

```
In [ ]: # PROBLEMA 1
# The star 9 Sagittarii is a star in the principal sequence with a spectral
# Its apparent magnitude is mv = 6.0. What is the distance to 9 Sagittarii (
import math
Mv = -5.7
mv = 6.0
d = 10**((mv - Mv + 5)/5)
print("The distance to 9 Sagittarii is", d, "kpc")
# change units to light years
d = d*3.262
print("The distance to 9 Sagittarii is", d, "light years")
```

The distance to 9 Sagittarii is 2187.761623949552 kpc

The distance to 9 Sagittarii is 7136.478417323438 light years

PROBLEMA 2

Esto se debe a diferencias en el tamaño relativo y la luminosidad de las dos estrellas que forman el sistema. Si la estrella más luminosa pasa por delante de la otra, el eclipse causará una mayor disminución en el brillo total del sistema, por tanto el mínimo en la curva de luz será de mayor profundidad. De darse el caso contrario, y la estrella menos luminosa pase por delante, el eclipse causará una menor disminución en el brillo total del sistema, y un mínimo de menor profundidad.

```
In [ ]: # Apartado 2
# Demonstrate that in a binary eclipsing system, the primary eclipse (the de
import numpy as np
import matplotlib.pyplot as plt

# Generate synthetic data for two stars with different temperatures
time = np.linspace(0, 100, 1000) # Time array
hot_star_flux = np.random.normal(1, 0.05, 1000) # Flux of the hotter star
cool_star_flux = np.random.normal(1.05, 0.05, 1000) # Flux of the cooler st

# Simulate eclipses
hot_eclipse = np.ones_like(time) # No eclipse for the hotter star
cool_eclipse = np.ones_like(time) # No eclipse for the cooler star

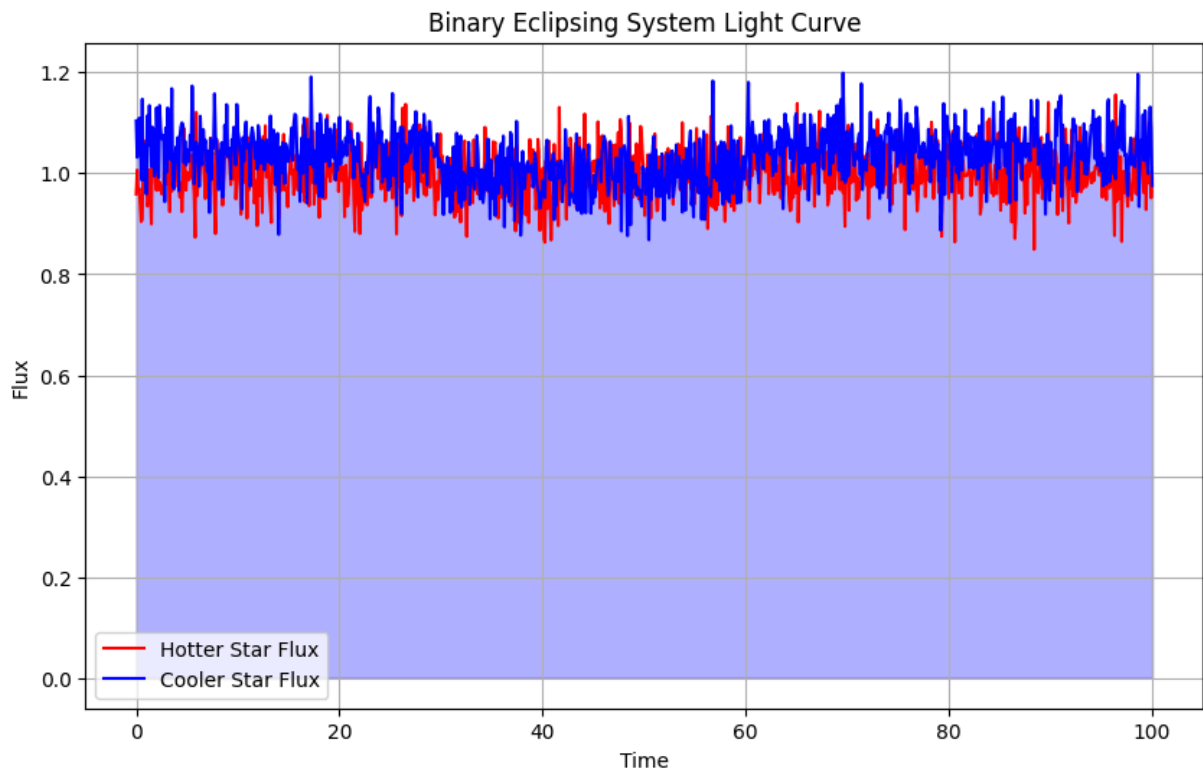
# Define the time interval for the eclipse
eclipse_start = 300
eclipse_end = 600

# Simulate eclipse for the cooler star (offset for visibility)
cool_eclipse[eclipse_start:eclipse_end] = 0.95

# Plot the light curve
plt.figure(figsize=(10, 6))
plt.plot(time, hot_star_flux, label='Hotter Star Flux', color='red')
plt.plot(time, cool_star_flux * cool_eclipse, label='Cooler Star Flux', colc
```

```
plt.fill_between(time, cool_star_flux * cool_eclipse, color='blue', alpha=0.5)
plt.xlabel('Time')
plt.ylabel('Flux')
plt.title('Binary Eclipsing System Light Curve')
plt.legend()
plt.grid(True)
plt.show()
```

```
print("La gráfica muestra el flujo de dos estrellas a lo largo de un tiempo.")
print("Esto se debe a que el eclipse primario ocurre cuando la estrella más
```



La gráfica muestra el flujo de dos estrellas a lo largo de un tiempo. El flujo de la estrella fría decrece durante el eclipse. Esto se debe a que el eclipse primario ocurre cuando la estrella más caliente es eclipsada por la estrella más fría.

```
In [ ]: #pip install astroquery
```

```
In [ ]: # PROBLEMA 3
# Apartado 1
%pip install astroquery
from astroquery.gaia import Gaia
import matplotlib.pyplot as plt

job = Gaia.launch_job("""
SELECT TOP 10000
phot_g_mean_mag + 5 * LOG10(parallax) - 10 AS mg, bp_rp
FROM gaiadr2.gaia_source
WHERE parallax_over_error > 10
AND phot_g_mean_flux_over_error > 50
AND phot_rp_mean_flux_over_error > 20
AND phot_bp_mean_flux_over_error > 20
AND bp_rp_excess_factor < 1.3 + 0.06 * POWER(phot_bp_mean_mag - phot_g_mean_mag, 2)
""")
```

```

AND phot_bp_rp_excess_factor > 1.0 + 0.015 * POWER(phot_bp_mean_mag - phot
AND visibility_periods_used > 8
AND astrometric_chi2_al / (astrometric_n_good_obs_al - 5) < 1.44 * GREATES
AND phot_bp_mean_mag < 19.5
""")
data = job.get_results()
# Print the length of the data
print(len(data))

# Apartado 4
# Visualize what columns has the table downloaded
print(data.colnames)
data.pprint(max_lines=10)

# Apartado 8
# Plot the HR diagram
plt.figure(figsize=(10, 10))
plt.hist2d(data['bp_rp'], data['mg'], bins=100, cmap='inferno')
plt.colorbar(label='Número de estrellas')
plt.xlabel('Color BP - RP')
plt.ylabel('Magnitud G')
plt.show()

plt.figure(figsize=(10, 6))
plt.scatter(data['bp_rp'], data['mg'], s=0.1, alpha=0.5)
plt.gca().invert_yaxis() # Invert the y-axis because the magnitude scale is
plt.xlabel('Color (BP - RP)')
plt.ylabel('Absolute Magnitude')
plt.title('Hertzsprung-Russell Diagram')
plt.grid(True)
plt.show()

print("Secuencia principal: La mayoría de las estrellas, incluyendo nuestro
print("Gigantes rojas: Estas estrellas han agotado el hidrógeno en sus núcle
print("Enanas blancas: Estas son las remanentes de estrellas de baja y media
print("Estrellas de la rama horizontal y la rama de las gigantes asintóticas

```

Requirement already satisfied: astroquery in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (0.4.7)Note: you may need to restart the kernel to use updated packages.

Requirement already satisfied: numpy>=1.18 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (1.26.4)

Requirement already satisfied: astropy>=4.2.1 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (6.0.0)

Requirement already satisfied: requests>=2.19 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (2.31.0)

Requirement already satisfied: beautifulsoup4>=4.8 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (4.12.3)

Requirement already satisfied: html5lib>=0.999 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (1.1)

Requirement already satisfied: keyring>=15.0 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (25.2.1)

Requirement already satisfied: pyvo>=1.1 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astroquery) (1.5.1)

Requirement already satisfied: pyerfa>=2.0 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astropy>=4.2.1->astroquery) (2.0.1.1)

Requirement already satisfied: astropy-iers-data>=0.2023.10.30.0.29.53 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astropy>=4.2.1->astroquery) (0.2024.3.25.0.29.50)

Requirement already satisfied: PyYAML>=3.13 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astropy>=4.2.1->astroquery) (6.0.1)

Requirement already satisfied: packaging>=19.0 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from astropy>=4.2.1->astroquery) (24.0)

Requirement already satisfied: soupsieve>1.2 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from beautifulsoup4>=4.8->astroquery) (2.5)

Requirement already satisfied: six>=1.9 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from html5lib>=0.999->astroquery) (1.16.0)

Requirement already satisfied: webencodings in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from html5lib>=0.999->astroquery) (0.5.1)

Requirement already satisfied: jaraco.classes in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from keyring>=15.0->astroquery) (3.4.0)

Requirement already satisfied: jaraco.functools in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\l

ocal-packages\python311\site-packages (from keyring>=15.0->astroquery) (4.0.1)
Requirement already satisfied: jaraco.context in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from keyring>=15.0->astroquery) (5.3.0)
Requirement already satisfied: importlib-metadata>=4.11.4 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from keyring>=15.0->astroquery) (7.1.0)
Requirement already satisfied: pywin32-ctypes>=0.2.0 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from keyring>=15.0->astroquery) (0.2.2)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from requests>=2.19->astroquery) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from requests>=2.19->astroquery) (3.6)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from requests>=2.19->astroquery) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from requests>=2.19->astroquery) (2024.2.2)
Requirement already satisfied: zipp>=0.5 in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from importlib-metadata>=4.11.4->keyring>=15.0->astroquery) (3.18.1)
Requirement already satisfied: more-itertools in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from jaraco.classes->keyring>=15.0->astroquery) (10.2.0)
Requirement already satisfied: backports.tarfile in c:\users\recio\appdata\local\packages\pythonsoftwarefoundation.python.3.11_qbz5n2kfra8p0\localcache\local-packages\python311\site-packages (from jaraco.context->keyring>=15.0->astroquery) (1.1.1)

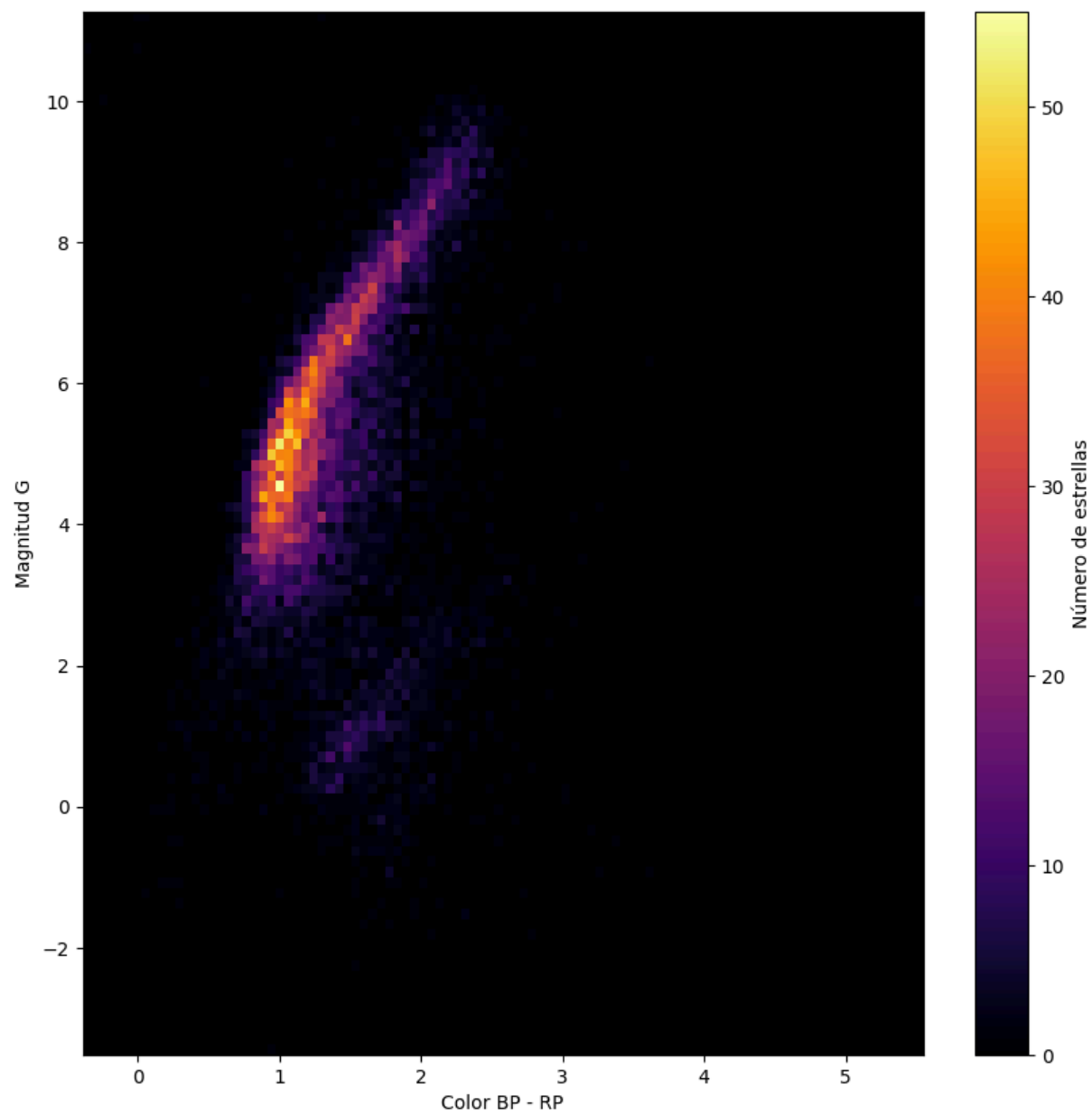
10000

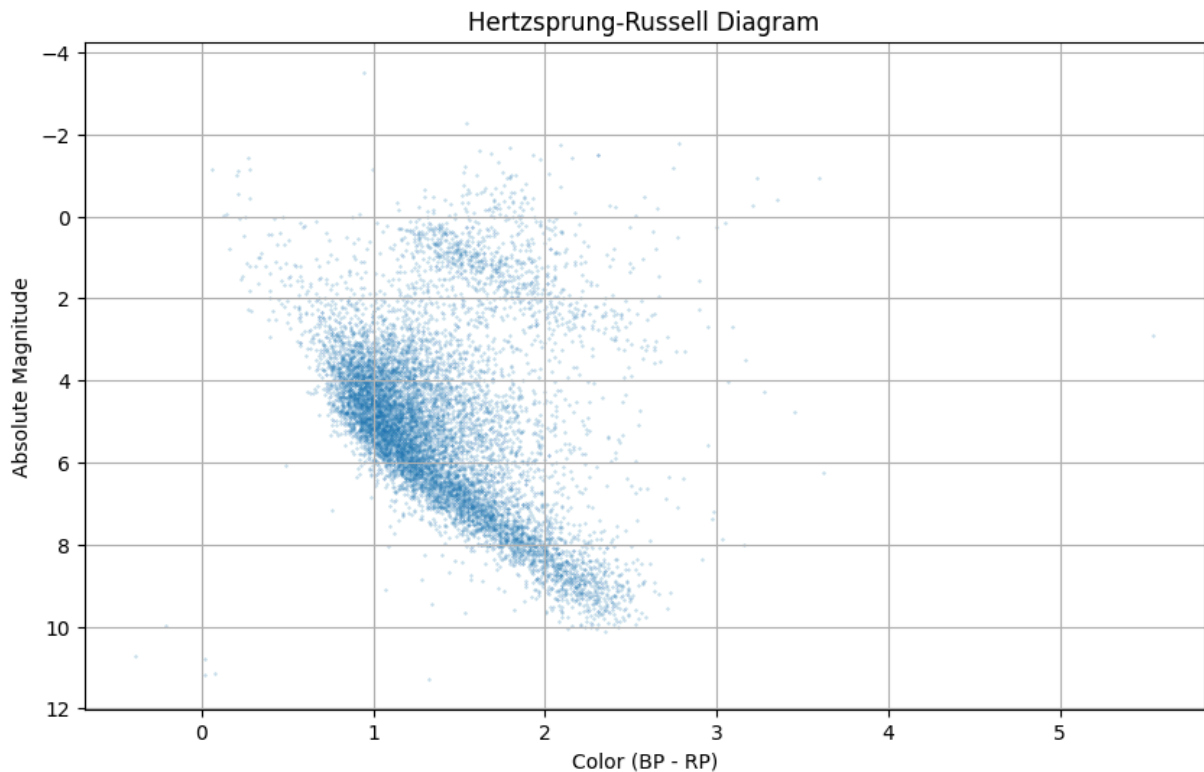
['mg', 'bp_rp']

mg	bp_rp
----	-------

0.6687816584015582	1.8660717
6.34421812224231	1.2628975
...	...
5.640886043582048	1.0158672
4.053360372760995	1.3260689
6.716542188274111	1.3154297

Length = 10000 rows





Secuencia principal: La mayoría de las estrellas, incluyendo nuestro Sol, se encuentran en la secuencia principal, que va desde la parte superior izquierda (estrellas azules, calientes y luminosas) a la parte inferior derecha (estrellas rojas, frías y débiles).

Gigantes rojas: Estas estrellas han agotado el hidrógeno en sus núcleos y se han expandido y enfriado, moviéndose hacia la parte superior derecha del diagrama.

Enanas blancas: Estas son las remanentes de estrellas de baja y media masa que han agotado su combustible nuclear. Son muy calientes pero de baja luminosidad, por lo que se encuentran en la parte inferior izquierda del diagrama. **Estrellas de la rama horizontal y la rama de las gigantes asimptóticas:** Estas son estrellas que están en etapas avanzadas de su evolución y se encuentran en regiones específicas del diagrama HR.

Apartado 5

Filtro G : Desde 330 hasta 1050 nm. Se utiliza para medir la magnitud en la banda G, que es la principal banda de observación de Gaia y proporciona información sobre el brillo aparente de las estrellas en una amplia gama de longitudes de onda.

Filtro BP y RP : El BP abarca desde 330 hasta 680 nm y el RP desde 630 hasta 1050 nm. Estos filtros permiten una mejor distinción del color de las estrellas y nos dan información sobre su temperatura superficial y composición.

Banda R : Desde 630 hasta 690 nm. Se utiliza para mediciones adicionales de color y brillo en determinadas regiones del espectro visible.

Apartado 6

parallax : proporciona la medida del paralaje estelar y tiene unidades de milisegundos de arco (mas).

phot_g_mean_mag : proporciona la magnitud promedio en la banda G (es el brillo aparente de una estrella en la banda G) y tiene unidades de magnitudes (mag).

bp_rp : proporciona el índice de color Bp-Rp que es la diferencia de magnitud entre las bandas azul y roja. Tiene unidades de magnitudes (mag).

Apartado 7

Proporciona la magnitud absoluta de una estrella a partir de su magnitud aparente en la banda G y su paralaje. Esto nos da información sobre la luminosidad intrínseca de la estrella. La magnitud absoluta (M) de una estrella se define como la magnitud que tendría si estuviera a una distancia de 10 parsecs. La magnitud aparente (m) es la magnitud de la estrella tal como se observa desde la Tierra.

La relación entre la magnitud aparente y la magnitud absoluta se da por la ecuación de la magnitud de la distancia:

$$m - M = 5 \log(d) - 5$$

donde d es la estrella en parsecs. Si resolvemos para M, obtenemos:

$$M = m - 5 \log(d) + 5$$

El paralaje de una estrella es el inverso de su distancia en parsecs, es decir, $p = 1/d$. Si sustituimos esto en la ecuación, obtenemos:

$$M = m - 5 \log(1/p) + 5 = m + 5 \log(p) - 5$$

que es la fórmula buscada. m es la magnitud aparente en la banda G (phot_g_mean_mag) y p es la paralaje.

```
In [ ]: # PROBLEMA 4
# Apartado 1
# Plot an histogram of the galactic latitude of the selected stars and comme
# the data is gaia_data.csv, load the data
import pandas as pd
import matplotlib.pyplot as plt
data = pd.read_csv('gaia_data.csv')
```



```

# Plot the histogram
plt.figure(figsize=(10, 6))
plt.hist(data['b'], bins=50, color='skyblue', edgecolor='black', alpha=0.7)
plt.xlabel('Galactic Latitude (b)')
plt.ylabel('Frequency')
plt.title('Histogram of Galactic Latitude')
plt.grid(True)
plt.show()

# Apartado 2
# Plot an histogram of the distances of the selected stars.
# What are the possible causes of the distribution of distances?
# Plot the histogram
# Extraer las distancias de paralaje (en parsecs)
distancias = 1 / (data['parallax'] * 1e-3) # Convertir de milisegundos de a
# Crear el histograma
plt.hist(distancias, bins=30, color='skyblue', edgecolor='black')
# Añadir etiquetas y título
plt.xlabel('Distancia (parsecs)')
plt.ylabel('Frecuencia')
plt.title('Histograma de Distancias a las Estrellas Seleccionadas')
# Mostrar el histograma
plt.show()
print("La distribución de distancias muestra un pico alrededor de 1000 parsecs")

# Apartado 3
# Plot a diagram of the galactic coordinates (l,b) of the selected stars.
# Use a scatter point with a point size of 0,0001 and the set_aspect('equal')
# Plot the diagram
plt.figure(figsize=(10, 6))
plt.scatter(data['l'], data['b'], s=0.0001, color='skyblue')
plt.xlabel('Galactic Longitude (l)')
plt.ylabel('Galactic Latitude (b)')
plt.title('Galactic Coordinates of Selected Stars')
plt.gca().set_aspect('equal', adjustable='box')
plt.grid(True)
plt.show()

# Apartado 4
# Plot the same diagram as in the previous point, but color in red the stars with positive radial velocity
# Plot the diagram
plt.figure(figsize=(10, 6))
# Estrellas con velocidad radial positiva (rojo)
plt.scatter(data[data['radial_velocity'] > 0]['l'], data[data['radial_velocity'] > 0]['b'], s=0.0001, color='red')
# Estrellas con velocidad radial negativa (azul)
plt.scatter(data[data['radial_velocity'] < 0]['l'], data[data['radial_velocity'] < 0]['b'], s=0.0001, color='blue')
plt.xlabel('Galactic Longitude (l)')
plt.ylabel('Galactic Latitude (b)')
plt.title('Galactic Coordinates of Selected Stars with Radial Velocity')
plt.gca().set_aspect('equal', adjustable='box')
plt.legend()
plt.grid(True)
plt.show()

# Apartado 5
# Plot the same diagram as in the previous point, but assign a transparency

```

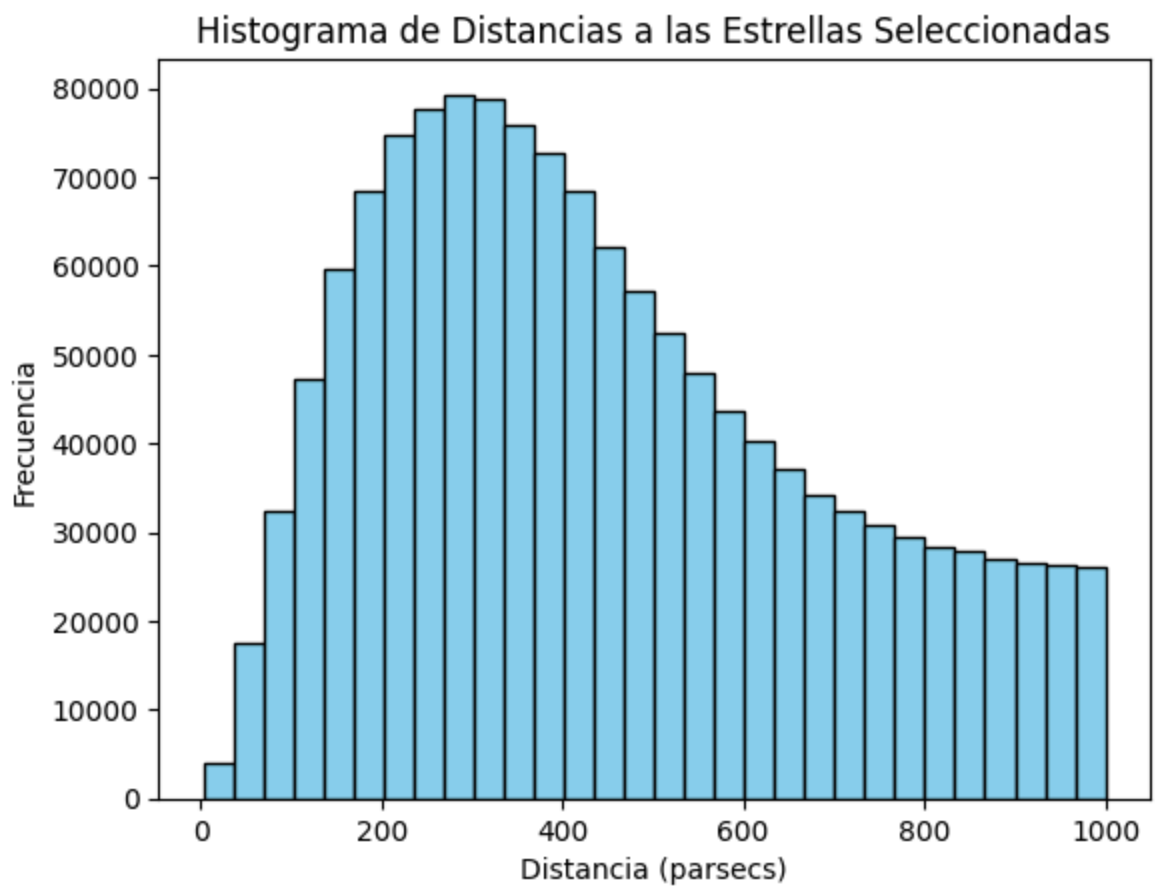
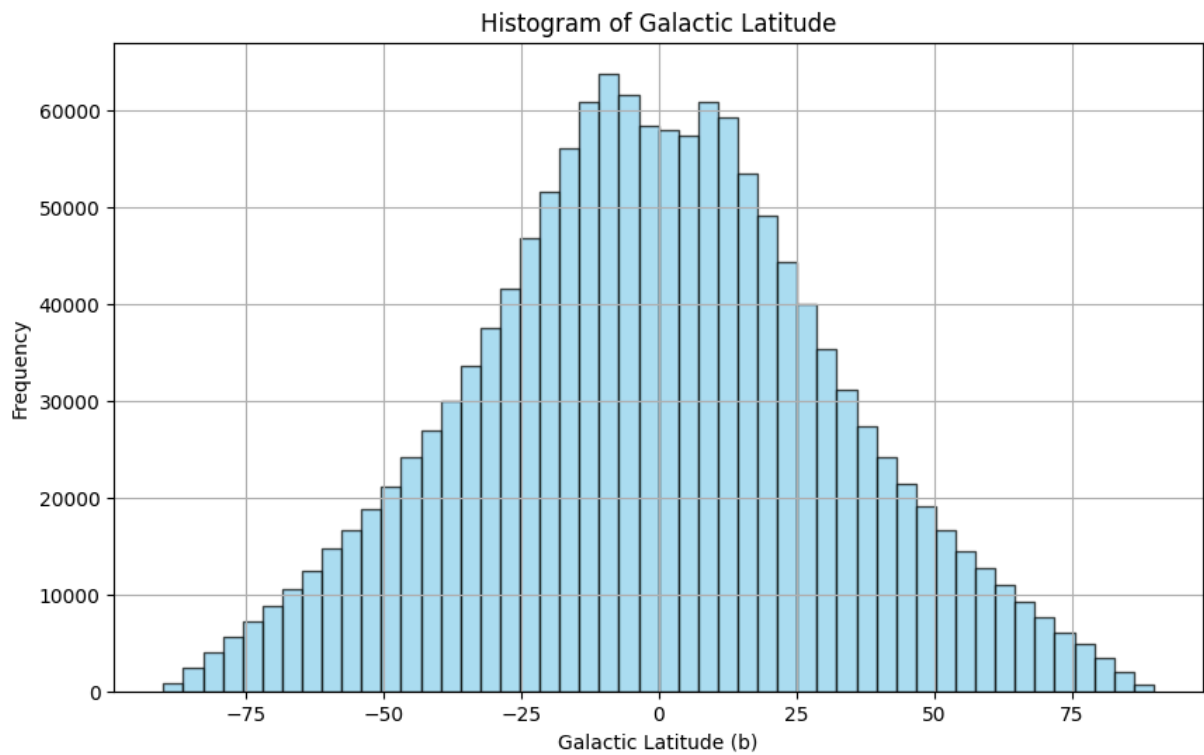
```

# Use a point size of 0,01
# Plot the diagram
plt.figure(figsize=(10, 6))
# Calcular el valor de transparencia alpha
alpha = abs(data['radial_velocity']) / data['radial_velocity'].abs().max()
# Estrellas con velocidad radial positiva (rojo)
plt.scatter(data['l'], data['b'], s=0.01, color='red', alpha=alpha)
plt.xlabel('Galactic Longitude (l)')
plt.ylabel('Galactic Latitude (b)')
plt.title('Galactic Coordinates of Selected Stars with Radial Velocity (Tran
plt.gca().set_aspect('equal', adjustable='box')
plt.grid(True)
plt.show()

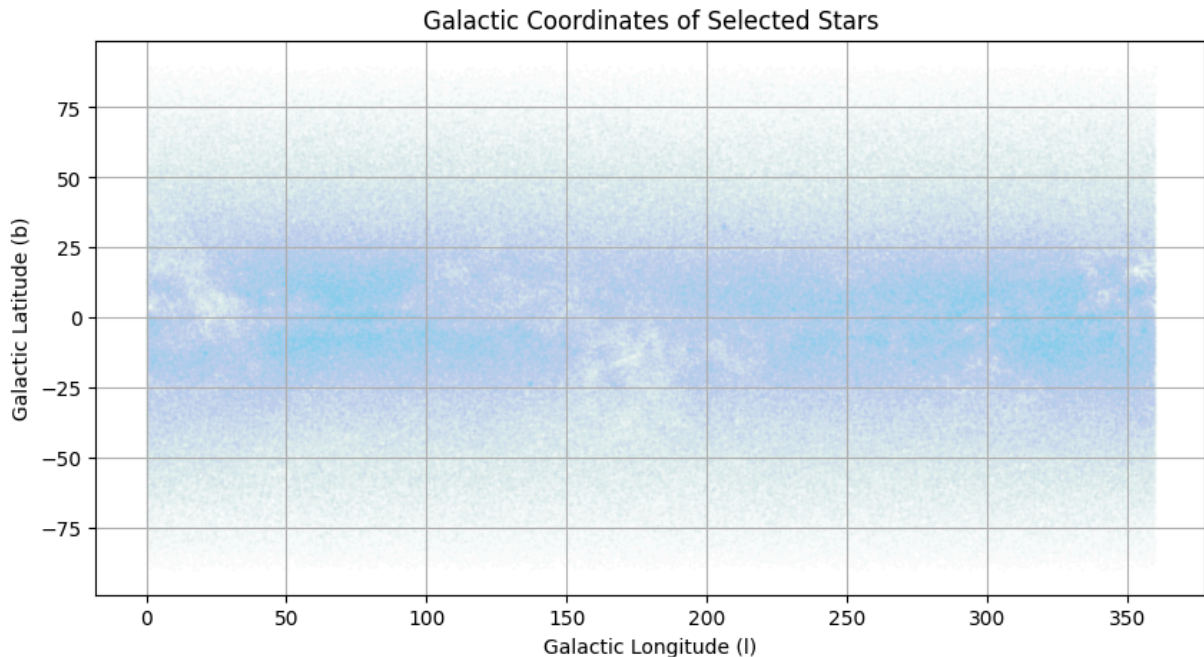
# Apartado 6
# comment the observed patterns in the previous diagrams and the possible ca
print("Los diagramas muestran la distribución de estrellas en la Vía Láctea
print("Las estructuras observadas en los diagramas pueden ser causadas por l
print("En resumen, los diagramas de coordenadas galácticas y velocidad radia

# Apartado 7
# Calculate the mean value and the estandar desviation of the radial veloci
# Calculate the mean and standard deviation of radial velocity for 5 bins on
data['l_bin'] = pd.cut(data['l'], bins=5, labels=False)
mean_radial_velocity = data.groupby('l_bin')['radial_velocity'].mean()
std_radial_velocity = data.groupby('l_bin')['radial_velocity'].std()
# Calculate the bin centers
bin_centers = data.groupby('l_bin')['l'].mean()
# Plot the mean radial velocity with error bars
plt.figure(figsize=(10, 6))
plt.errorbar(bin_centers, mean_radial_velocity, yerr=std_radial_velocity, fr
plt.xlabel('Galactic Longitude (l)')
plt.ylabel('Mean Radial Velocity')
plt.title('Mean Radial Velocity vs. Galactic Longitude')
plt.grid(True)
plt.show()

```

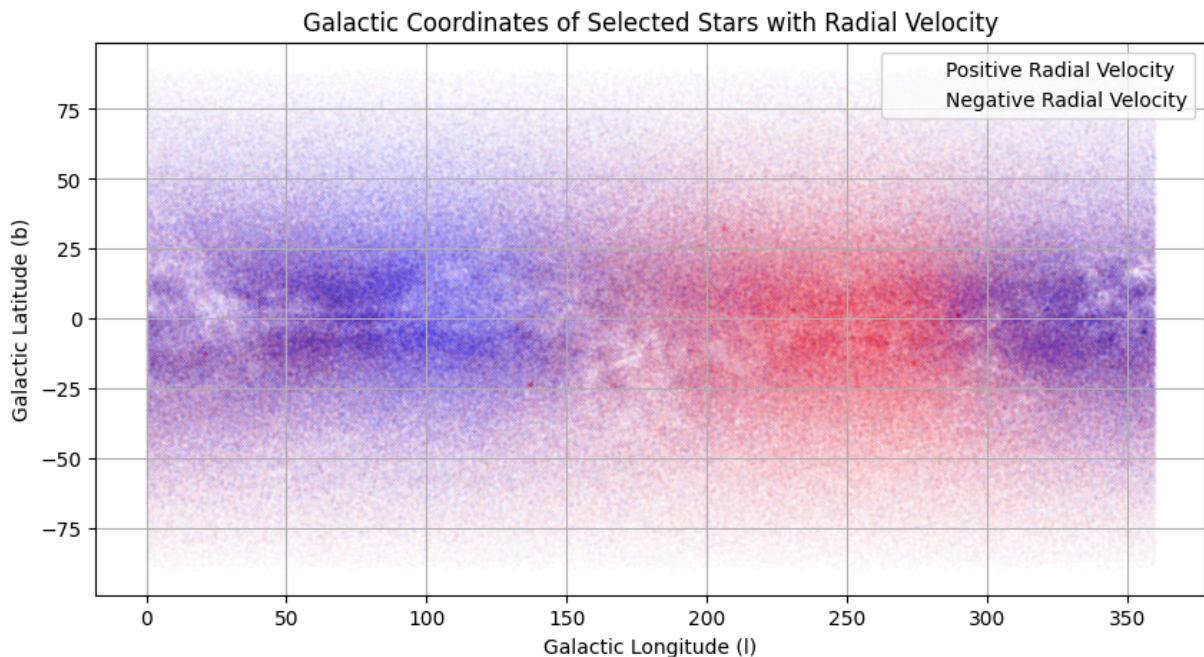


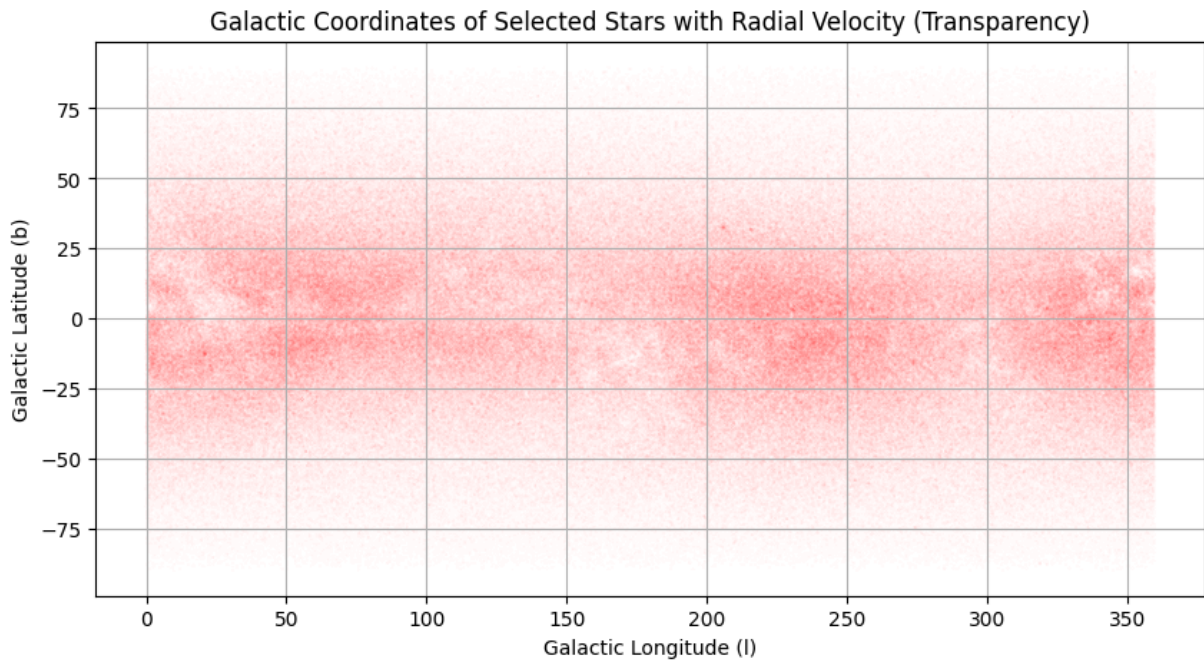
La distribución de distancias muestra un pico alrededor de 1000 parsecs, lo que sugiere que hay una mayor concentración de estrellas a esa distancia. Esto podría deberse a la presencia de un brazo espiral de la Vía Láctea a esa distancia, lo que resulta en una mayor densidad de estrellas en esa región. Además, la distribución de distancias también muestra una cola larga hacia distancias mayores, lo que indica la presencia de estrellas más distantes en la muestra. Esto podría deberse a la presencia de estrellas en otros brazos espirales de la Vía Láctea o incluso a estrellas en otras galaxias cercanas.



C:\Users\recio\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n2kfra8p0\LocalCache\local-packages\Python311\site-packages\IPython\core\pylabtools.py:152: UserWarning: Creating legend with loc="best" can be slow with large amounts of data.

fig.canvas.print_figure(bytes_io, **kw)





Los diagramas muestran la distribución de estrellas en la Vía Láctea en función de sus coordenadas galácticas y su velocidad radial. En el primer diagrama, se observa una distribución de estrellas en función de su latitud galáctica, lo que puede revelar la estructura tridimensional de la Vía Láctea. En el segundo diagrama, se distinguen las estrellas con velocidades radiales positivas y negativas, lo que puede indicar la presencia de corrientes estelares o movimientos colectivos de estrellas en la galaxia. En el tercer diagrama, la transparencia de las estrellas se relaciona con su velocidad radial, lo que puede revelar patrones de movimiento o interacciones gravitacionales en la Vía Láctea.

Las estructuras observadas en los diagramas pueden ser causadas por la dinámica de la Vía Láctea, incluyendo la presencia de brazos espirales, corrientes estelares, interacciones gravitacionales, y la rotación diferencial de la galaxia. Estos patrones pueden proporcionar información sobre la formación y evolución de la Vía Láctea, así como sobre la distribución y movimiento de las estrellas en nuestra galaxia.

En resumen, los diagramas de coordenadas galácticas y velocidad radial de las estrellas seleccionadas revelan información valiosa sobre la estructura y dinámica de la Vía Láctea, así como sobre los procesos astrofísicos que dan forma a nuestra galaxia.

