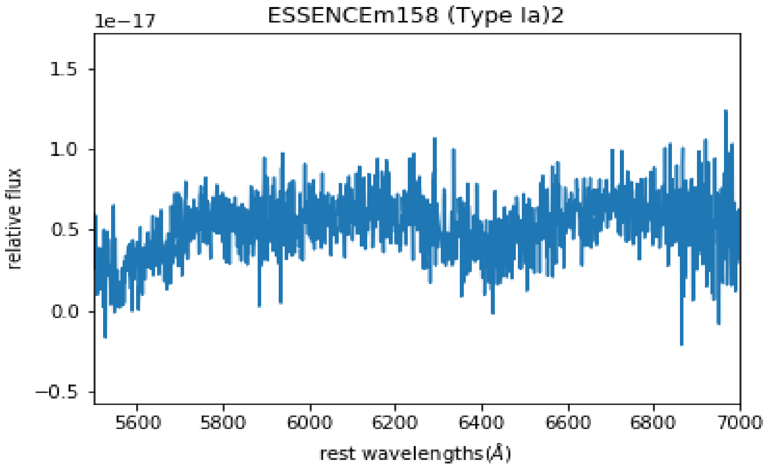

Figure 3: Observational spectrum of supernova ESSENCEm158

3 Example Codes 2: format conversion.py

The following code is a little different. Note that the file "Data_from_Catalog.py" is actually a small part of a .json file in Astro Catalogs which collects most of the observational data of supernovae. I have manually copied everything related to spectra and pasted it into the file. This process is a waste of time, so the following code will be essential if you want to download the original observational data in .json format and do further analysis more straightforwardly.

```
1 import json
2 
3 #open the files
4 with open(r'ESSENCEm158.json') as f:
5     source=json.load(f) #source is a dict
6 
7 #write data in terms of 2-column ASCII (like Fig. 1)
8 wrt=open(r'ESSENCEm158_from_json.txt','w')
9 
10 #required data: only spectra, the other information are trivial in my research
11 spec=source['ESSENCEm158']['spectra'][0]['data']
12 
13 #another way to arange these data
14 for i in spec:
15     wrt.write(i[0])
16     wrt.write(' ')
17     wrt.write(str(float(i[1])*1.e15))
18     wrt.write('\n')
19 wrt.close()
```

With this code, you don't have to copy the raw text from the database to make "Data_from_ Catalog.py" by yourself. The only thing I do here is to understand how .json works, then pick out the information I need, just like line 12 in the code above. After running the two codes, you can check whether the results match. The two .txt files should look the same.

4 Appendix

Data source: Astro Catalogs

I have combined all the stuff above into a single .ipynb file. The file and all components can be downloaded on my Github Page.