

A Cosmology Calculator

July 31, 2015

The propose of the program is to provide some of the more commonly used cosmological parameters, as a function of redshift. The parameters provided are the comoving distance, r , the luminosity distance, D_L , the angular diameter distance D_A and the age of the universe at redshift z , t_{age} .

If you're in astro, this will be a program you will almost certainly use at some point. If you're not in astro, it will at least be a simple program for you to use to learn python.

These notes are in no way intended to give you a full understanding of the parameters and their meaning, or derivation. They give a general introduction to the theory for those interested, and provide the maths you need to write a cosmology calculator. If you are not interested in the theory, it is entirely possible to just use the maths in boxes in these notes to write the code, without even understanding it. If you are interested however you can read the notes to understand what you are computing. For a more thorough run down of the cosmology, and a full understanding of the derivation and significance of these parameters, I recommend Stuart Wyithe and Bart Pindors Cosmology Notes for the Masters Cosmology Course, in particular chapters 2 though to 10.

I will explain each of the parameters in question below, but first I need to introduce the Friedman Equation and some other things.

The metric used to describe the universe is called the Robertson-Walker Metric. An description of the features of this metric, and an explanation of why it is used is not included here. See the aforementioned Masters Cosmology notes for more information. The Friedman Model is a solution to the Robertson-Walker Metric, which can be derived from Einsteins Equations.

From the Friedman model one can obtain an expression for the time rate of change of the *scale factor* , a

$$H = \frac{\dot{a}}{a}. \quad (1)$$

The scale factor describes how big space is at time t . Computing the derivative

in this equation yields

$$H(z)^2 = H_0^2[\Omega_m a^{-3} + \Omega_R a^{-4} + \Omega_\Lambda], \quad (2)$$

where Ω_m is the density of mass in the universe, Ω_R is the density of radiation in the universe, and Ω_Λ is the dark energy density. This equation is called the Friedman Equation. Note that here and throughout this document Ω_R is explicitly included for completeness, however in the calculator code (and in general in Astrophysics), it is assumed to be zero for all redshifts below $z \sim 1000$. This is because it goes as a^{-4} and a goes to infinity at low redshifts, so the contribution from this term becomes negligible at smaller redshifts. The scale factor can be written in terms of the redshift, since light is being redshifted in space by the size of space changing

$$a = \frac{1}{(1+z)}. \quad (3)$$

The Friedman equation gives the evolution of the Hubble parameter with redshift for a given *cosmology* (here cosmology refers to the choice of Ω_m , Ω_R and Ω_Λ). The Hubble *constant*, which you may be familiar with, doesn't take into account the evolution of the universe. It gives the expansion of the universe *today* at redshift 0. The way the expansion of the universe evolves with redshift is encapsulated by the terms in the square brackets in the Friedman equation above.

Note that redshift doesn't get larger with time, it gets larger the further *back* in time you go, and is zero now. In other words it is defined to be zero at the present, and to be infinite at the big bang. This makes sense when you think about what redshift means. Redshift gives the fractional shift in the wavelength of light emitted at some time in the past, relative to how we see it now. So it makes sense that something emitted now will have a redshift of 0 because it hasn't had its wavelength shifted at all. The further back you look (which is equivalent to saying the further away you look), the more the wavelength of the light has been stretched (i.e. redshifted) due to the expansion of the universe.

A characteristic scale in cosmology is the *Hubble distance*, d_H . This is the size of the observable universe at a given redshift,

$$d_H = \frac{c}{H(z)}. \quad (4)$$

Another way of thinking of this is that the Hubble distance is the distance between the Earth and the galaxies which are currently receding from us at the speed of light (i.e. things which are just now becoming invisible to us because their light can never reach us).

We now have what we need to get the comoving distance.

1 Comoving distance (or proper distance)

This is the distance to an object in *comoving* coordinates. That is, this distance takes the expansion of the universe into account.

The comoving distance is obtained by integrating the Hubble distance from the redshift in question to now (so from redshift z to redshift 0)

$$r_{\text{com}} = \int_0^z d_H dz, \quad (5)$$

or expanding

$$r_{\text{com}} = \left(\frac{c}{H_0} \right) \int_0^z \frac{1}{[\Omega_m(1+z)^3 + \Omega_R(1+z)^4 + \Omega_\Lambda]} dz \quad (6)$$

A diagram of the comoving distance is shown in the first panel of Figure 1.

2 Luminosity distance

As you no doubt know, flux falls off as radius squared. If you know the intrinsic brightness of something, this can be used measure the distance to it. In an expanding universe however, the amount of space for the light to propagate through is increasing, and so its no longer as simple as radius squared.

The luminosity of an object observed at Earth, in terms of its emitted flux is

$$L_{\text{obs}} = L_{\text{em}} a^2 \quad (7)$$

or

$$L_{\text{obs}} = \frac{L_{\text{em}}}{(1+z)^2}, \quad (8)$$

where one factor of a is due to time dilation (photons lose energy), and one is due to red shift (and they arrive at a lower rate).

The distance one would measure in a static universe is

$$d_L = \sqrt{\frac{L_{\text{em}}}{4\pi F_{\text{obs}}}}, \quad (9)$$

and

$$F_{\text{obs}} = \frac{L_{\text{obs}}}{4\pi r^2}. \quad (10)$$

Combining these, one can write the luminosity distance in terms of redshift

$$D_L = \frac{r_{\text{com}}}{a} = r_{\text{com}}(1+z) \quad (11)$$

So you can see that once you have the comoving distance, getting the luminosity distance is trivial!

A diagram of the luminosity distance is shown in the second panel of Figure 1.

3 Angular diameter distance

This is the distance according to the angular size of an object on the sky, compared to some known size.

When you look at a 30 cm ruler at a distance, it has a certain angular size, that is, it subtends some angle in your vision. For very distant objects which have a small angular size (like a galaxy in the night sky), you can use the *small angle approximation* to relate the angular size of the object to the true size and the distance to the object

$$d \approx R\theta, \tag{12}$$

where θ is the angular size of an object, d is the true size of the object and R is the distance to the object. If d and θ are known, then the distance to an object can be obtained using this relation.

This is all assuming that the lines from the edges of the object to the observer are straight, so that you can use trigonometry in the normal way. But the way we determine the location of the edges of an object is from the light coming from it, and in an expanding universe these are no longer straight.

If we consider two points separated by some apparent angular separation θ , at a distance of R , the light from these two points will travel along curved null geodesics to reach us, and so the angle of incidence of the light will not be the same as the angle of emission. This is like the way lenses can create apparent images, because they alter the angle of incidence of the light rays from a source on our eyes.

To account for this, we need to re-define the edges of the small angle triangle, and take the expansion of the universe into account. The apparent distance with expansion will be the comoving distance, multiplied by a , the scale factor.

$$D_A = r_{\text{com}}a = \frac{r_{\text{com}}}{1+z} \tag{13}$$

This is the angular diameter distance, and as with the luminosity distance, getting it is trivial once you have the comoving distance.

A diagram of the angular diameter distance is shown in the third panel of Figure 1.

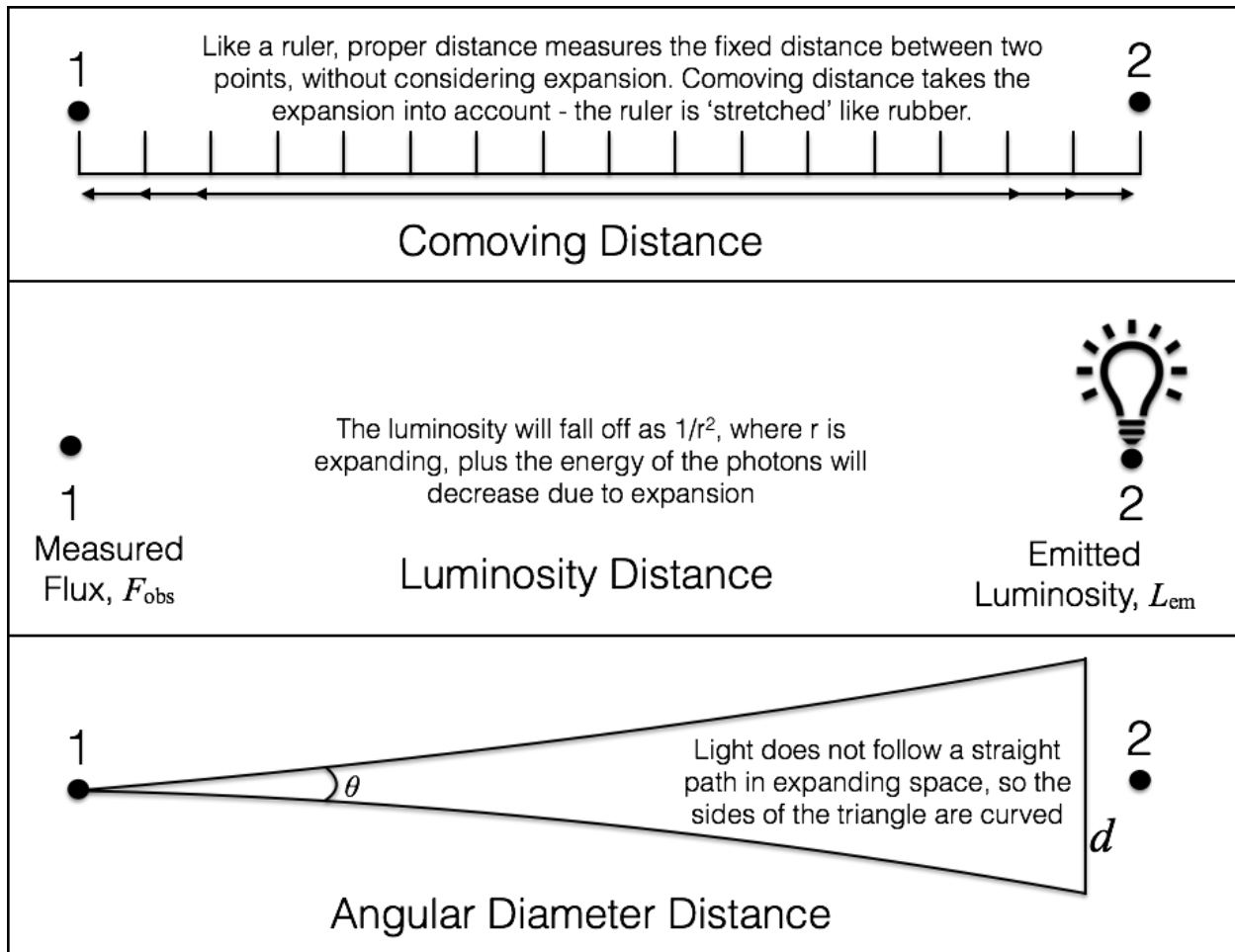


Figure 1: The three types of distances: Comoving distance, luminosity distance and angular diameter distance.

4 Age of the universe at redshift z

The age of the universe at a given redshift can be obtained by integrating up all the time between the beginning of the universe and the time at z .

$$t_{\text{age}} = \int_0^a (t_{\text{age}}) \frac{dt}{da'} da'. \quad (14)$$

Now, as mentioned above, the Hubble parameter gives the time rate of change of the scale factor, a

$$H = \frac{\dot{a}}{a} = \frac{da}{dt} \frac{1}{a}. \quad (15)$$

Rearranging and substituting into the equation for t_{age} gives

$$t_{\text{age}} = \frac{1}{H_0} \int_{\infty}^{z_{\text{age}}} (1+z)^{-1} [\Omega_m(1+z)^3 + \Omega_R(1+z)^{-4} + \Omega_\Lambda]^{1/2} dz \quad (16)$$

5 The code

You are required to write a program which will perform the following:

1. Take as its input a redshift range, and optionally, values of H_0 , Ω_m , and/or Ω_Λ (Ω_R is always 0). To begin with you may want to just hard-code these values and change them in the code each time you want to run it for a different set of values. Once you have everything else working however you should change your code so these are taken as optional arguments in terminal (To do this you want to use the function `ARGPARSER` (Google it!)).
2. Use the redshift range to generate a list or array of redshift values. To do this you will either want to use a *list comprehension*, or a `NUMPY` array made with `NUMPY.RANGE` or `NUMPY.Linspace`. Again, Google these things to learn about them (including the list comprehension).
3. Compute the associated cosmological values for each redshift value. This will require performing two numerical integrals for each redshift value, one for the comoving distance, and one for the age of the universe at that redshift. To perform these integrals I recommend using either `SCIPY.INTEGRATE.QUAD` or `MPMATH.QUAD`. Google them to learn about each, and use the one you like the look of! You will probably need to install the `MPMATH` module to use that integrator. You almost certainly already have the `NUMPY` and `SCIPY` modules with your python installation.
4. Plot the results of your code for the range of z on a two-by-two set of axes. To do this you will use `MATPLOTLIB.PYplot` and its associated functions. The Matplotlib website has a extensive set of documentation which you can use to figure out how to do this. In particular, if you go to the Matplotlib Gallery (there's a link on the webpage), you can find examples

Table 1: Required fundamental constants for writing the code

Constant	Value	Units
Speed of light, c	2.99792458×10^5	km s^{-1}
km in 1 Mpc	$3.08567758147 \times 10^{19}$	km
Seconds in 1 year	3.1557600×10^7	s

similar to what you want to do, and copy/modify the source code of the example to suit your needs.

The above list is peppered with instructions to google things. This is because this is 90% of learning python. Any problem you have someone has had before, and they have asked the internet for help. So there's no point in reinventing the wheel - if you have a problem or want to know how to do something, Google it! The best places to look for help or learn how to use a function are the documentation on the module's website (Numpy, Scipy and Matplotlib all have excellent online documentation, as does the core python website), or StackOverflow, an online coding forum.

Achieving accurate values for your output requires very accurate input values of the known constants, most of which are hidden in the Hubble constant. In fact, the only units that appear in these calculations are in the Hubble constant, and the speed of light, c . The Hubble constant has units $\text{km s}^{-1}\text{Mpc}$, and you should use the speed of light in units of km s^{-1} . This will give you distances in units of Mpc. To get the correct value for the age of the universe in years, you need to convert the units of the Hubble constant to be yr^{-1} for use in equation (16). You can do this by dividing by the number of km in 1 Mpc and then dividing by the number of seconds in 1 year. Table 1 gives the values and units of all the constants required for the code.