Astronomy Lab: Report 5

Kiavash Teymoori 99100585
Alireza Talebi 400109821

Introduction

In this experiment our goal is to determine Hubble's constant, which describes the rate of expansion of the universe.

Experiment

Cepheid Luminosity

Cepheid variable stars are a type of star whose luminosity varies with time. Their periodic changes in brightness follow the relation:

$$M_v = alog(P[days]) + B$$

where M_v is the absolute magnitude, P is the pulsation period and a and B are constants.
in our dataset we have m_v values, which are apparent magnitude. we can calculate absolute magnitude as follows:

$$M = m - 5log(\frac{d}{10}))$$

in which d is distance in parsec.

```
d_ly = 2.4e6
d = d_ly / 3.26 # distance in parsecs
M = m - 5 * np.log10(d/10)
print(M)
```

m_v	M_v
21.54	-2.7949
20.45	-3.8849
21.25	-3.0849
20.35	-3.9849
22.37	-1.9649
21.35	-2.9849
22.09	-2.2449
20.36	-3.9749
20.98	-3.3549
22.17	-2.1649
20.17	-4.1649
22.30	-2.0349
20.34	-3.9949
20.33	-4.0049
21.18	-3.1549
22.54	-1.7949

Table 1: apparant and absolute magnitudes respectively

Now that we obtained absolute magnitude, we can fit a linear equation to log(p) and M to find a and B constants and errors.

```
def cepheids_fit(x, a, b):
    return a*np.log10(x)+b
p_sorted = np.sort(p)
log_p = np.log10(p)
```

```
log_p_sorted = np.log10(p_sorted)
parameters, covariance = curve_fit(cepheids_fit, p, M, sigma=1/delta_m**2)
print(parameters)
errors = np.sqrt(linalg.eigvals(covariance))
print(errors)
```

Which results in:

$$a = -3.8062 \pm 0.0034$$
$$B = 1.46075 \pm 0.0045$$

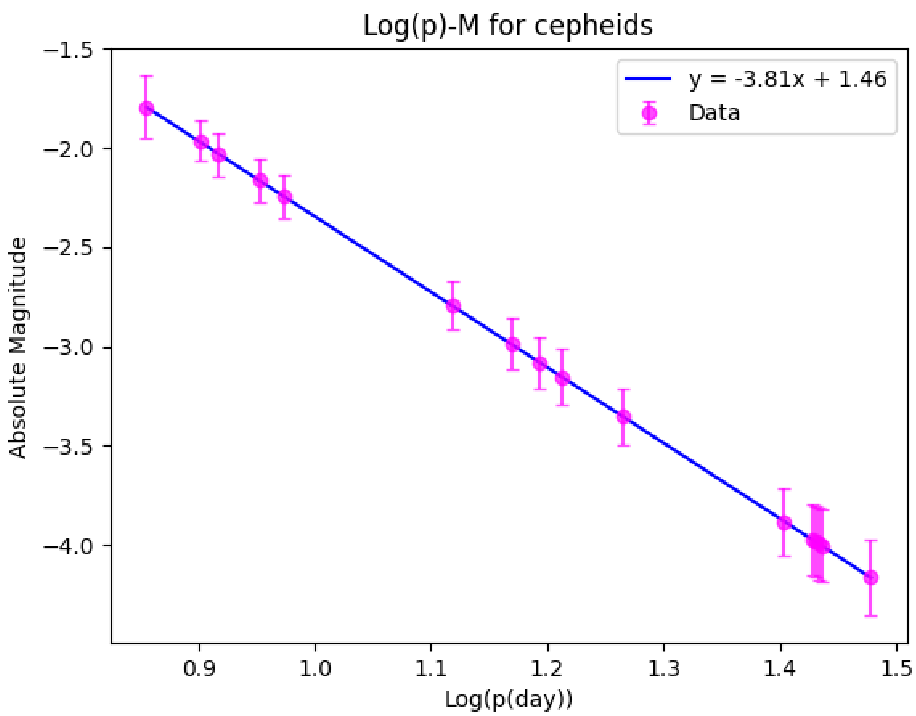

Figure 1: Log(p[days]) vs absolute magnitude for cepheids

Cepheid Distance Measurement

Now we assume that cepheid variable is the same in every galaxy, so now that we have calibrated our magnitude-time relation, we can find the Magnitude of a distant cepheid and then calculate it's distance.

```
M2 = parameters[0]*np.log10(p2)+parameters[1]
print(M2)
```

```
d2 = 10 * np.power(10, (m2-M2)/5) * 1e-6
print(d2)
```

Hubble's constant

$$V = Hr$$

We have V in our dataset and found r in the previous section. That means we can determine Hubble's constant.

```
# Hubble equation fitting

def hubble_fit(r, H):
    return H*r

d2_sorted = np.sort(d2)
H, H_covariance = curve_fit(hubble_fit, d2, v2, sigma=delta_v2)
H0 = H[0]
print(H0)
H0_error = np.sqrt(H_covariance)
print(H0_error)
```

and for the errors:

```
log_p2 = np.log10(p2)
delta_M2 = np.sqrt((log_p2 ** 2) * (errors[0] ** 2) + (errors[1] ** 2))
delta_d2 = (d2 * np.log(10) / 5) * np.sqrt(delta_m2**2 + delta_M2**2)
```

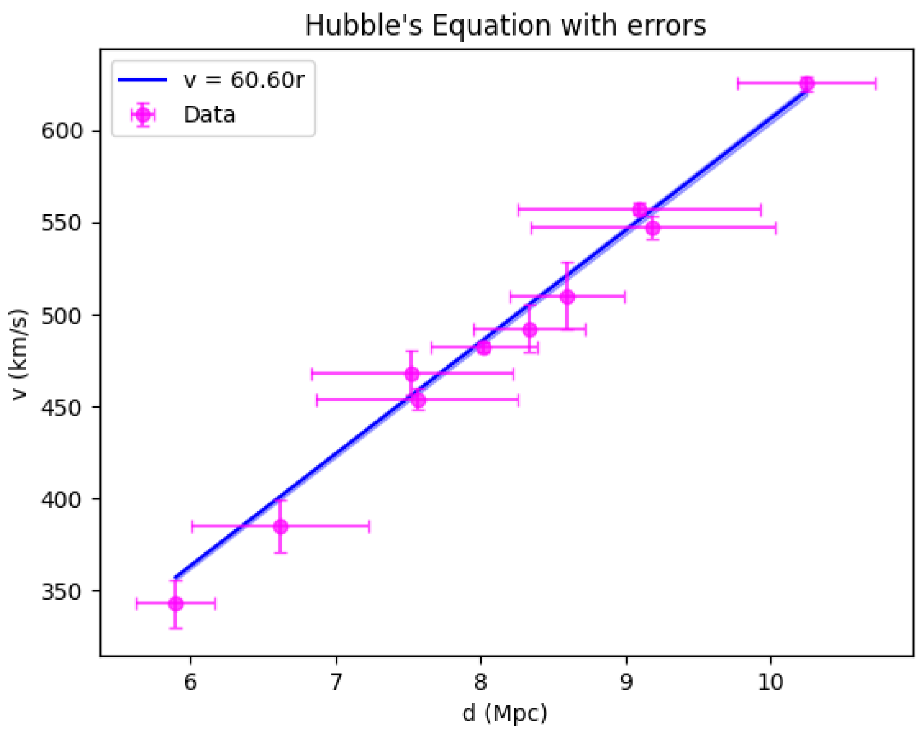

Figure 2: Hubble's equation fitting

we found that $H = 60.60 \pm 0.24 km/s/Mpc$ which is close enough to the actual 71km/s/Mpc.