Praktische Sterrenkunde Report for Assignment 4: Distance and age of the Hyades using GAIA DR2 Data

Damian Opmeer, s4040481

Version 1, May 3, 2024

Abstract

The Hyades is the closest open cluster to earth and has already extensively been researched. Using the GAIA DR2 data, we have made different selections based on positions, parallax, and proper motions to extract the Hyades. The selection of position is a circle with a radius of 8 degrees, the proper motions a square of 35 mas/year in both directions, and the parallax has a range of (17, 27) mas determined by its tidal radius. The best selection is a combination of all methods. Constructing the CMD using the same GAIA data results in a distance modulus of 3.3 ± 0.3. We obtained the distance using the mean parallax, 47.2 ± 0.8 parsecs , and using the distance modulus, 46 ± 6 parsecs. Furthermore, we used the same isochrone data as in the official GAIA DR2 paper and estimated the age of the Hyades to be 600 ± 50 million years. The quantitative results match well with the literature.

1 Introduction

The Hyades, also formally known as Cl Melotte 25, is the closest open star cluster to Earth and lies in the constellation Taurus. An open cluster is a loose collection of stars bounded by mutual gravitational attraction, in contrary to the more crowded globular clusters. Open star clusters often have less than a few thousand stars and are on average younger than globular clusters. The Hyades is one of the best studied open star cluster. Using the J2000.0 epoch system, the right ascension is 4h 27m and declination is +15° 52'. Previous work has shown that the age of the Hyades is 625 million years and its distance is 47 parsecs (Perryman et al. (1997)). The metallicity is measured at +0.14 (Perryman et al. (1997)), which will be important when we use isochrones to determine its age. The brightest stars in the Hyades are the dubblestar system θ^1 Tauri and θ^2 Tauri, and HD28527

This research is important for the the scientific world, because it verifies previous research using professionally gathered data from the GAIA DR2. Doing Astrometry, which is the accurate determination of the positions of stars in the sky, is an incredibly necessary method to understand the kinematics of star systems. By determining these positions and distances from 'close' objects, we get vital information on the cosmic distance ladder, for example. This helps to determine the distances of very far objects, often in the mega parsec range. Lastly, the age of the Hyades is necessary for various reasons, including the evolution of stars and the forming of planetary systems inside these open clusters. The average distance of the Hyades can be determined by using the parallax of the stars. This has a limitation based on the distance, because very far away stars have very small parallax and are thus much harder to accurately measure. Luckily, in our case the Hyades are not that far away and thus parallax can be used. To determine the age of the Hyades, Isochrones are often used. These are theoretical models that show how stars evolve and move in the CMD. They can be used in our case, because all the stars in an open cluster have roughly the same age.

This paper is organized as follows. Section 2 discussion the origin of our data, as well as basic astronomy formulas for various uses. The selection of the data is discussed in section 3, where we try to find the most bona-fide members of the Hyades using position, parallax ,and proper motion. Also, we construct the CMD in this section and fit the isochrones. Section 4 and 5 deal with the discussion and conclusion for this paper, where we validate our results and compare them against previous literature.

2 Methods

The data has been obtained by the GAIA DR2 release, which happened on the 25th of April 2018. This mission lasted 22 months, started in 25 July 2014 and ended in 23 may 2016 Gaia et al. (2018), and has cataloged high-precision astrometry and three-band photometry for a staggering 1.3 billion astronomical objects Koposov et al. (2018). This data contains a lot of quantities, but the most important for this research is position and proper motion in RA and Dec, parallax, G mean magnitude and BP-RP colour. The data in the FITS file has been obtained using the following Queries to the Gaia database:

FROM `gaiadr2.gaia_source`

WHERE `parallax_over_error` > 10.0

AND `phot_g_mean_flux_over_error` > 10.0

AND `phot_bp_mean_flux_over_error` > 10.0

AND `phot_rp_mean_flux_over_error` > 10.0

AND $1.0 + 0.015 \times (\texttt{phot\_bp\_mean\_mag} - \texttt{phot\_rp\_mean\_mag})^2 < \texttt{phot\_bp\_rp\_excess\_factor}$

AND $\texttt{phot\_bp\_rp\_excess\_factor} < 1.3 + 0.06 \times (\texttt{phot\_bp\_mean\_mag} - \texttt{phot\_rp\_mean\_mag})^2$

AND $\sqrt{\frac{\texttt{astrometric\_chi2\_al}}{\texttt{astrometric\_n\_good\_obs\_al}-5}} < 1.2 \times \mathrm{GREATEST}(1.0, \exp(-0.2 \times (\texttt{phot\_g\_mean\_mag} - 19.5)))$

AND `parallax` $> 10.$

This data filtering makes sure that only stars enter that have:

1. Significant parallaxes
2. Significant magnitudes in all 3 colours

3. Have reliable astrometry

4. And can be expected to lie within 100 pc.

The Hyades, in particular, were chosen for this research because it is one of the best studied open star cluster. This means that there is a lot of previous research to rely on, and that is easier to compare our results with the extensive previous literature. Due to the fact that all the previous independent literature agree on the distance of the Hyades, it is an important step on the cosmic distance ladder. This extra-galactic distance scale is vital to determine the distances to further celestial objects.

The formulas used in this paper can be seen as relatively standard formulas. Firstly, we get the distance by measuring the parsec and applying formula 1. Secondly, we use the standard definition for the distance modulus, equation 3. And Lastly, we can also get the distance using the distance modulus, equation 4. All σ mean error and have been obtained using standard error analysis. The subscript is the quantity of which it is the error :

$$d = \frac{1}{\pi} \tag{1}$$

$$\sigma_d = \frac{\sigma_\pi}{\pi^2} \tag{2}$$

d : distance from earth to the object.
π : parallax of the object as seen from the earth at opposite places in solar orbit.

$$\mu = m - M = 5\log(d) - 5 \tag{3}$$

$$\Updownarrow$$

$$d = 10^{\frac{\mu}{5}+1} \tag{4}$$

$$\sigma_d = 0.2 * \ln 10 * 10^{\frac{\mu}{5}+1} * \sigma_\mu = 0.461d * \sigma_\mu \tag{5}$$

μ : distance modulus
m : apparent magnitude
M : absolute magnitude

3 Results

3.1 Selection

As said earlier, our raw data at this point is the filtered data of the GAIA DR2 mission. This data is shown in 1. As you can see, we can see the whole sky. This is very messy and crowded, because it contains 242582 stars. So, the first step to selecting only the Hyades is by selecting a region in space close to stars we want. We do this by finding where the Hyades are located and adding a certain **delta** value in all directions. The delta values range from 1 to 20 degrees. The Hyades are located at an RA of 67.447 and Dec of 16.948 degrees. in figure 2, there are three plots showing delta values of 1,8,19 respectively and the impact it has on the (a) map of positions (RA,Dec), (b) histogram of parallax values, (c) proper motion in RA and Dec, (d) CMD of G versus $Gbp - Grp$.

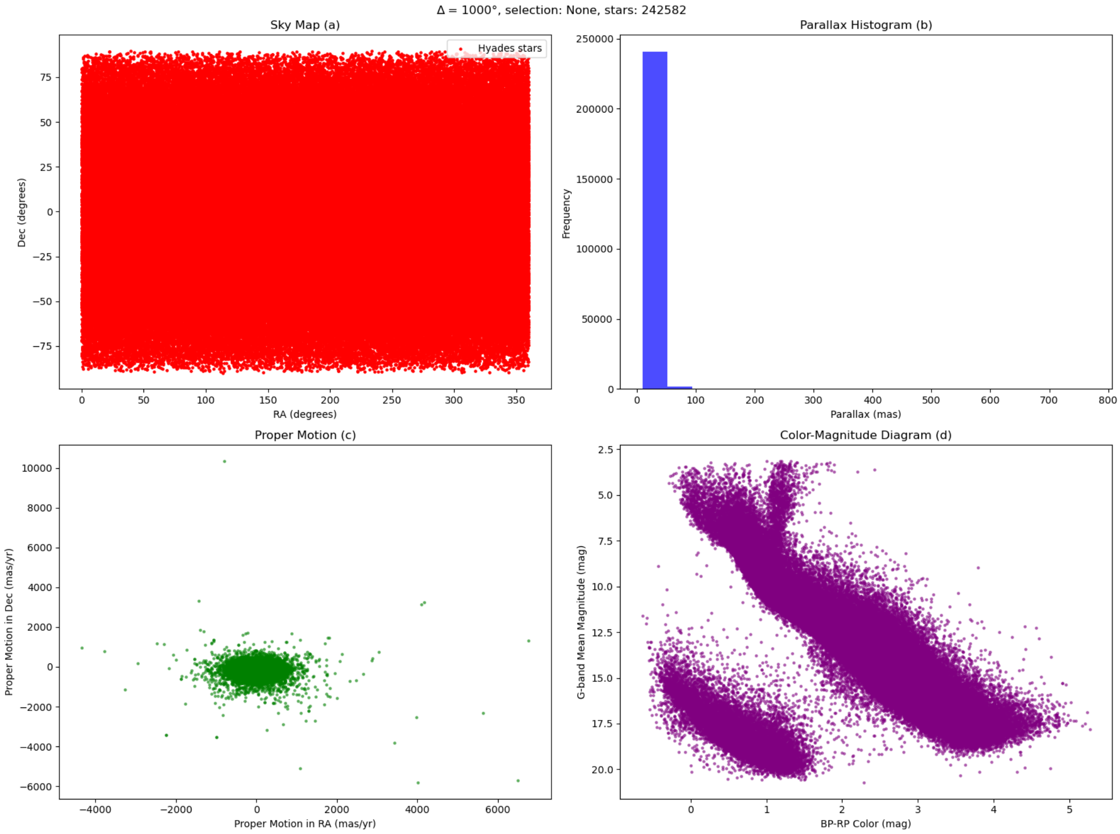
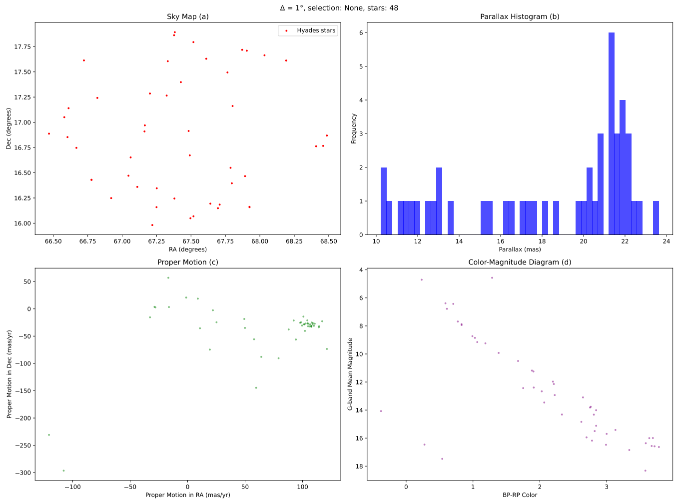
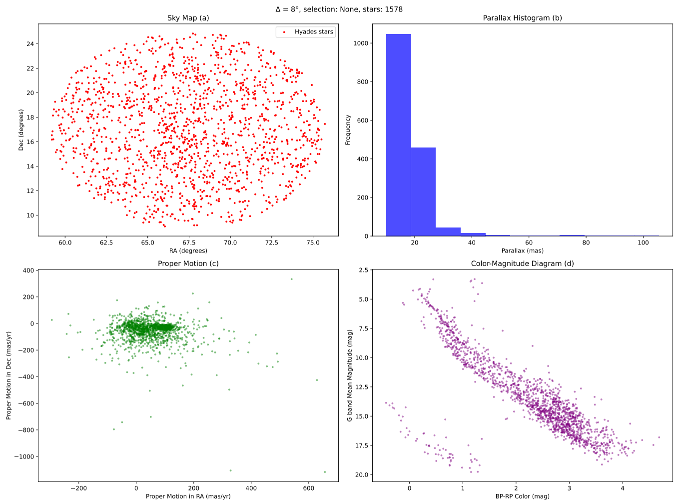


Figure 1: 4 plots of (a) map of positions (RA,Dec), (b) histogram of parallax values, (c) proper motion in RA and Dec, (d) CMD of G versus $Gbp - Grp$. This contains all the stars in our data.

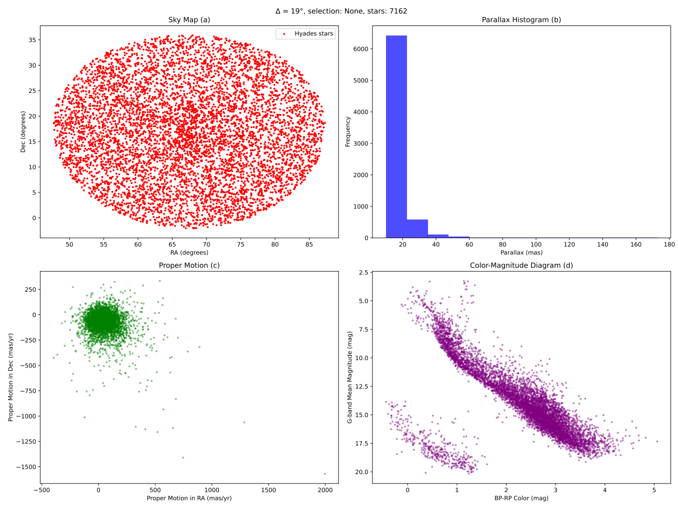
Now, using our eyes and a clever thinking, we can eliminate a lot of options. all delta values under 6 have way too little stars that can be part of the Hyades. All delta values above 10 have the opposite problem: These show way too much stars and thus, there will be a higher chance of picking stars that are definitely not in the Hyades. After careful analysis of all plots, The best delta value seems to be around 7 or 8. As you can see in figure 2, We have many different stars with different parallaxes and different proper motions. So, now it is a good idea to select based on the delta value **and** something else.



(a)



(b)



(c)

Figure 2: A selection based on different delta values centered at the coordinates of the Hyades. A small delta yields very little stars, and a very big delta yields way too much stars.

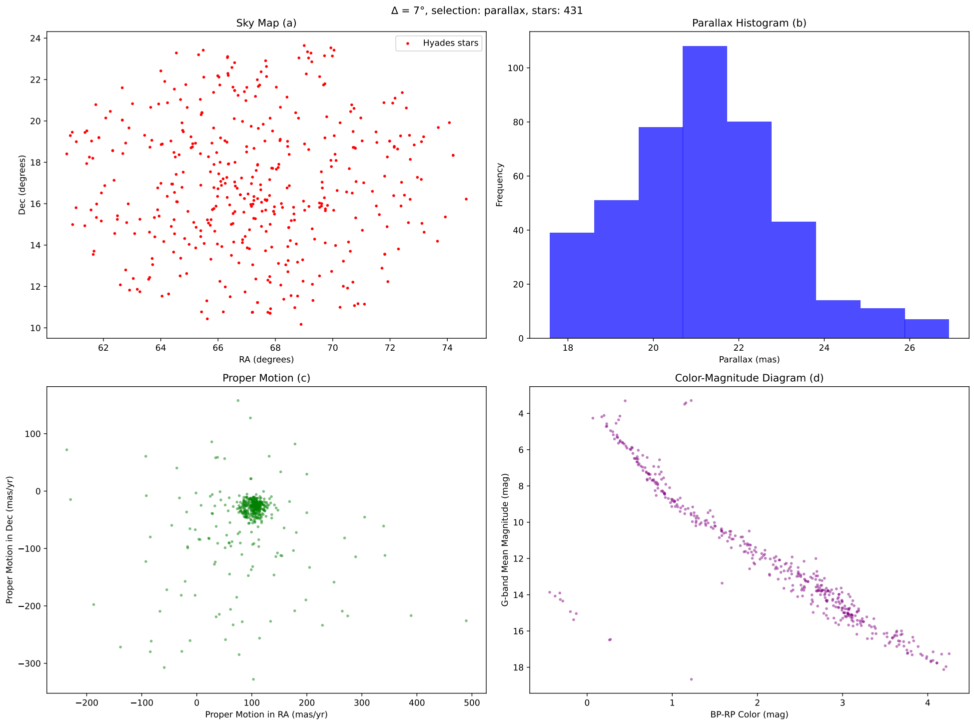
Parallax

The first selection we can make is using the parallax. This is a very natural quantity to select with, as the Hyades all lie relatively close to each other and thus have similar distances and parallaxes. The average parallax of the Hyades is 21.052 milli-arcseconds (mas). But for individual stars in the population, this varies. We will now employ a similar system as our delta values and choose some range in which the Hyades can lie. For this, we can use the fact that the radius of the core is 2.7 parsecs and that the tidal radius us 10 parsecs. The maximum distance a star in the Hyades can be is $47+10 = 57$ parsecs and the minimum is $47-10 = 37$ parsecs. If we convert these distances to parallax using our formula we get that the parallaxes must be between 17 and 27. Since the Hyades is an open cluster, this means the stars can very easily escape the mutual gravitational attraction and can thus be much further away from this tidal radius. In fact, around a third of stars are well outside this boundary, which means we expect around 200-300 stars in our selection. Figure 3 shows this selection on parallax

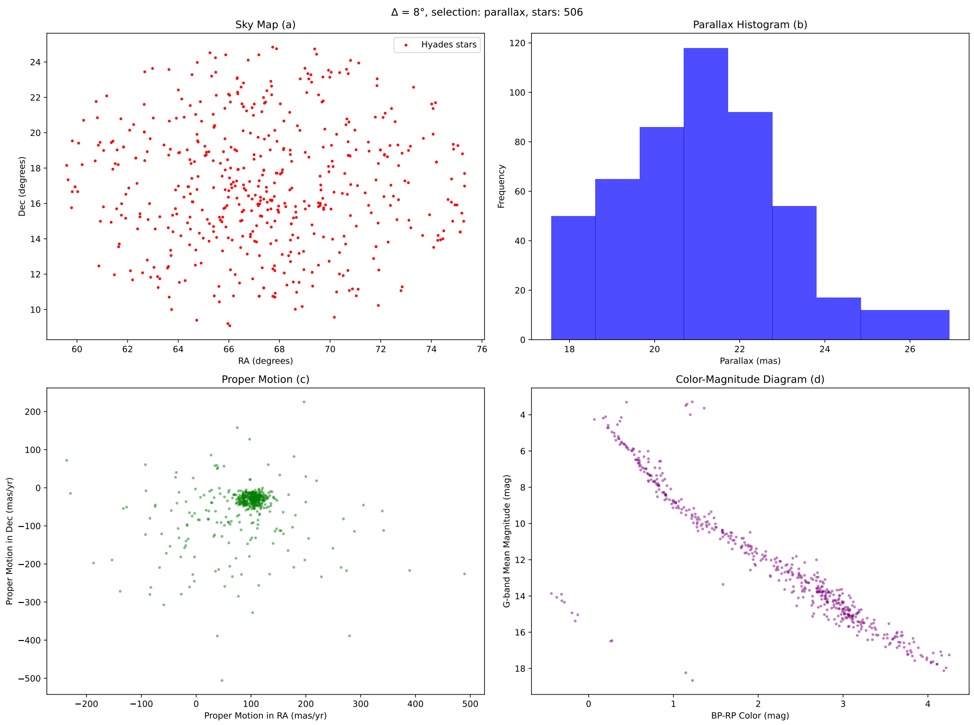
Proper Motion

There is also another way of selection the stars, namely the proper motion of the stars. The average proper motion is 104.92 in RA and -28.00 in Dec. We can again employ a similar method as the previous selections and fit a box around the average. Using our eyes and estimating, we get that the RA fall between () and the Dec fall between (). The selection we make now includes the original delta values and the proper motion. We get the following result in figure 4

The difference between these filtering methods is interesting to note. The proper motions filters less thorough, meaning the selection has more stars. In the histogram, You can see that there are still relatively many stars that have a parallax less than 15. In the parallax selection, you can see most stars' proper motion lie concentrated next to each other. Both selections can be viable, and in our case, it is a great idea to combine both selection method to have more certainty that we select only the Hyades. After all, it is better to have less data, but have the absolute certainty that it is the correct data. The last selection is therefore seen in 5. From this last selection we must also choose an appropriate delta value. This seems to be a delta value of 8 degrees.

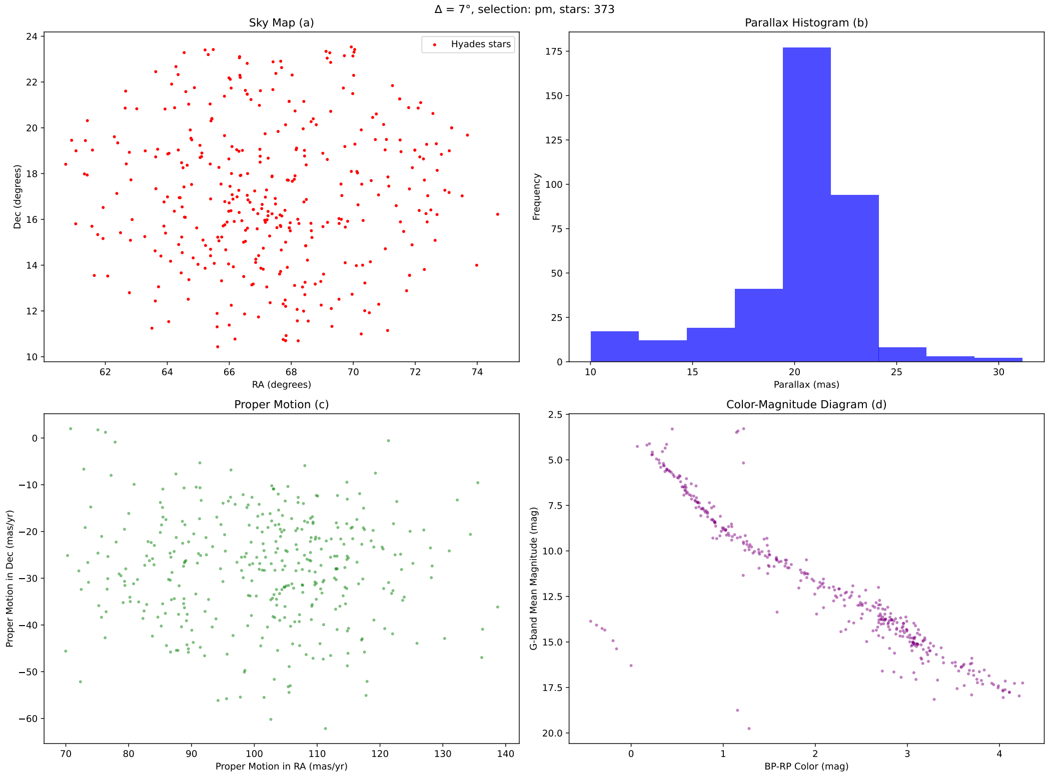


(a) delta = 7

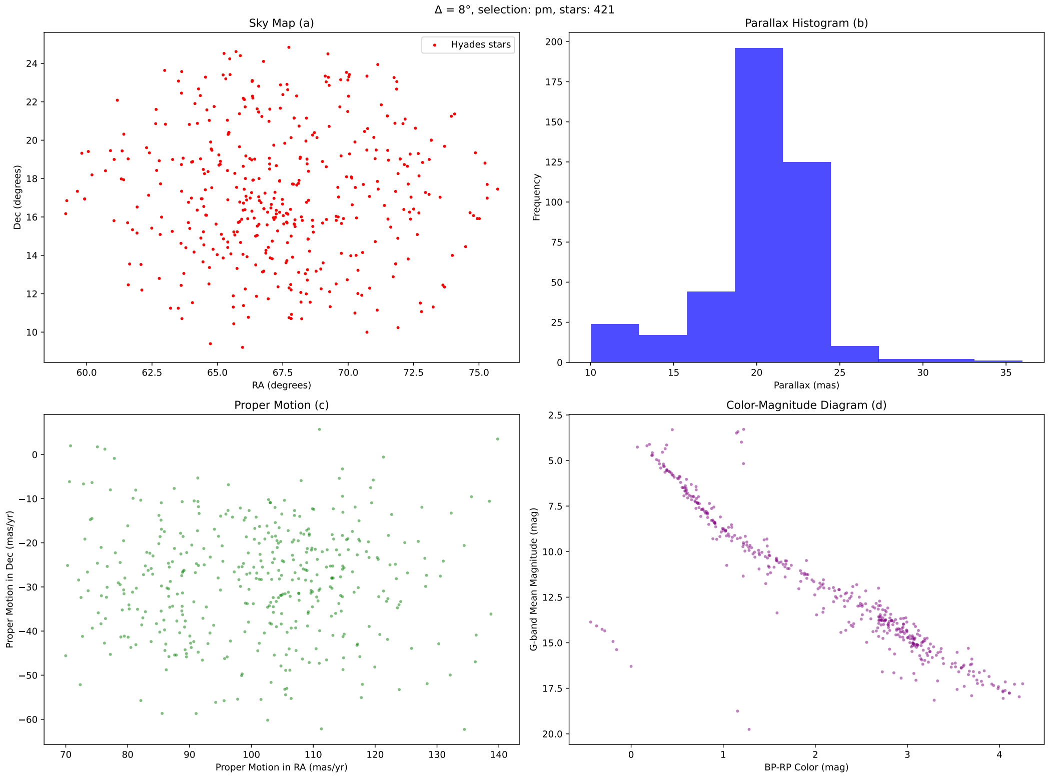


(b) delta = 8

Figure 3: A selection of the stars based on the delta value, and based on the parallax

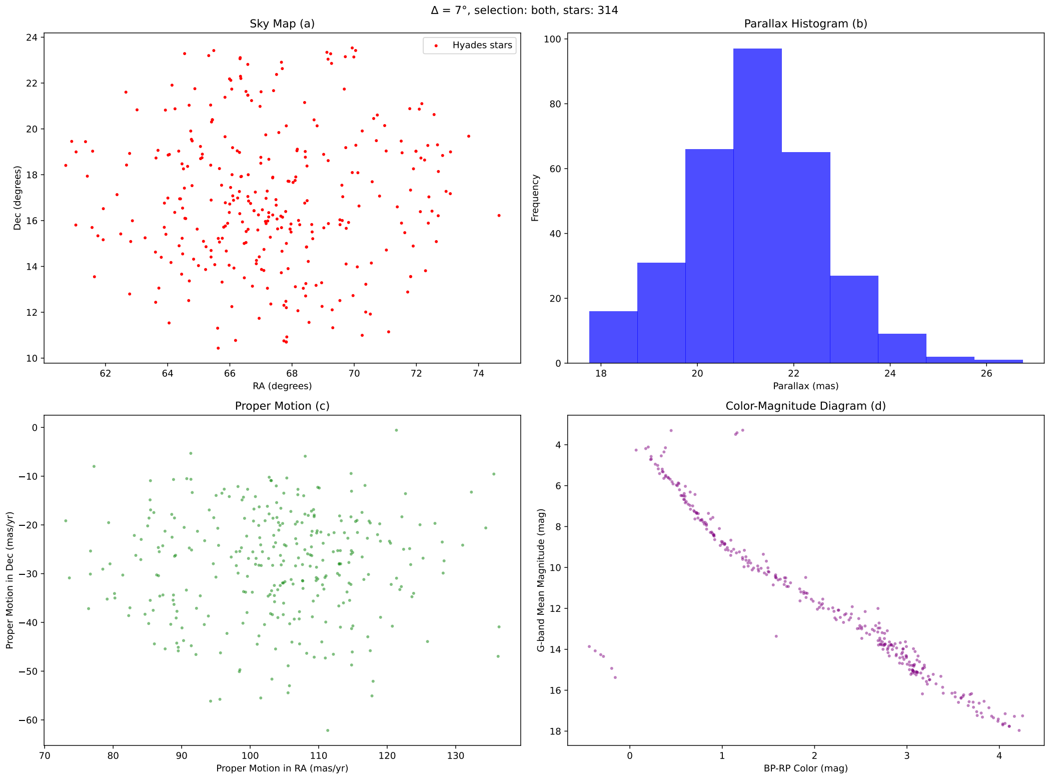


(a)

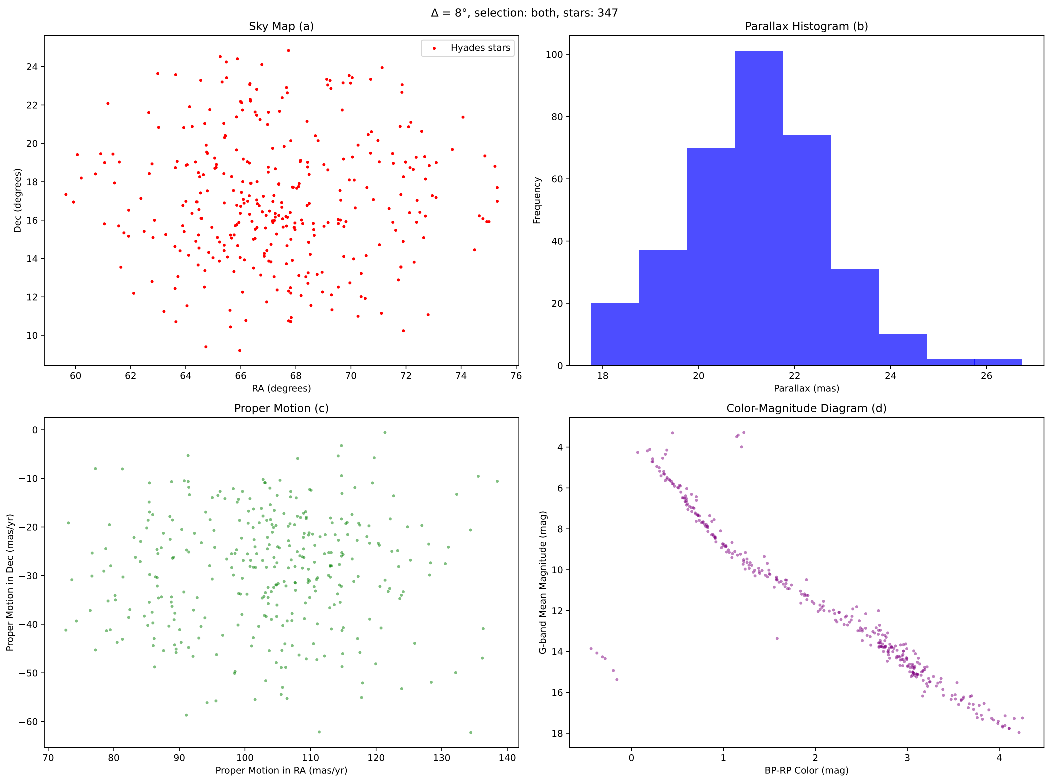


(b)

Figure 4: A selection of the stars based on the delta value, and based on the proper motion



(a)



(b)

Figure 5: A selection of the stars based on the delta value, based on the parallax and based on the proper motion.

3.2 Construction CMD

Now that we have made our final and best selection, we can focus on just the CMD. for clarity, we construct the CMD directly using the data from GAIA. The G-band mean magnitude has already been given including the BP-RP colour. BP stands blue photometer and RP for red photometer. These do not have an error associated with them. The difference between the brightness in these filters give us the BP-RP colour. If we plot the G-band mean magnitude against this BP-RP colour we get a colour-magnitude diagram. From here we can identify the main sequence, the dwarfs, and the giants very easily. The main sequence is a somewhat straight diagonal line where most stars lie. The dwarfs are located in the bottom left, and the giants lie above and to the right of the main sequence. To verify our selection of stars for our CMD, we can plot some stars we know for sure are in the Hyades. The brightest stars are θ^1, θ^2 tauri and HD 28527. the CMD with these stars highlighted is seen in 6

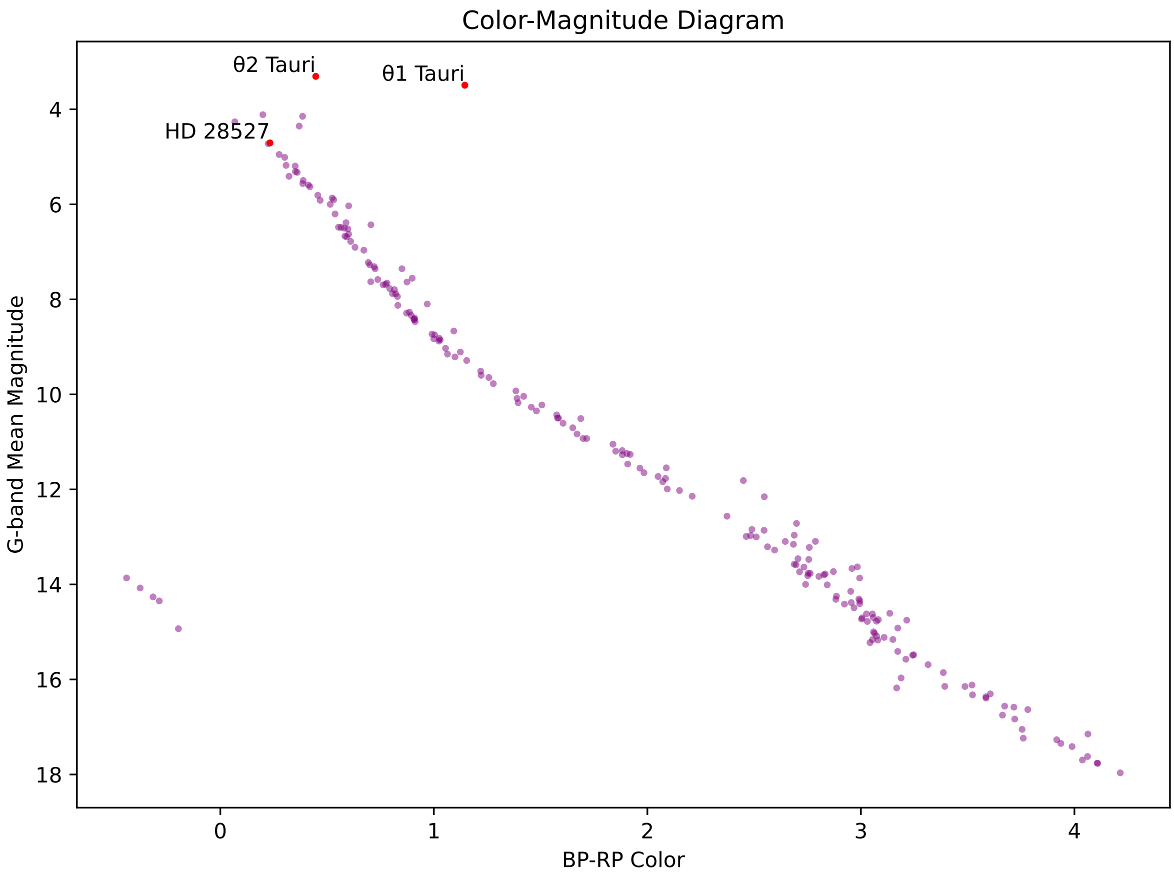


Figure 6: The CMD of the Hyades. To verify the result, the brightest stars have been highlighted and they are indeed in our selection.

We can immediately make use of the CMD to tell us if these stars are the most massive. Since more massive stars appear higher in the CMD, we can conclude that θ^1 Tauri is one of the most massive stars in the Hyades. θ^2 Tauri is lacking behind, but is still evolved away from the main sequence. Thus the double star system is one of the most massive, with θ^1 Tauri being the brightest. HD28527 is seen at the end of the main sequence, which means its starting to evolve off it. Since heavier stars age faster we can conclude that HD25257 is still relatively massive, but just not part of the most massive stars.

We can also estimate the distance modulus of the Hyades. We do this by finding where the star Vega would lie, with a BP-RP colour of zero. We could, theoretically, do this by fitting a line to the CMD. This is actually not a good method because its very hard to fit an appropriate line to the CMD. The better way to do this is by estimating with our eyes and making an educated guess on the error. Using this method, we get that the apparent magnitude of Vega is 3.9 $\pm$ 0.3 mag. The error on this measurement is substantial. To get the distance modulus, we use that the Absolute magnitude of Vega is 0.582 mag. The distance modulus according to formula 3 is 3.42 mag. From this, we can also get another estimate of the distance of the Hyades. Using the formula 4 and 5, we get a distance d = 46 $\pm$ 6 parsecs

We can also get the distance by using the parallax. The first step is to get the average parallax from the selection. We do this by utilizing a weighted average and corresponding error as shown in equations 6 and 3.2. This minimizes the error in our average.

$$\bar{\pi} = \frac{\sum_{i=1}^{n} \frac{\sigma_i}{\pi_i^2}}{\sum_{i=1}^{n} \frac{1}{\sigma_i^2}} \tag{6}$$

$$\sigma_{\bar{\pi}} = \sqrt{\frac{1}{\sum_{i=1}^{n} \frac{1}{\sigma_i^2}}} \tag{7}$$

$\bar{\pi}$: average parallax value
π_i : individual value of all stars

Using these formulas The mean parallax is 21 $\pm$ 0.4 mas Then we can use equations 1 and 2 to get the distance and error from the parallax. This yields a mean distance of the Hyades of 47.2 $\pm$ 0.8 parsecs

3.3 Isochrone Comparison

Now that we have our final CMD, we can estimate its age using the theoretical evolution with isochrones. For this we use the same data as the official main Gaia DR2 release. As cited in the paper : "Here we made use of PARSEC isochrones (Chenet al. 2014) for metallicites Z= 0.017 and Z = 0.020 updated to the latest transmission curve calibrated on Gaia DR2 data (Evans et al. 2018)". The isochrones for the Hyades are for Z = 0.020. This data still has to be accounted to our specific star cluster. We do this by subtracting the distance modulus we have calculated from the isochrones. We get the following result seen in figure 7.

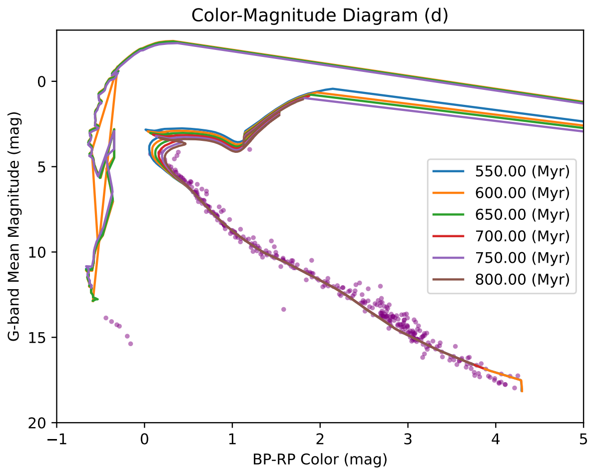


Figure 7: CMD of the Hyades showing several isochrones that lie closest to the data.

By fitting with our eyes we get that the best line is the isochrone corresponding with an age of 650 million years. This single age is seen in figure 8. The error can be determined by looking at the isochrones closest to the age and saying that is our error, So the final age estimate is 650 $\pm$ 50 million year.

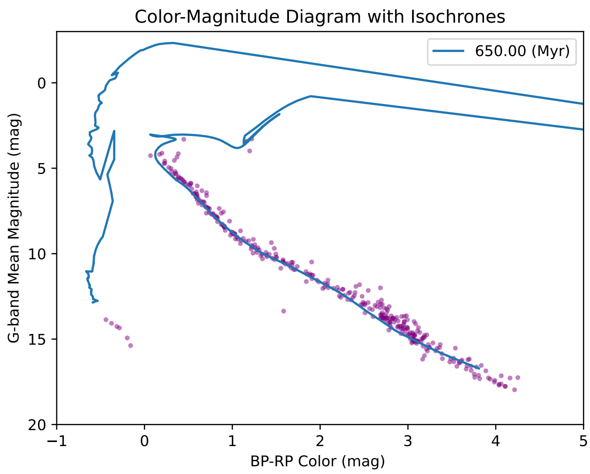


Figure 8: CMD of the Hyades showing our final isochrone for an age of 650 million years. This is the best fit.

4 Discussion

There are some ways to improve these results. First of all, is the way stars are selected. Since the Hyades is an open cluster, over a third of the stars have already escaped its tidal radius, where the milky way gravitational attraction dominates over that of the other stars in the Hyades. This mean it would be ideal to have a bigger search area regarding the parallax. The proper motion selecting range is also quite substantial, meaning the spread of the proper motions of the stars is large. Some whole different way of selecting stars would be ideal, one that take into accounts the large possible ranges and perhaps adds another condition. According to literature, There are between 200 - 300 stars that are sure to be part of the Hyades using its tidal radius. This barely matches the number of stars in our selection. This means that while we lie in the correct interval, we probably sorted a bit too aggressively.

Furthermore, A different methodology for finding the distance modulus would be very beneficial. The error on the distance modulus is way too large resulting in a large error on the distance using this method. One could improve this by correctly fitting a line on data of the main sequence only. This would result in a much smaller and more quantitative error, which leads to a better estimate of the distance. The value of distance using the distance modulus lies closely to our value obtained using parallax. These values also closely match the literature, which set the distance to be 46.85 ± 0.92. The error of our distance using distance modulus has a way larger error, and the error of the distance using parallax seems to be in agreement with the literature. Our error is probably a bit smaller, because we simply have less stars in our selection and thus have a smaller spread.

Lastly, the isochrones are very hard to fit by eye. Since there is a lot of spread around the main sequence turnoff. The alternative is therefore a better methodology. A more quantitative fit would result in a better estimate of the age, as well as a smaller error. Another way to improve the fit by eye is by increasing the stars around the main sequence turnoff. This would mean a better selection, but would also have a big limitation. There could also simply not be any more stars in the Hyades in this particular area of the CMD. The literature states the age to be 625 ± 50 million years, which aligns strongly with our result.

5 Conclusions

In conclusion, our study leverages the precision of Gaia Data Release 2 to refine our understanding of the Hyades, the nearest open star cluster to Earth. By employing a multifaceted approach, combining positional, parallax, and proper motion criteria, we find the cluster's boundaries with reasonable accuracy. Our comprehensive selection method, utilizing a circular positional boundary, a rectangular proper motion region, and a parallax range informed by the cluster's tidal radius, yields the most robust identification of Hyades members. Through the construction of a Color-Magnitude Diagram (CMD) from Gaia data, we derive a distance modulus of 3.4 ± 0.3, consistent with previous literature. Our distance estimates, obtained both from parallax (47.2 ± 0.8 parsecs) and distance modulus (46 ± 6 parsecs), align closely with each other and with established values. Moreover, employing isochrone data consistent with official Gaia DR2 publications, we estimate the age of the Hyades cluster to be 600 ± 50 million years. Our quantitative findings not only reinforce existing knowledge but also underscore the utility of Gaia DR2 in advancing our understanding of stellar clusters, facilitating future investigations into their properties and evolution.

References

Gaia, C., Brown, A., Vallenari, A., Prusti, T., De Bruijne, J., Babusiaux, C., Juhász, Á., Marschalkó, G., Marton, G., Molnár, L., et al. (2018). Gaia data release 2 summary of the contents and survey properties. *Astronomy & Astrophysics*, 616(1).

Koposov, S., Collaboration, G., et al. (2018). Gaia data release 2. observational hertzsprung-russell diagrams. *Astronomy & Astrophysics*, 616.

Perryman, M. A., Brown, A., Lebreton, Y., Gomez, A., Turon, C., De Strobel, G. C., Mermilliod, J., Robichon, N., Kovalevsky, J., and Crifo, F. (1997). The hyades: distance, structure, dynamics, and age. *arXiv preprint astro-ph/9707253*.

Appendix

```
import numpy as np
from astropy.io import fits
import matplotlib.pyplot as plt
from astropy.coordinates import SkyCoord
import astropy.coordinates as cd
from astropy import units as u

file_path = r"Gaia_dr2_in100pc1.fits"



# In[2]:


# Open the FITS file
with fits.open(file_path) as hdul:
    # Print header information
    header = hdul[1].data.columns
    print(header)


# In[3]:


# Exercise 1
data = fits.getdata(file_path)

# xtract necessary information
ra = data['ra']
dec = data['dec']
parallax = data['parallax']
pmra = data['pmra']
pmdec = data['pmdec']
phot_g_mean_mag = data['phot_g_mean_mag']
bp_rp = data['bp_rp']

# Error estimates on parameters
ra_error = data['ra_error']
dec_error = data['dec_error']
parallax_error = data['parallax_error']
pmra_error = data['pmra_error']
```

```
pmdec_error = data['pmdec_error']

# These dont have an error
#phot_g_mean_mag_error = data['phot_g_mean_mag_error']
#bp_rp_error = data['bp_rp_error']




# In[4]:


# Exercise 1 continuation
def find_delta_values(delta_values, selection=None):
    # Step 1: Obtain coordinates of the Hyades cluster
    icrs_coord = cd.get_icrs_coordinates('Hyades')

    # Extract RA and Dec components
    RA_Hyades, Dec_Hyades = icrs_coord.ra.degree, icrs_coord.dec.degree
    for delta in delta_values:
        # Initialize an empty list to store the number of stars for each
                                                value
        num_stars_selected = []
        # Define angular distance in degrees
        delta_deg = delta * u.deg

        # Create a SkyCoord object for the Hyades cluster
        hyades_coords = SkyCoord(ra=RA_Hyades * u.deg, dec=Dec_Hyades * u.deg,
                                                frame='icrs')

        # Select stars within a circular region around the Hyades coordinates
        separation = hyades_coords.separation(SkyCoord(ra=ra*u.deg, dec=dec*u.
                                                deg))
        hyades_parallax = 21.052 # Simbad
        hyades_distance = 47
        parsec_error = 10 # Range
        parallax_minimum =  1000 / (hyades_distance + parsec_error)
        parallax_maximum = 1000 / (hyades_distance - parsec_error)
        pmra_hyades = 104.92 #simbad
        pmra_error = 35
        pmra_minimum = pmra_hyades - pmra_error
        pmra_maximum = pmra_hyades + pmra_error
        pmdec_error = 35
        pmdec_hyades = -28.00 #simbad
        pmdec_minimum = pmdec_hyades - pmdec_error
        pmdec_maximum = pmdec_hyades + pmdec_error
        if selection  == 'parallax':
            mask = ((separation < delta_deg)
                    & (data["parallax"] > parallax_minimum)
                    & (data["parallax"] < parallax_maximum))
        elif selection == 'pm':
            mask = ((separation < delta_deg)
                    & (data["pmra"] > pmra_minimum)
                    & (data["pmra"] < pmra_maximum)
                    & (data["pmdec"] > pmdec_minimum)
                    & (data["pmdec"] < pmdec_maximum))
        elif selection == 'both':
            mask = ((separation < delta_deg)
                    & (data["parallax"] > parallax_minimum)
                    & (data["parallax"] < parallax_maximum)
```

```
            & (data["pmra"] > pmra_minimum)
            & (data["pmra"] < pmra_maximum)
            & (data["pmdec"] > pmdec_minimum)
            & (data["pmdec"] < pmdec_maximum))
else:
    mask = separation < delta_deg
# Count the number of stars in the selected subset
num_stars = np.sum(mask)

# Append the number of stars to the list
num_stars_selected.append(num_stars)
print(f"For     = {delta}  , Number of selected stars: {num_stars}")

# Plot settings
plt.figure(figsize=(16, 12))

# Adding a main title above all subplots
plt.suptitle(f'    = {delta}  , selection: {selection}, stars: {
                                        num_stars}')

# Subplot (a): Map of positions (RA, Dec) in the sky
plt.subplot(2, 2, 1)
#plt.scatter(ra, dec, s=1, c='black', alpha=0.5)
plt.scatter(ra[mask], dec[mask], s=5, c='red', label='Hyades stars')
plt.xlabel('RA (degrees)')
plt.ylabel('Dec (degrees)')
plt.title('Sky Map (a)')
plt.legend(loc='upper right')

# Subplot (b): Histogram of parallax values
# Choose bins according to Sturge's Rule
plt.subplot(2, 2, 2)
plt.hist(parallax[mask], bins=int(1 + 3.322*np.log10(num_stars)), color=
                                        'blue', alpha=0.7)
plt.xlabel('Parallax (mas)')
plt.ylabel('Frequency')
plt.title('Parallax Histogram (b)')

# Subplot (c): Plot of proper motion in RA and Dec
plt.subplot(2, 2, 3)
plt.scatter(pmra[mask], pmdec[mask], s=5, c='green', alpha=0.5)
plt.xlabel('Proper Motion in RA (mas/yr)')
plt.ylabel('Proper Motion in Dec (mas/yr)')
plt.title('Proper Motion (c)')

# Subplot (d): CMD of G versus Gbp    Grp
plt.subplot(2, 2, 4)
plt.scatter(bp_rp[mask], phot_g_mean_mag[mask], s=5, c='purple', alpha=0
                                        .5)
plt.gca().invert_yaxis()
plt.xlabel('BP-RP Color (mag)')
plt.ylabel('G-band Mean Magnitude (mag)')
plt.title('Color-Magnitude Diagram (d)')

# Adjust layout
plt.tight_layout()

if selection  == 'parallax':
    plt.savefig(f'hyades_plots_delta_{delta}_parallax.pdf')
elif selection == 'pm':
```

```python
            plt.savefig(f'hyades_plots_delta_{delta}_pm.pdf')
        elif selection =='both':
            plt.savefig(f'hyades_plots_delta_{delta}_both.pdf')
        elif delta > 30 :
            plt.savefig(f'hyades_plots_delta_unfiltered.png')
        else:
            plt.savefig(f'hyades_plots_delta_{delta}.pdf')

        # Show the plot
        plt.show()




# In[5]:


find_delta_values([1000])


# In[6]:


# Execercise 2
# Choose a range of values for
delta_values = np.arange(2, 21, 1)  # between 2  and 20

find_delta_values(delta_values)


# In[7]:


# Exercise 3

# I chose 2 delta values
delta_values = [7,8]

find_delta_values(delta_values, selection='parallax')


# In[8]:


find_delta_values(delta_values, selection='pm')


# In[9]:


find_delta_values(delta_values, selection='both')


# In[10]:


# Exercise 4

delta_best = 8
```

```
delta_best_deg = delta_best * u.deg

icrs_coord = cd.get_icrs_coordinates('Hyades')

# Extract RA and Dec components
RA_Hyades, Dec_Hyades = icrs_coord.ra.degree, icrs_coord.dec.degree
# Create a SkyCoord object for the Hyades cluster
hyades_coords = SkyCoord(ra=RA_Hyades * u.deg, dec=Dec_Hyades * u.deg, frame='
                                        icrs')

# Select stars within a circular region around the Hyades coordinates
separation = hyades_coords.separation(SkyCoord(ra=ra*u.deg, dec=dec*u.deg))

# Lets make new data based on our best found delta value.
hyades_parallax = 21.052
hyades_distance = 47
parsec_error = 10 # Tidal radius
parallax_minimum =  1000 / (hyades_distance + parsec_error)
parallax_maximum = 1000 / (hyades_distance - parsec_error)
pmra_hyades = 104.92 # Simbad
pmra_error = 35 # Estimate
pmra_minimum = pmra_hyades - pmra_error
pmra_maximum = pmra_hyades + pmra_error
pmdec_error = 35 # Estimate
pmdec_hyades = -28.00 # Simbad
pmdec_minimum = pmdec_hyades - pmdec_error
pmdec_maximum = pmdec_hyades + pmdec_error

# Create mask
delta_best = 8
delta_deg_best = delta_best * u.deg
mask = ((separation < delta_deg_best)
        & (data["parallax"] > parallax_minimum)
        & (data["parallax"] < parallax_maximum)
        & (data["pmra"] > pmra_minimum)
        & (data["pmra"] < pmra_maximum)
        & (data["pmdec"] > pmdec_minimum)
        & (data["pmdec"] < pmdec_maximum))

# Load the masked data in seperate variables
hyades_data = data[mask]
hyades_ra = hyades_data['ra']
hyades_dec = hyades_data['dec']
hyades_parallax = hyades_data['parallax']
hyades_pmra = hyades_data['pmra']
hyades_pmdec = hyades_data['pmdec']
hyades_phot_g_mean_mag = hyades_data['phot_g_mean_mag']
hyades_bp_rp = hyades_data['bp_rp']

# Error estimates on parameters
hyades_ra_error = hyades_data['ra_error']
hyades_dec_error = hyades_data['dec_error']
hyades_parallax_error = hyades_data['parallax_error']
hyades_pmra_error = hyades_data['pmra_error']
hyades_pmdec_error = hyades_data['pmdec_error']


# Cacluclate mean parallax using weighted average and corresponding error
mean_parallax = np.sum(hyades_parallax/(hyades_parallax_error**2)) / np.sum(1/(
                                        hyades_parallax_error**2))
```

```
mean_parallax_error = np.sqrt(1/ np.sum((hyades_parallax_error**2)))
print(f'{mean_parallax:.2f}    {mean_parallax_error:.2f} mas')

# Calculate distance from parallax using standard relation
distance = 1 / (mean_parallax/1000)
distance_error = (mean_parallax_error/1000) / ((mean_parallax/1000)**2)
print(f'{distance:.2f}     {distance_error:.2f} parsec')

# Apparent magnitude of Vega
m_G_Vega = 0

# Absolute magnitude of Vega
M_G_Vega = 0.58


# Distance modulus from CMD
# Assuming the vertical position of the main sequence gives the apparent
                                        magnitude
# You may need to adjust this based on your actual data and methodology
# For this example, let's assume you've identified the main sequence apparent
                                        magnitude
m_G_main_sequence = 3.9

# Distance modulus from CMD
distance_modulus_CMD = m_G_main_sequence - M_G_Vega
distance_modulus_error = 0.3
print(f"The distance modulus is {distance_modulus_CMD}")

# Calculate the distance
distance_CMD = 10 ** ((distance_modulus_CMD + 5) / 5)
distance_CMD_error = 0.2*np.log(10)*distance_CMD * distance_modulus_error
print(f"Distance to the Hyades from CMD: {distance_CMD:.2f} +- {
                                        distance_CMD_error:.2f} parsecs")

# Compare distances
distance_difference = np.abs(distance - distance_CMD)
print(f"Difference in distances: {distance_difference:.2f} parsecs")


# In[11]:


# Exercise 5

# Plotting isochrones

isochrone_file = r'isochronesGaiaDR2_z0.020.dat'
chrons = np.genfromtxt(fname=isochrone_file, skip_header=16, names=True,
                       usecols=(0,7,8,9))
'''
-The datafile contains the following structure:
- The first line give the number N_c of colour passbands that are used for each
                                        entry,
- Then follow N_c lines describing these colours, ending with a reference
- After that there is a header line, describing the columns
  The columns are Log(age[years]), Log(mass[solar_mass]), Log(luminosity[
                                          solar_lum]), Log(effective temperature
                                          [K]), [ N_c times
                                          Magnitude_in_colour_filter ]
- Followed by the very many isochrone points
```

```
'''
timesteps = np.unique(chrons["LOG_T"])
plt.figure()
plt.xlim(-3,10)
plt.ylim(-7,17)
#print(timesteps)
print("I have ", len(timesteps), "different ages")
years = [5.7, 6 ,6.3,7, 7.3, 8]
#years = [6.5]
for year in years:
    timesteps = np.unique(chrons["LOG_T"])
    # Assume a give age for the target
    targage = year * 1E8 # In yr
    # Calculate the difference between age in the isochrones and my target age
    deltaage = np.abs(timesteps - np.log10(targage))
    # Select the index where this is minimum
    iage = np.argmin(deltaage)
    print("Nearest age: ", iage, timesteps[iage], 10**timesteps[iage] / 1.e6,
          "(Myr)")

    # Select that isochrone
    isoc = chrons[chrons["LOG_T"]==timesteps[iage]]

    # Super basic plot
    plt.plot(isoc['M5']-isoc['M6'], isoc['M4'] + distance_modulus_CMD, label= f"
                                                       {(10**timesteps[iage] / 1.e6):.2f} (
                                                       Myr)")

# Make the CMD
plt.scatter(hyades_bp_rp, hyades_phot_g_mean_mag, s=5, c='purple', alpha=0.5)
plt.gca().invert_yaxis()
plt.xlabel('BP-RP Color (mag)')
plt.ylabel('G-band Mean Magnitude (mag)')
plt.title('Color-Magnitude Diagram (d)')
plt.xlim(-1,5)
plt.ylim(20, -3)
plt.legend()


plt.savefig("isochrones.pdf")
plt.show()


# In[12]:


years = [6.5]
for year in years:
    timesteps = np.unique(chrons["LOG_T"])
    # Assume a give age for the target
    targage = year * 1E8 # In yr
    # Calculate the difference between age in the isochrones and my target age
    deltaage = np.abs(timesteps - np.log10(targage))
    # Select the index where this is minimum
    iage = np.argmin(deltaage)
    print("Nearest age: ", iage, timesteps[iage], 10**timesteps[iage] / 1.e6,
          "(Myr)")

    # Select that isochrone
```

```
    isoc = chrons[chrons["LOG_T"]==timesteps[iage]]

    # Super basic plot

    plt.plot(isoc['M5']-isoc['M6'], isoc['M4'] + distance_modulus_CMD, label= f"
                                        {(10**timesteps[iage] / 1.e6):.2f} (
                                        Myr)")

# Make the CMD
plt.scatter(hyades_bp_rp, hyades_phot_g_mean_mag, s=5, c='purple', alpha=0.5)
plt.gca().invert_yaxis()
plt.xlabel('BP-RP Color (mag) ')
plt.ylabel('G-band Mean Magnitude (mag)')
plt.title('Color-Magnitude Diagram with Isochrones')
plt.xlim(-1,5)
plt.ylim(20, -3)
plt.legend()


plt.savefig("isochrones_best.pdf")
plt.show()


# In[13]:


# Exercise 6
# Make the CMD
plt.figure(figsize=(8, 6))
plt.scatter(hyades_bp_rp, hyades_phot_g_mean_mag, s=5, c='purple', alpha=0.5)
plt.gca().invert_yaxis()
plt.xlabel('BP-RP Color (mag)')
plt.ylabel('G-band Mean Magnitude (mag) ')
plt.title('Color-Magnitude Diagram with Isochrones')

# Define specific stars you want to highlight
star_names = ['HD 28527',' 1  Tauri',' 2  Tauri']
star_parallax = []
for star in star_names:
    atau = cd.get_icrs_coordinates(star)

    search_accuracy = 2
    match = (np.abs(hyades_dec-atau.dec.deg) < search_accuracy / 3600) & \
        (np.abs(hyades_ra-atau.ra.deg) < search_accuracy / 3600)
    if (match.sum() == 1):
        star_parallax.append(hyades_parallax[match])
        plt.scatter(hyades_bp_rp[match], hyades_phot_g_mean_mag[match], s=5, c='
                                            red', label=f'{star}')
        plt.text(hyades_bp_rp[match], hyades_phot_g_mean_mag[match], star,
                                            fontsize=10, ha='right', va='
                                            bottom')
    elif (match.sum() > 1):
        print("Too much stars found, choose a smaller search accuracy")
    else:
        print("Nothing found in these data")


print(star_parallax)
# Adjust layout
plt.tight_layout()
```

```
# Save the plot with a different name for each    value
plt.savefig(f'final_hyades_plots_delta_{delta_best}.pdf')

# Show the plot
plt.show()
```