Title: Code the Cosmos

Team Members: Devansh Verma, Pranav Sutar, Harsh Asati from IITT

Link to Github Page

Introduction

This document presents the analysis and results of the problem statement provided. The problem involves analysing the expansion of the universe using the Hubble parameter at various redshifts.

Overview

Given the dataset for the Hubble parameter at various redshifts, the task is to constrain and find the best fit values for the current density parameters of matter (non-relativistic), curvature, and dark energy, denoted as m, k, and respectively and plot 1D and 2D marginalized probability distributions for all three parameters and report uncertainty corresponding to 1, 2, and 3 sigma.

Methodology

- We Simplify the equation in this format:

```
# H(z) = H0 [ Omega_m*(1 + z)**3 + k*(1 + z)**2 + Omega_A ]**0.5
# H0 =  73.04 +/- 1.04 km s-1Mpc-1

# Thus, we can write the equation as:
#  Omega_m*(1 + z)**3 + k*(1 + z)**2 + Omega_A = (H(z)/H0)**2
## Sq( H(z)/H0 ) = A*Omega_m + B*Omega_k + Omega_l
## Sq( H(z)/H0 ) = a*m + b*k + l


# a = (1+z)**3
# b = (1+z)**2
# l = Omega_l
# m = Omega_m
# k = Omega_k
```

- Now, the task is to find optimal values m,k and l given z and various boundary conditions for the RHS.
- We add certain columns in the table to define different boundary levels for the value of Sq(H(z)/H_0) with different strictness as follows:

```
# We added a few columns to the dataframe for defining various error bounds
# Here, hz0 is the central value of Sq(Hz/H0)
# Hz1 and Hz-1 are loosely the upper and lower bounds of the error
# Hz2 and Hz-2 are the upper and lower bounds of the error (hz+e)/(h0-e0) and
(hz-e)/(h0+e0)
# Hz1.5 and Hz-1.5 are the average of the two bounds
# Hzu and Hzl are defined 1.5 times the difference between the central value and
the upper and lower bounds
# i.e. Hzu = hz0 + 1.5*(hz2 - hz0) and Hzl = hz0 - 1.5*(hz0 - hz-2)
```

- Now, we defined various losses:

```
# We defined the loss functions for the four different tolerances

# loss_0: The sum of the absolute differences between the model and the central value
of the tolerance
# loss_1: The sum of the differences prediced values and hz1 and hz-1 if the predicted
value is outside the bounds
# loss_1_5, loss_2, loss_ul: Similar to loss_1 but for hz1.5, hz2, and hzu, hzl
respectively
```

- We used inbuilt gradient descent to calculate the argmins for 4 loss functions (loss_0 to loss_2)
- We evaluated them with multiple parameters, but finally by loss_ul
- The loss_ul was 0 for two approaches

Approach 1:

- To sort the rows in ascending order on the error term, and choose the first 3 rows.
- Solve for the values of $\Omega_m,\ \Omega_k\ and\ \Omega_\Lambda$
- Use these values as initial values to run the gradient descent algorithm to minimize loss_1_5
- This will give us optimal values for $\Omega_m,\ \Omega_k\ and\ \Omega_\Lambda$

Approach 2:

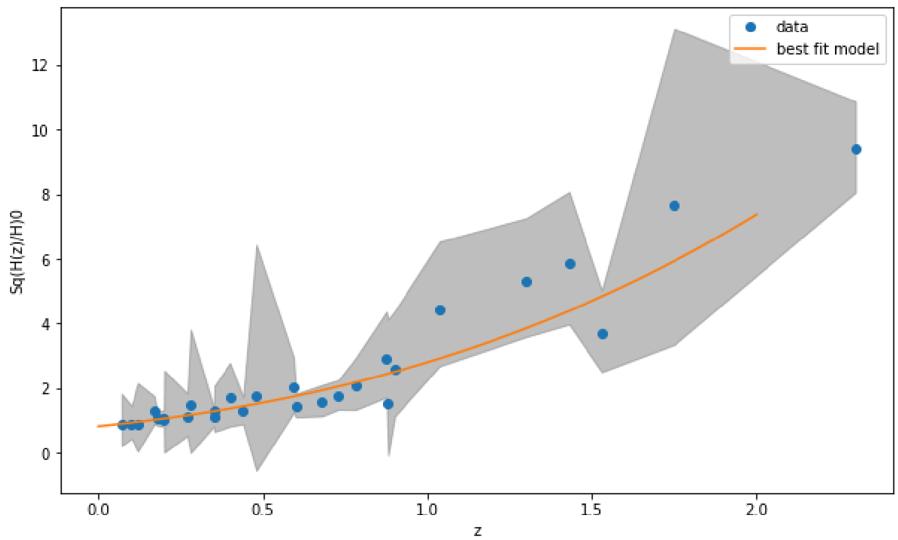
- To use the inbuilt function - curve_fit to calculate the initial values for fitting.
- Now, using the inbuilt minimize function, minimize the loss_0 of the function.

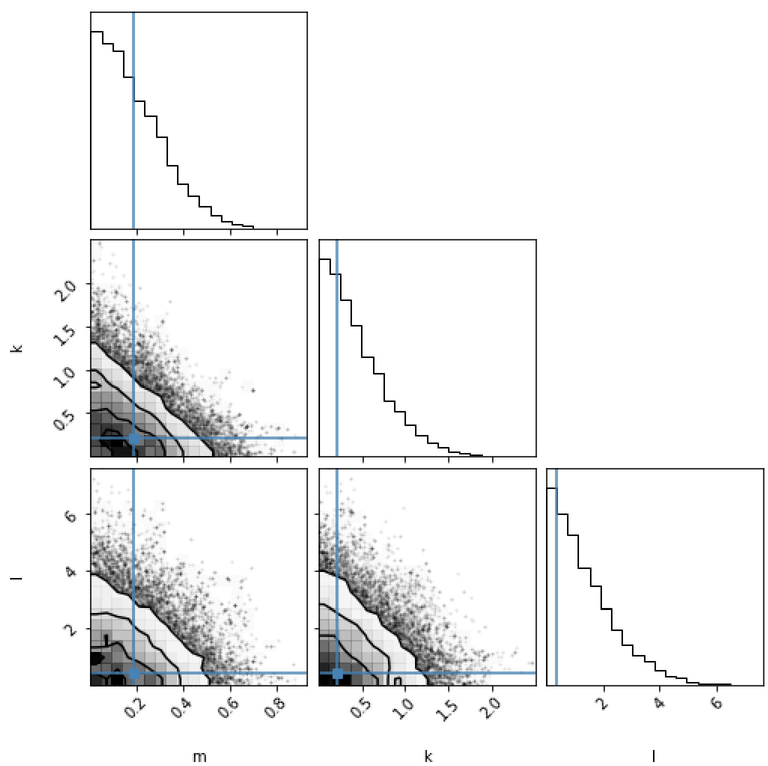
Results

(Here, you would present the results of your analysis. This could include the best fit values for m, k, and , the plots of the 1D and 2D marginalized probability distributions, and the reported uncertainties.)

The best first values obtained:

$$\Omega_m = 0.17394948, \ \Omega_k = 0.25353387, \ \Omega_\Lambda = 0.39313364$$





```
1 Sigma, 2 Sigma, 3 Sigma for m: (0.1591006813666565, 0.16722484921377861, 0.111612923650955)
1 Sigma, 2 Sigma, 3 Sigma for k: (0.36149236486967407, 0.4320008254026708, 0.2555813361798478)
1 Sigma, 2 Sigma, 3 Sigma for l: (1.0913138655732921, 1.3576793070442732, 0.7840317374427945)
```

	m	k	l
Uncertainty			
1 Sigma	0.148621	0.382337	1.101405
2 Sigma	0.178367	0.462658	1.308043
3 Sigma	0.104605	0.272893	0.788947

Values for runner-up model:

```
m, k, and l:[ 0.0158797   0.757422   -0.15538738]
```

	m	k	l
Uncertainty			
1 Sigma	0.158073	0.363483	1.098613
2 Sigma	0.174113	0.473571	1.457422
3 Sigma	0.109589	0.266253	0.793896

Conclusion

The inbuilt curve fitting function works the best initialization, and loss_0 is the best function to minimize with this minimization.

The values obtained henceforth are

$\Omega_m = 0.17394948,\ \Omega_k = 0.25353387,\ \Omega_\Lambda = 0.39313364$