Developments of the Applications and Nuances of Hubble's Constant

Authors: Ember Boicey, Everett Helm

Abstract -

Given observational data a line of best fit can be acquired to estimate Hubble's constant. This line of best fit returns a linear line that indicates constant acceleration of the galaxy, but more importantly can be used to calculate backwards in time. This constant can then be used alongside our current observations to calculate the age of the universe by reversing the velocity giving a glimpse into the size of the galaxy throughout time. This in term results in our best approximation of the age of the universe as we know it today.

1. Introduction

There is great curiosity in understanding the true scale of our universe, one of the great minds in astronomy, Edwin Powell Hubble is remembered by his "constant". Hubble's constant is a way to calculate the rate at which the galaxy is changing. This constant can be derived from measurements in the velocity of distant stars in comparison to a single reference point (Earth). This can be done by using the unique properties of Cepheids to reference different distances from Earth. The calculated distances of Cepheids can be found by using a phenomenon similar to the doppler effect called blue shift/red shift.

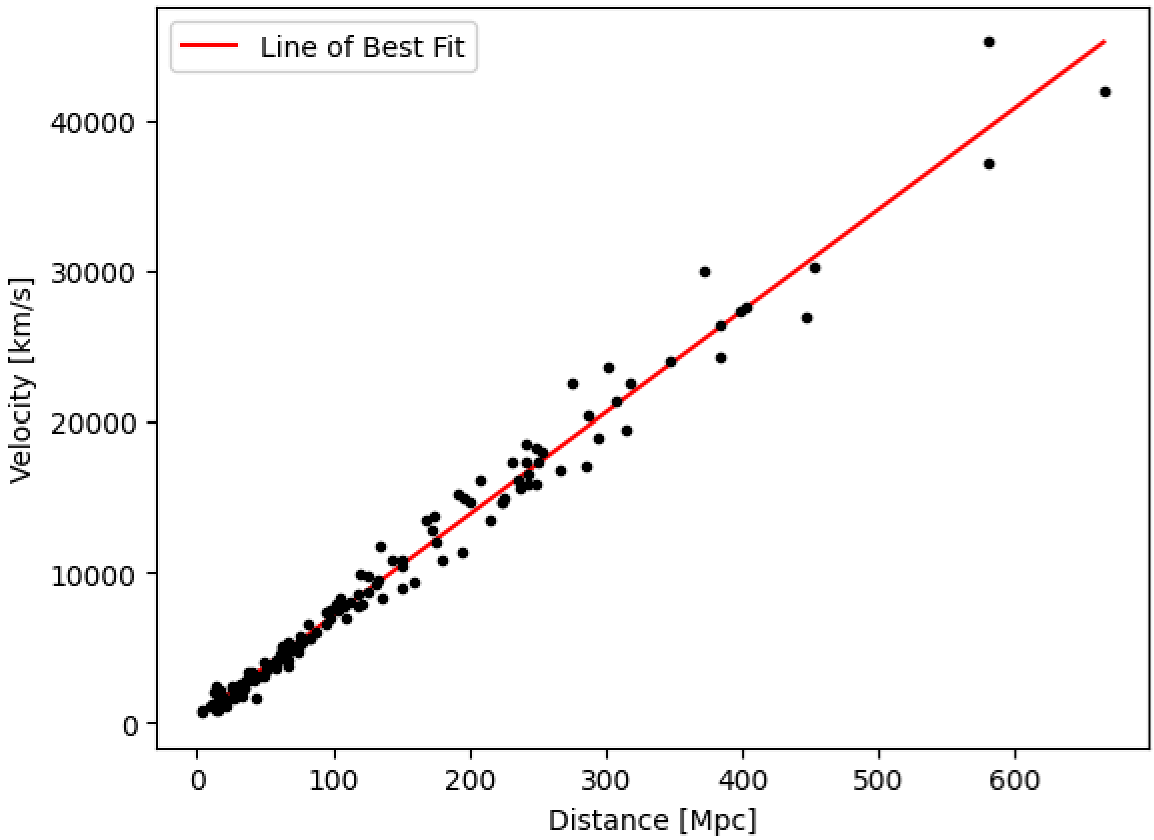
2. Process

By comparing the results gathered, trends may start to appear to show a constant (Hubble's constant) that will determine the change in size of the universe. Plotting observed data comparing astral bodies velocities and distance from the earth indicate a non-constant rate. This requires us to determine a line of best fit for the data, we did this as follows ([1] & [2]):

```
y = velocity.to(u.km / u.s).value[ind] # Referencing new y-axis window of the data
x = distance.to(u.mpc).value[ind] # Referencing new x-axis window of the data
velocity_error = dat["col7"] * u.km / u.s
yerr = velocity_error.to(u.km / u.s).value[ind]

z = np.polyfit(x, y, 1) # Slope
fit_fn = np.poly1d(z) # Fitting the slope into the data
x_fit = np.linspace(np.min(x), np.max(x), 100) # Calculating line of best fit based
off the x-axis points
y_fit = fit_fn(x_fit) # Finishing the line of best fit calculation by refrencing the
y-axis based off the x-axis
```

```
plt.errorbar(x, y, yerr=yerr, marker=".", color="black", linestyle="none") # Plotting
the data
plt.plot(x_fit, y_fit, color="red", label="Line of Best Fit") # plotting the line of
best fit
```



We determined that the best model to represent our data was a linear fit that has a slope of 67.48 km/s/Mpc.

---

3. Results

The slope found is the most current approximation of Hubble's constant. This constant turns out to not be constant at all, but instead a linear equation relating to distance. These results also show that the universe is indeed growing outward at an increasing rate. This data would be of great evidence to the debate about constant or increasing velocity of the galaxy.

---

4. Discussion

This data is very insightful to how the galaxy has grown over the course of its lifespan. These results should not be taken as pure fact however, further discoveries need to be made to strengthen the current approximations. We are limited by the technology of today to create an accurate estimation of Hubble's constant. As seen through the course of recent history the constant as we know it today is drastically changed from when Hubble initially proposed and published his (and many unrecognized colleagues) work.

---

5. Conclusions

Though these findings are ever closer to discovering the well kept secrets of the universe we must continue our development of observational equipment to gather more accurate information. It will only be through more precise measurements that we can reduce our margins of error to calculate the true age and span of the universe.

References

[1]- *Project 2 code,* Section [7-10], Ember Boicey, 10/8/2024
[2]- *Project 2 code*, Graph 3, Ember Boicey, 10/8/2024

AI Statement

Our group did have assistance of generative AI through Gemini AI, this was used in the coding aspect of the project to help auto fill lines while coding. There was no notable or heavy reliance on AI to create any specific code, and such no inquiries or prompts were used. AI was used to look up certain syntax issues and to explain bugs in the code when debugging.

Work Statement

Work was evenly distributed with both members working on all aspects of the project. To cater to each group members strengths each group member specialized as we saw fit, this distribution is as follows:

Everett was the lead in the written report with a majority of the written components and formatting. It was also his responsibility to proofread and correct the slides for the oral presentation.

Ember was the lead on the oral report slides and presentation preparations. In addition most code referenced is Ember’s code for this project. It was also her responsibility to proofread and correct the written report.

Both group members helped each other on their respective part of the project to ensure the best product we could produce.