Learnings from Cosmological Data Analysis Project

1. Technical Tools You Learned

- Python for scientific computing
- Libraries: NumPy, SciPy, Pandas, Matplotlib, Astropy
- Jupyter Notebook workflows
- Curve fitting using scipy.optimize.curve_fit
- Data cleaning, CSV handling, and visualization
- Residual analysis and uncertainty propagation

2. Cosmological Concepts You Applied

- Redshift and distance modulus relationship
- Flat LambdaCDM model assumptions
- Luminosity distance calculation with integration
- Estimating Hubble constant (H0) and matter density (Omega_m)
- Interpreting residuals and their significance
- Understanding the Hubble tension

3. Self-Learning and Problem Solving Skills

- Debugging broken code and Jupyter cell imports
- Converting a raw dataset from scratch (txt ? CSV)
- Interpreting warnings and errors (e.g., quad, curve_fit issues)
- Restarting kernel and knowing when to re-import
- Balancing perfectionism and pragmatism in research

4. How This Translates to Real Skills

- You can now process real astronomy datasets
- You understand basic cosmological modeling
- You?ve reproduced results in a real academic debate (Hubble tension)
- You?ve got a project that?s GitHub/LinkedIn/portfolio ready

Learnings from Cosmological Data Analysis Project

5. Where to Go Next (Expansion Ideas)

- Implement the same model with Planck CMB data and compare
- Try fitting a non-flat model or evolving dark energy (w ? -1)
- Learn to read FITS files (for JWST/SN spectra)
- Use MCMC (e.g., emcee or PyMC3) for parameter inference
- Publish a simple blog/medium article summarizing what you did
- Add GUI or web interface using Flask to explore supernova plots interactively

6. Soft Skills Developed

- Scientific writing and documentation
- Report structuring and presentation formatting
- Managing frustration and pushing through learning walls
- Asking deeper ?why? questions instead of just ?how?