

# cluster\_plot\_teacher-answer\_key

October 4, 2023

## 0.0.1 Answer Key

All my answers in markdown cells will be in red.

## 1 Main sequence fitting of clusters (40 points)

I expect that you will be able to complete most of the work in class. If you do not, you have till **weds. Sept. 20** to hand it in. Instructions on how to do so are at the end of the document.

This exercise is intended to give you a practical exercise in determining the distance to star clusters in the milky way. You will do this by plotting the color-magnitude diagrams of 4 clusters compared to that of a cluster with a known distances. This “reference” cluster is the nearby Pleiades (M 45), which is visible to the naked eye in late fall/winter. It has a distance of 130pc.

There are four other clusters that you will be studying: M 34 M 36 NGC 6231 NGC 3293

I have written a program to plot the color magnitude diagram of each cluster on top of the Pleiades cluster. You run this code by typing:

```
cp.cmdplot(clustname='cluster name',magshift=0.0,colshift=0.0)
```

For each cluster you will use the code to plot that cluster and input color and magnitude shifts to align the cluster with the Pleiades. Using your knowledge of how distance affects color and magnitude, you will need to decide if you use color shifts, magnitude shifts, or both.

Using these shifts you will write a snippet of code below each plot in the space provided. This code should be a short calculation to determine what the distance is based on the shifts.

*Note that it is not required that color and magnitude shifts be used in your calculation. You get to decide which are important.*

In order to get full credit for this assignment all cells have to execute without errors. You also need to fill out the cells at the bottom and add code where required

The colors and magnitudes being used in this program are called GRP and GBP. These stand for “GAIA Red Passband” and “GAIA Blue Passband” and are custom filters used for the GAIA mission.

All sections of code that you need to work on will be indicated by a ‘#### Write code here’

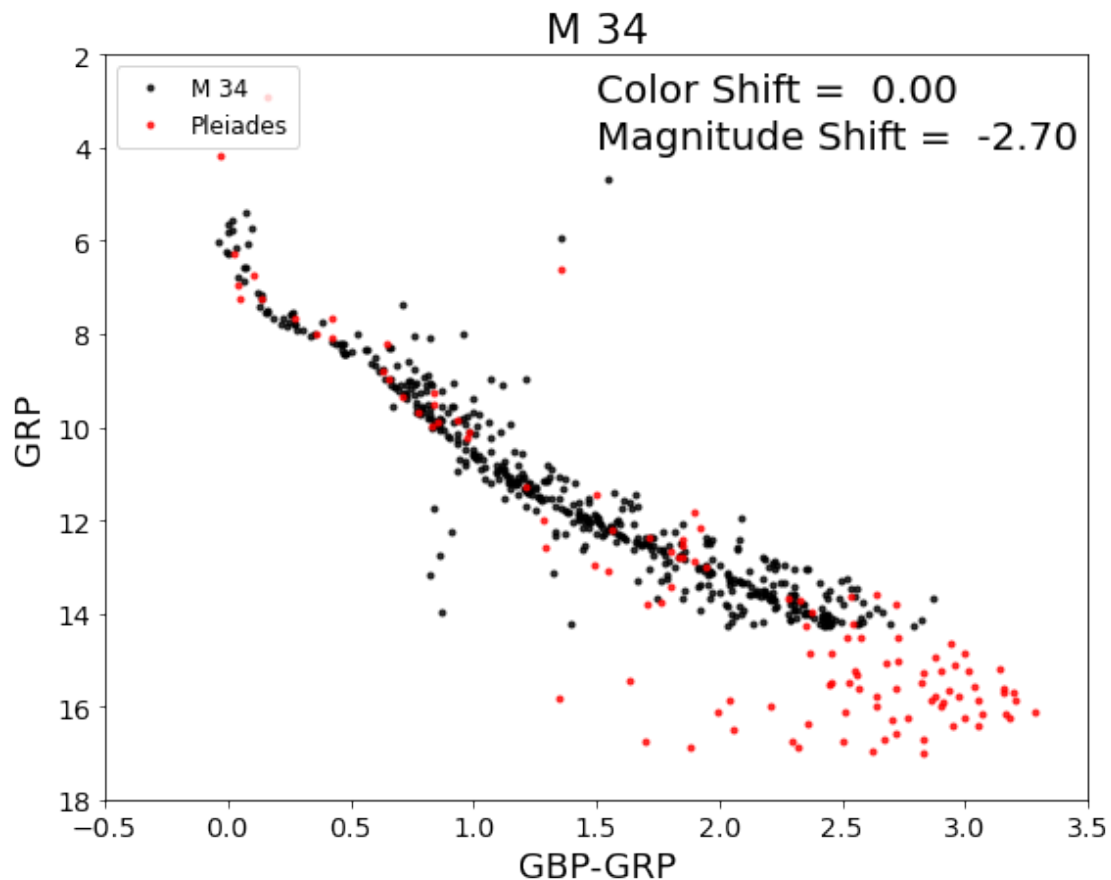
```
[1]: import cluster_plot as cp
import numpy as np
from matplotlib import pyplot as plt
```

```
import warnings
warnings.filterwarnings('ignore')
```

1.1 From here onward is the assignment

M34 (3 points)

```
[2]: cp.cmdplot(clustname='M 34',magshift=-2.7,colshift=0.0)
```



In this cell, describe what you chose for your shifts and why.

I only shifted in magnitude because distance only affects the magnitude and not the color if we ignore the effects of dust reddening. I tried to line up the main sequences between the two clusters.

Use the derived shifts to compute a distance to the cluster relative to the Pleiades cluster by writing a small snippet of code.

```
[3]: #write a calculation here that you execute and which prints out the ratio of
      ↳ the distance of your cluster to that of the Pleiades
      #To compute the log_10 you need to write "np.log10(number)"
      #### Write code here.
```

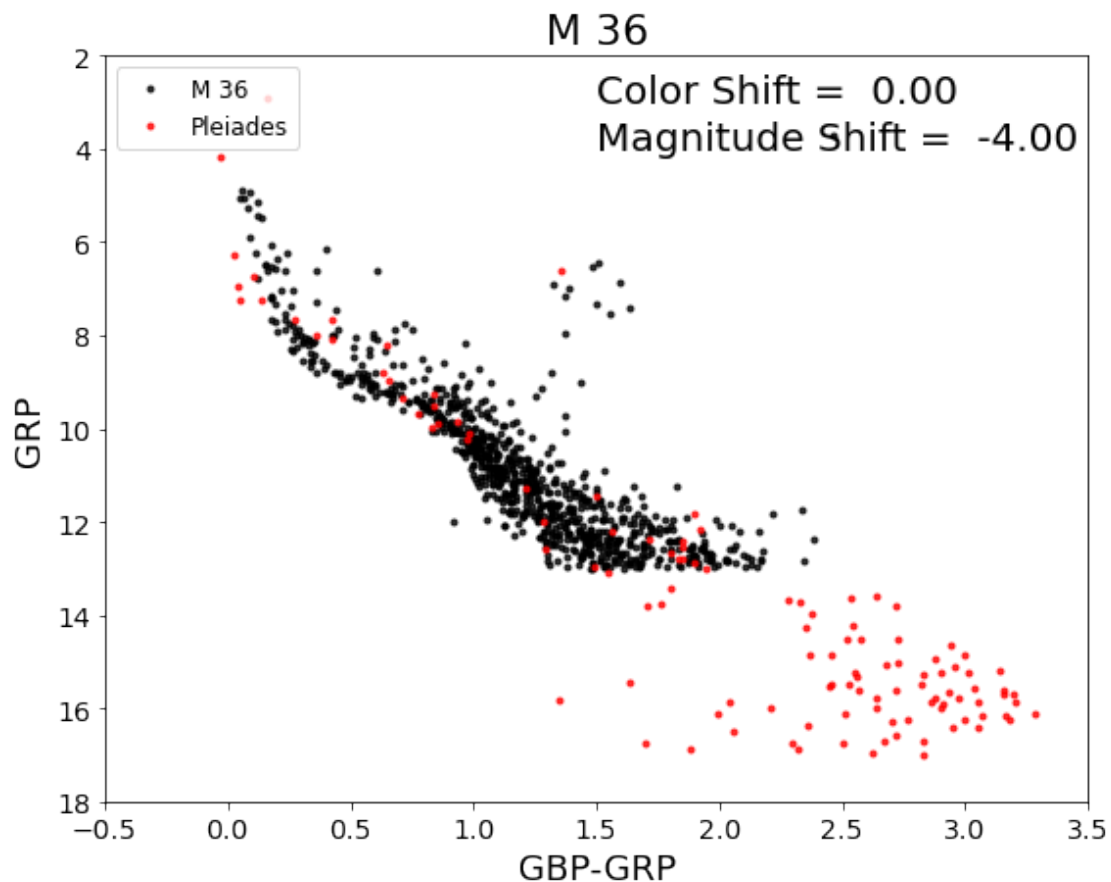
```
# enter your calculated shift here
shift_m34 = -2.7

# Assuming the distance to M45 is 130pc write an code expression that converts
→ your shift into the ratio of the distances
#the Pleiades to this cluster
dist_ratio_m34 = 10**((shift_m34)/5)
print(dist_ratio_m34)
```

0.28840315031266056

### M 36 (3 points)

```
[4]: cp.cmdplot(clustname='M 36',magshift=-4,colshift=0.0)
```



In this cell, describe what you chose for your shifts and why.

I only shifted in magnitude because distance only affects the magnitude and not the color if we ignore the effects of dust reddening. I tried to line up the main sequences between the two clusters.

Use the derived shifts to compute a distance to the cluster relative to the Pleiades cluster by writing

a small snippet of code.

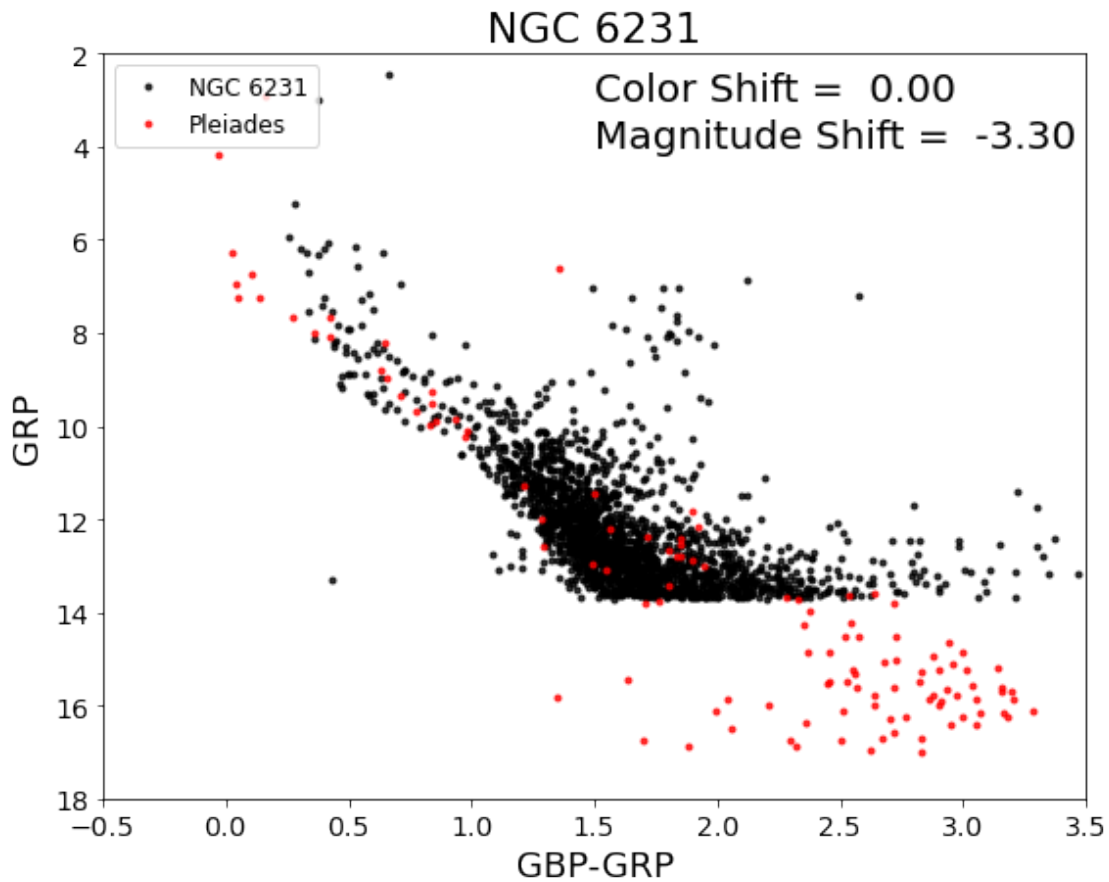
```
[5]: #write a calculation here that you execute and which prints out the ratio of
      ↳ the distance of your cluster to that of the Pleiades
      #To compute the log_10 you need to write "np.log10(number)"
      ##### Write code here.
      # enter your calculated shift here
      shift_m36 = -4.0

      # Assuming the distance to M45 is 130pc write an code expression that converts
      ↳ your shift into the ratio of the distances
      #the Pleiades to this cluster
      dist_ratio_m36 = 10**((shift_m36)/5)
      print(dist_ratio_m36)
```

0.15848931924611134

### NGC 6231 (3 points)

```
[6]: cp.cmdplot(clustname='NGC 6231',magshift=-3.3,colshift=0.0)
```



In this cell, describe what you chose for your shifts and why.

I only shifted in magnitude because distance only affects the magnitude and not the color if we ignore the effects of dust reddening. I tried to line up the main sequences between the two clusters.

Use the derived shifts to compute a distance to the cluster relative to the Pleiades cluster by writing a small snippet of code.

```
[7]: #write a calculation here that you execute and which prints out the ratio of
      ↳ the distance of your cluster to that of the Pleiades
      #To compute the log_10 you need to write "np.log10(number)"
      ##### Write code here.

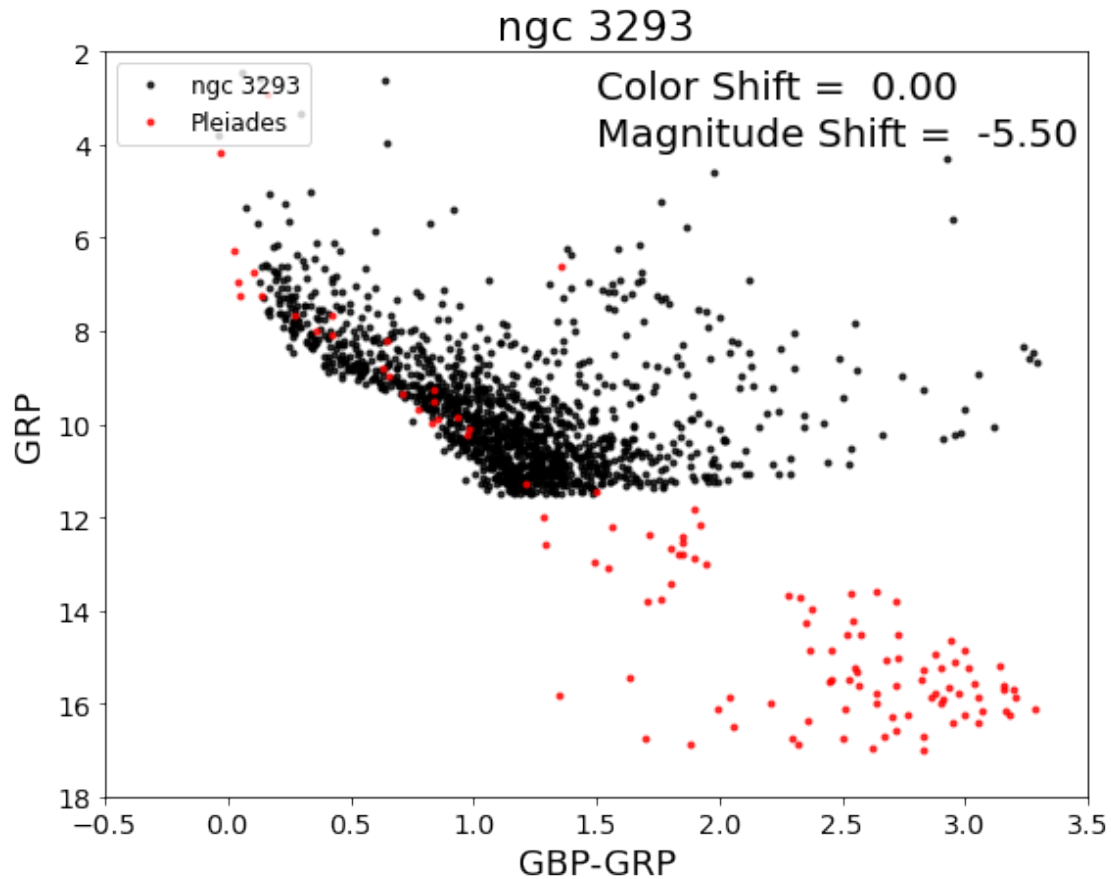
      # enter your calculated shift here
      shift_ngc6231 = -3.3

      # Assuming the distance to M45 is 130pc write an code expression that converts
      ↳ your shift into the ratio of the distances
      #the Pleiades to this cluster
      dist_ratio_ngc6231 = 10**((shift_ngc6231)/5)
      print(dist_ratio_ngc6231)
```

0.2187761623949553

**NGC 3293 (3 points)**

```
[8]: cp.cmdplot(clustname='ngc 3293',magshift=-5.5,colshift=0.0)
```



In this cell, describe what you chose for your shifts and why.

I only shifted in magnitude because distance only affects the magnitude and not the color if we ignore the effects of dust reddening. I tried to line up the main sequences between the two clusters.

Use the derived shifts to compute a distance to the cluster relative to the Pleiades cluster by writing a small snippet of code.

```
[9]: #write a calculation here that you execute and which prints out the ratio of
    ↳ the distance of your cluster to that of the Pleiades
    #To compute the log_10 you need to write "np.log10(number)"
    ##### Write code here.

    # enter your calculated shift here
    shift_ngc3293 = -5.5

    # Assuming the distance to M45 is 130pc write an code expression that converts
    ↳ your shift into the ratio of the distances
    #the Pleiades to this cluster
    dist_ratio_ngc3293 = 10**((shift_ngc3293)/5)
```

```
print(dist_ratio_ngc3293)
```

0.07943282347242814

## 2 plotting distances vs. distance modulus (12 points)

If the distance to the Pleiades cluster is 130pc, compute the distance modulus to each cluster. You can do the algebra on a separate piece of paper, but below I want you to code up the formula for each cluster in the field below

```
[10]: #For each cluster put in a line of code that calculates the distance modulus. ␣  
      →I have given part of the code below and expect you to fill it in.  
  
      #this creates a python array that contains all of the distance ratios you␣  
      →calculated above.  
      #Now you can do operations on this array and it will operate on every element  
dist_ratios = np.array([dist_ratio_m34, dist_ratio_m36, dist_ratio_ngc6231,␣  
      →dist_ratio_ngc3293])  
  
      #### Write code here.  
  
      #write an expression to calculate the distance to each cluster  
dist_pleiades = 130.0  
dists = dist_pleiades / dist_ratios  
  
print("Distance for M 34 = ", dists[0], " pc")  
print("Distance for M 36 = ", dists[1], " pc")  
print("Distance for NGC 6231 = ", dists[2], " pc")  
print("Distance for NGC 3293 = ", dists[3], " pc")  
print()  
  
      #write an expression to derive the distance modulus from the distance ratio on␣  
      →distance  
dist_mods = 5 * np.log10(dists / 10.0)  
  
print("Distance Modulus for M 34 = ", dist_mods[0])  
print("Distance Modulus for M 36 = ", dist_mods[1])  
print("Distance Modulus for NGC 6231 = ", dist_mods[2])  
print("Distance Modulus for NGC 3293 = ", dist_mods[3])
```

```
Distance for M 34 = 450.7579055882912 pc  
Distance for M 36 = 820.2445478242513 pc  
Distance for NGC 6231 = 594.2146464993375 pc  
Distance for NGC 3293 = 1636.6030353324177 pc
```

```
Distance Modulus for M 34 = 8.269716761534184  
Distance Modulus for M 36 = 9.569716761534185
```

Distance Modulus for NGC 6231 = 8.869716761534184  
Distance Modulus for NGC 3293 = 11.069716761534185

Now make a plot of the Distance modulus you calculated above vs. the log10 of the distance. I will write some of the code below and you will need to fill it in to get the appropriate values.

I then want you to overplot the relation that you would expect given the relation between distance and distance modulus

In the `plot` statement below there are missing arguments. the `'ko', markersize=9)` part of the call indicates that you should have black circles (`'ko'`) and that the size of the points should be 9, in whatever units pyplot uses (bigger number = larger points)

```
[11]: #You will need to make an array that contains the log of the distances

#### Write code here
logdist = np.log10(dists)

#now complete this plot statement to plot the log of the distance vs. the
↪distance modulus.
plt.plot(logdist, dist_mods, 'ko', markersize=9)
plt.xlabel('log(distance [pc])')
plt.ylabel('Distance modulus')

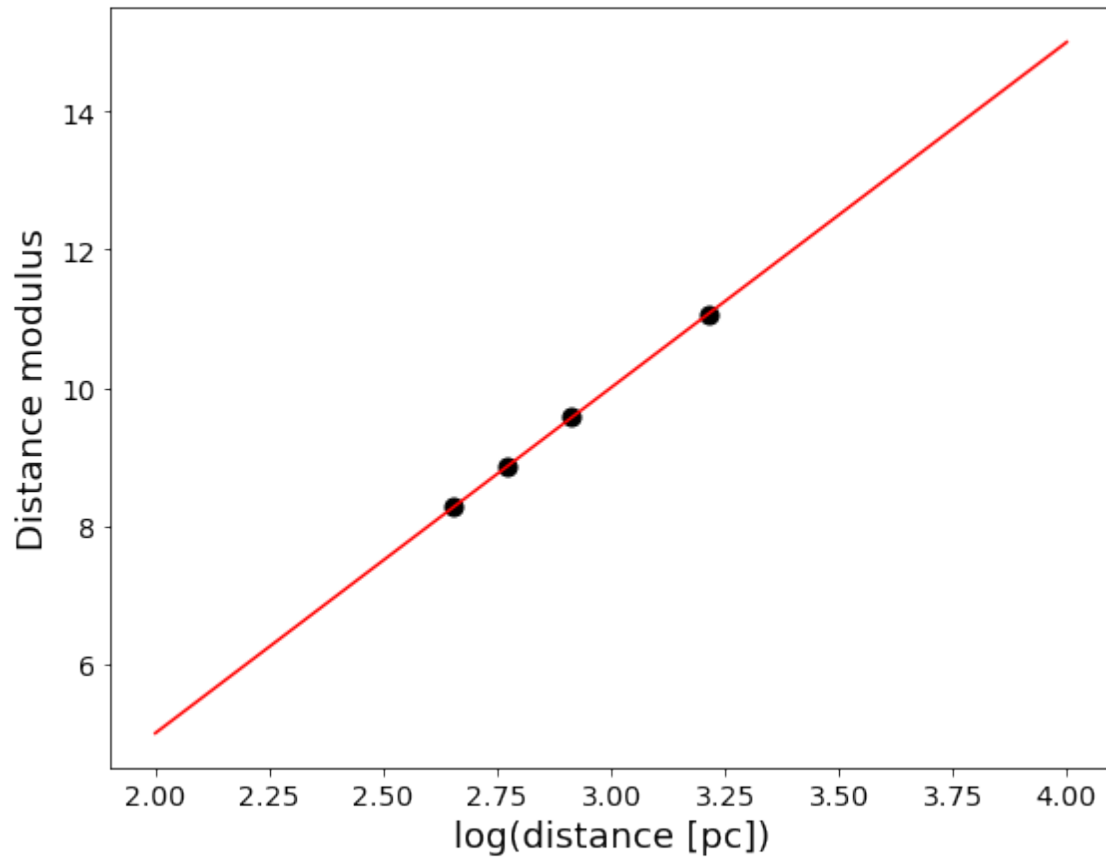
# here you will need to draw a line on your plot that shows the prediction
#of the distance modulus given the distance.
# these are the two x-points of your line, in log distance
logdistpred = np.array([2,4])

# write down the function that turns logdistpred into the distance modulus
#
dmpred = 5*(logdistpred - np.log10(10.0))

#plot this line on the previous plot
#the 'r-' indicates a red line
plt.plot(logdistpred, dmpred, 'r-')
```

```
[11]: [
```





### 2.0.1 below this line you need to modify the cells and include text (16 points)

In each cell add text to answer the following questions

- How the line you plotted matches (or not) the trend you see in the data.

The line perfectly matches the data.

- Why do more distant clusters have more stars? Note that I used parallaxes to make a cut of roughly 30% in distance around each cluster. This means I took the mean distance of the cluster and took stars between 70% and 130% of that distance.

If there is a 30% cut in distance around each clusters, this corresponds to a larger actual distance cut around each cluster. Since the volume of the Milky way around each cluster is larger for the more distant clusters, you should see more random field stars.

- What astrophysical effect that we are ignoring might affect our derived distances and in what way?

Dust extinction and reddening.

- How might you do a better job determining the shifts than doing it by eye. If you would like to try it out your method, you can take some of the data files in the Project\_clusters/

directory and try to implement a better method. The quantities we use to make the plots are `bp_rp` for the x-axis and `phot_rp_mean_mag` for the y-axis.

**Hint:** There are functions in numpy, `numpy.median()` and `numpy.average()` that take the mean and median of an array passed to it. So you could use Boolean commands to select a portion of one of the axes and one of the numpy commands to compute the median or average value of the other parameter.

The quantities we use to make the plots are `bp_rp` for the x-axis and `phot_rp_mean_mag` for the y-axis

For each cluster you could take stars in a given color range and compute their median magnitude. Then you could use the difference in the median magnitude of those stars to figure out the shift.

## 2.1 Instructions for handing the assignment in

Make sure to save your notebook.

Add this file to your `CMD_fitting_distance_modulus` github repository.

No need to add me as a collaborator. As long as you named the exercise accordingly I will be able to find it in your repository.