Categorizing the Hipparcos and Gliese Stars

Phys 4200 Elvin Rivera

May 12, 2019

1 Introduction:

The entire prospect of this project is to data mine both the Hipparcos[1] satellite and the Gliese[2] catalog. The data will be used to create a Hertzsprung-Russell diagram (H-R diagram). I will be using a python distribution called Anaconda[3] to plot all the data.

2 Magnitude Vs. Color:

In order to get an apparent quantitative meaning of our light, we impose two filters. In this case filter b and filter v. These are the most commonly used filters. The B-filter will show brighter stars, and the V-filter is within visible range for humans. For these filters to be useful we subtract the B filter by the V filter. This is typically referred to a color filter (b-v). The purpose of this is to get an apparent measurement of how much flux is being transmitted between two filters. Taking the difference between the two we can find the Absolute Magnitude with this equation.

$$Magnitude_V = v + 5 * (\log_{10} \frac{d}{100}) \tag{1}$$

Where the variable v is the measurements of the v-band filter taken directly from the databases, and 'd' is the distance in parallax measured in arcseconds. The distance is also taken directly from both the Hipparcos satellite database and the Gliese catalog. As far as equations go, that is all we need for a basic H-R diagram!

3 Coding in Python:

The quantitative side of this project was by far significantly easier than the programming aspect. All the data was imported with Pandas[?]. I would highly recommend that module. It is very easy and pleasant to work with. The first thing that I did was change all the empty spaces into NAN values.

```
data = data.applymap(lambda x: np.nan if isinstance(x,str) and\
    x.isspace() else x)
```

This little piece of code searches through your data and replaces every empty space for a NAN value. This is done because a NAN value is much easier to remove than an empty spot in the data.

```
data = data.dropna()
```

Thankfully, the module Panda makes most of this particularly easy. The 'dropna()' removes all NAN values present in that data-frame. The trickiest part is filtering all the types of stars you have into their respective spectral categories. This line of code dose exactly that. It searches through your data-frame and keeps every element that holds the following spectral classification.

```
f = lambda s: s[0] in 'OBAFGKM'
data = data[data['stype'].map(f)]
```

The code above only ensure that the data we keep is of the proper stars we want; however, it doesn't categorize them by their respected groups; main sequence, dwarf stars, giants, etc..

To plot a H-R diagram, filtering the data even further into their proper classifications isn't necessary. You can strictly just plot b-v against the absolute magnitude and up with something like this

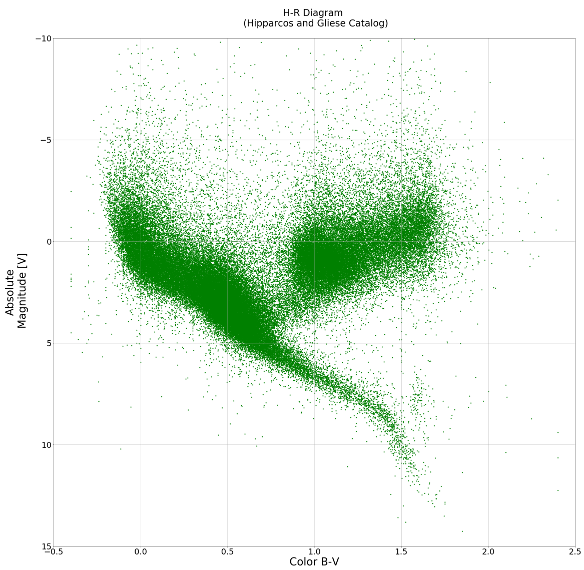


Figure 1: H-R Diagram with only color filters and absolute magnitude

; however, if you want to do any useful physics, it becomes easier to plot best fitted lines when they are separated into proper groups. This line of code does precisely that.

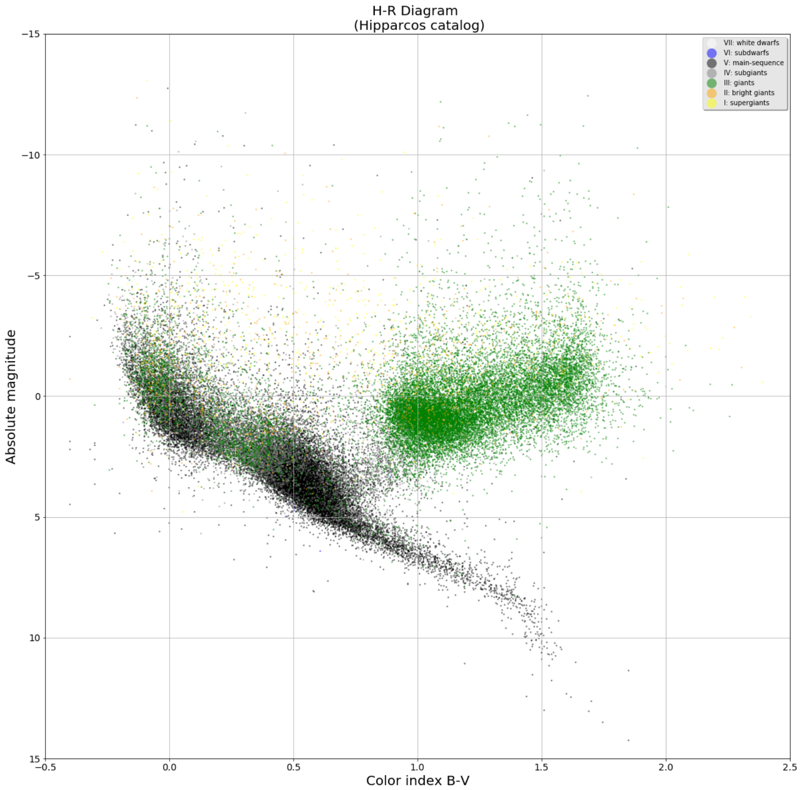
```
f = lambda s: 'VII' in s
b = data['stype'].map(f)
print ("Class VII: white dwarfs, there are %d stars"%sum(b))
```

The code above searches through the data to satisfy the Boolean statement. In this case it is being asked to search through the data-frame, and keep any data that has the string 'VII'. This particular example is for white dwarf stars. You'd repeat this as necessary depending on your data

4 Conclusion:

My initial idea was to take the previous work I had done on the Coma Galaxy Cluster and extend on that with multiple galaxy clusters. I quickly found out from The Sloan Digital Sky Survey[5] that it becomes extremely difficult to find reliable data the further you go into the more exotic galaxies. Because of this limitation isn't strictly only on galaxies, finding a fully-flared out set of data that shows the wide range of stars for a H-R diagram was also a difficult task. The Hipparcos and Gliese catalogs seem to be very common when to comes to making H-R Diagrams.

Even though this is not a new idea, there will always innovative ways to create the same thing. This is my ending result for categorizing over 50,000 stars



References

[1] `http://sci.esa.int/hipparcos/`

[2] `https://heasarc.gsfc.nasa.gov/W3Browse/star-catalog`

[3] `https://www.anaconda.com/distribution/`

[4] `https://pandas.pydata.org/`

[5] `https://www.sdss.org/`