

CLASS: Cosmological Boltzman Code

Resuelve el sistema de perturbaciones cosmológicas en un universo de FRW tipo LDCM (aunque incluye otras varias opciones), calcula el fondo cósmico de microondas y observables de formación de estructura a grandes escalas. Para ello:

- Resuelve las ecuaciones del background (factor de escala, distancias, densidades promedio)
- Ecuaciones de Boltzman en recombinación y reionización

<http://www.class-code.net>

Dos formas de obtener el código:

- 1) Descargar el archivo comprimido .tar.gz del sitio web y descomprimirlo en consola:

```
$ tar -zxvf class_public-vx.y.z.tar.gz
```

- 2) Usar git en la terminal:

```
$ git clone https://github.com/lesgourg/class_public.git class
```

INSTALAR Y CORRER CLASS (FORMA TRADICIONAL)

https://github.com/lesgourg/class_public/wiki/Installation

http://lesgourg.github.io/class_public/class_public-2.7.0/doc/manual/html/index.html

II- Compilar:

REQUERIMIENTOS LINUX: gcc y python (de preferencia anaconda o bien tener instalado pip3)

```
$ cd class/
```

```
class]$ make
```

Genera: ejecutable, libclass.a y un wrapper de python (classy).

O

```
class]$ make class
```

Genera: ejecutable no se crea libclass.a

Genera: ejecutable, libclass.a y un wrapper de python (classy).

III- Correr class:

```
Class]$ ./class explanatory.ini
```

Genera: outputs en archivos de texto

Writing output files in ...

Inifile: explanatory.ini

1) Hubble parameter : either 'H0' in km/s/Mpc or 'h' or '100*theta_s' where the latter is the peak scale parameter $100(ds_{dec}/da_{dec})$ close to 1.042143 (default: 'h' set to 0.67556)

#H0 = 67.556

h = 0.67556

#100*theta_s = 1.042143

2) photon density: either 'T_cmb' in K or 'Omega_g' or 'omega_g' (default: 'T_cmb' set to 2.7255)

T_cmb = 2.7255

#Omega_g =

#omega_g =

3) baryon density: either 'Omega_b' or 'omega_b' (default: 'omega_b' set to 0.022032)

#Omega_b =

omega_b = 0.022032

5) density of cdm (cold dark matter): 'Omega_cdm' or 'omega_cdm' (default: 'omega_cdm' set to 0.12038)

```
#Omega_cdm =  
omega_cdm = 0.12038
```

7) curvature: 'Omega_k' (default: 'Omega_k' set to 0)

```
Omega_k = 0.
```

8a) Dark energy contributions. At least one out of three conditions must be satisfied:

- i) 'Omega_Lambda' unspecified.
- ii) 'Omega_fld' unspecified.
- iii) 'Omega_scf' set to a negative value. [Will be referred to as unspecified in the following text.]

The code will then use the first unspecified component to satisfy the closure equation ($\sum_i \Omega_i$) equals (1 + Omega_k) (default: 'Omega_fld' and 'Omega_scf' set to 0 and 'Omega_Lambda' inferred by code)

```
Omega_Lambda = 0.7  
#Omega_fld = 0  
#Omega_scf = 0
```

OUTPUTS

```
lana@lana-Lenovo-YOGA-900-13ISK2:~/class/output$ ls
explanatory00_background.dat  explanatory00_cl_lensed.dat  explanatory00_unused_parameters
explanatory00_cl.dat          explanatory00_parameters.ini
lana@lana-Lenovo-YOGA-900-13ISK2:~/class/output$
```

Output\$ nano/explanatory00_background.dat

```
lana@lana-Lenovo-YOGA-900-13ISK2: ~/class/output
GNU nano 2.5.3                               File: explanatory00_background.dat

# Table of selected background quantities
# All densities are multiplied by (8piG/3) (below, shortcut notation (.) for this factor)
# Densities are in units [Mpc^-2] while all distances are in [Mpc].
#
# 1:z          2:proper time [Gyr]      3:conf. time [Mpc]      4:H [1/Mpc]      5:comov. dist.      6:ang.diam.dist.      7:lum. dist. $
1.000000000000e+14  7.558503632224e-26  4.634785000235e-09  2.157545946648e+22  1.416504541243e+04  1.416504541243e-10  1.416504541245
9.930486593843e+13  7.664693049753e-26  4.667526809499e-09  2.127654528182e+22  1.416504541243e+04  1.426420073032e-10  1.406657935695
9.861456399050e+13  7.772374322409e-26  4.700268618762e-09  2.098177236191e+22  1.416504541243e+04  1.436405013543e-10  1.396879777255
9.792906056653e+13  7.881568409264e-26  4.733010428026e-09  2.069108333218e+22  1.416504541243e+04  1.446459848638e-10  1.387169590125
9.724832231036e+13  7.992296563846e-26  4.765752237290e-09  2.040442161294e+22  1.416504541243e+04  1.456585067579e-10  1.377526901805
9.657231609767e+13  8.104580338272e-26  4.800313035958e-09  2.012173140839e+22  1.416504541243e+04  1.466781163052e-10  1.367951243105
9.590100903443e+13  8.218441587444e-26  4.833054845221e-09  1.984295769572e+22  1.416504541243e+04  1.477048631193e-10  1.358442148075
9.523436845525e+13  8.333902473306e-26  4.867615643889e-09  1.956804621446e+22  1.416504541243e+04  1.487387971611e-10  1.348999153995
9.457236192179e+13  8.450985469153e-26  4.900357453153e-09  1.929694345585e+22  1.416504541243e+04  1.497799687413e-10  1.339621801385
9.391495722125e+13  8.569713364009e-26  4.934918251820e-09  1.902959665248e+22  1.416504541243e+04  1.508284285225e-10  1.330309633945
9.326212236469e+13  8.690109267060e-26  4.969479050487e-09  1.876595376800e+22  1.416504541243e+04  1.518842275221e-10  1.321062198555
9.261382558559e+13  8.812196612153e-26  5.004039849155e-09  1.850596348697e+22  1.416504541243e+04  1.529474171148e-10  1.311879045235
9.197003533823e+13  8.935999162357e-26  5.040419637226e-09  1.824957520492e+22  1.416504541243e+04  1.540180490346e-10  1.302759727145
9.133072029615e+13  9.061541014589e-26  5.074980453893e-09  1.799673901845e+22  1.416504541243e+04  1.550961753778e-10  1.293703800545
9.069584935070e+13  9.188846604302e-26  5.109541234560e-09  1.774740571556e+22  1.416504541243e+04  1.561818486055e-10  1.284710824775
9.006539160943e+13  9.317940710246e-26  5.145921022631e-09  1.750152676602e+22  1.416504541243e+04  1.572751215457e-10  1.275780362235
9.843931639467e+13  9.448848459284e-26  5.182300810702e-09  1.725905431200e+22  1.416504541243e+04  1.583760473965e-10  1.266911978385
9.881759324197e+13  9.581595331288e-26  5.218680598773e-09  1.701994115866e+22  1.416504541243e+04  1.594846797283e-10  1.258105241695
9.820019189868e+13  9.716207164097e-26  5.255060386844e-09  1.678414076505e+22  1.416504541243e+04  1.606010724864e-10  1.249359723635
9.758708232243e+13  9.852710158545e-26  5.291440174915e-09  1.655160723501e+22  1.416504541243e+04  1.617252799938e-10  1.240674998645
9.697823467967e+13  9.991130883562e-26  5.329638952389e-09  1.632229530823e+22  1.416504541243e+04  1.628573569537e-10  1.232050644135
9.637361934426e+13  1.013149628135e-25  5.366018740460e-09  1.609616035145e+22  1.416504541243e+04  1.639973584524e-10  1.223486240445
9.577320689599e+13  1.027383367260e-25  5.404217517935e-09  1.587315834980e+22  1.416504541243e+04  1.651453399616e-10  1.214981370855
9.517696811915e+13  1.041817076187e-25  5.440597306006e-09  1.565324508981e+22  1.416504541243e+04  1.663013573413e-10  1.206535621505
9.458487400115e+13  1.056453564290e-25  5.478796083480e-09  1.543638019285e+22  1.416504541243e+04  1.674654668427e-10  1.198148581435
9.399689573103e+13  1.071295680415e-25  5.518813850358e-09  1.522251902310e+22  1.416504541243e+04  1.686377251106e-10  1.189819842535
9.341300469814e+13  1.086346313429e-25  5.557012627833e-09  1.501162076300e+22  1.416504541243e+04  1.698181891864e-10  1.181548999535
9.283317249071e+13  1.101608392786e-25  5.595211405307e-09  1.480364436334e+22  1.416504541243e+04  1.710069165107e-10  1.173335649985
9.225737089444e+13  1.117084888096e-25  5.635229172185e-09  1.459854934362e+22  1.416504541243e+04  1.722039649263e-10  1.165179394225
9.168557189121e+13  1.132778814703e-25  5.673427949660e-09  1.439629578415e+22  1.416504541243e+04  1.734093926807e-10  1.157079835385
9.114774765760e+13  1.148693224271e-25  5.713445716538e-09  1.419684431832e+22  1.416504541243e+04  1.746232584295e-10  1.149036579325
9.055387056366e+13  1.164831215379e-25  5.753463483416e-09  1.400015612492e+22  1.416504541243e+04  1.758456212385e-10  1.141049234685
9.999391317146e+13  1.181195929123e-25  5.793481250294e-09  1.380619292058e+22  1.416504541243e+04  1.770765405872e-10  1.133117412795
9.943784823382e+13  1.197790550732e-25  5.835318006575e-09  1.361491695232e+22  1.416504541243e+04  1.783160763713e-10  1.125240727695
9.888564869297e+13  1.214618310179e-25  5.875335773453e-09  1.342629099020e+22  1.416504541243e+04  1.795642889059e-10  1.117407876125
9.833728767921e+13  1.231682482819e-25  5.917172529735e-09  1.324027832009e+22  1.416504541243e+04  1.808212389282e-10  1.109651237465
9.779273850965e+13  1.248986390020e-25  5.957190296613e-09  1.305684273648e+22  1.416504541243e+04  1.820869876007e-10  1.101937673745
9.725197468684e+13  1.266533399813e-25  5.999027052894e-09  1.287594853551e+22  1.416504541243e+04  1.833615965139e-10  1.094277729635
9.671496989756e+13  1.284326927547e-25  6.040863809176e-09  1.269756050794e+22  1.416504541243e+04  1.846451276895e-10  1.086671032415
9.618196980114e+13  1.302370436552e-25  6.084519554861e-09  1.252164393233e+22  1.416504541243e+04  1.859376435834e-10  1.079117211925
9.565213307992e+13  1.320667438815e-25  6.126356311142e-09  1.2348154656831e+22  1.416504541243e+04  1.872392070884e-10  1.071615906025
9.512624933457e+13  1.339221495663e-25  6.170012056828e-09  1.217708864987e+22  1.416504541243e+04  1.885498815381e-10  1.064166733495

^G Get Help      ^O Write Out     ^W Where Is      ^K Cut Text      ^J Justify       ^C Cur Pos       ^V Prev Page     ^M- First Line   ^M- WhereIs Next ^M Mark Text
^X Exit          ^R Read File     ^_ Replace       ^U Uncut Text    ^T To Spell      ^_ Go To Line    ^V Next Page     ^M- Last Line    ^M- To Bracket   ^M Copy Text
```

INSTALAR Y CORRER CLASS EN PYTHON (FORMA “HUMAN-FRIENDLY”)

Dentro de python/ se encuentra el WRAPPER de PYTHON classy escrito en cython
El cual se puede instalar como librería de python en el sistema:

```
class]$ cd python/  
python]$ python setup.py install --user
```

Para checar que la instalación fue exitosa:

```
Python]$ python  
>>> from classy import Class
```

Si no aparece error classy esta bien instalada

TRES OPCIONES:

1. Usar python en terminal
2. Hacer un script.py y correr `$ python script.py`
3. **Usar un notebook de jupyter:**
con anaconda instalado y classy instalado:
correr dentro de la carpeta notebooks en indico `$ jupyter notebook`

notebooks/ - Google Chrome

[1605.06102] hi_cla x A Larger Estimate o x notebooks/ x

localhost:8888/tree/notebooks

Apps Electronic library Indico [Centro In CINVESTAV- Log Ultra-light Axion English to French Indico [Coloquio nbviewer Fitting data — S

jupyter

Select items to perform actions on them. Upload New ↺

	notebooks
..	
☐	cl_ST.ipynb
☐	cltt_terms.ipynb
☐	distances.ipynb
☐	many_times.ipynb
☐	neutrino hierarchy.ipynb
☐	one_k.ipynb
☐	one_time.ipynb
☐	Probadita.ipynb
☐	thermo.ipynb
☐	varying_neff.ipynb
☐	varying_pann.ipynb
☐	warmup.ipynb
☐	many_times.png
☐	one_k.pdf
☐	ratio-neff-cltt.pdf
☐	ratio-neff-Pk.pdf

lss.jpg ^ standard-mo...jpg ^ bigbang2.jpg ^ BBNabunda....png ^ bigbang.jpg ^ Show all ×



```
In [1]: %matplotlib notebook
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
from classy import Class
```

```
In [2]: font = {'size' : 20, 'family':'STIXGeneral'}
axislablefontsize='large'
matplotlib.rc('font', **font)
matplotlib.mathtext.rcParams['legend.fontsize']='medium'
```

```
In [3]: #Lambda CDM
LCDM = Class()
LCDM.set({'Omega_cdm':0.25,'Omega_b':0.05})
LCDM.compute()
```

```
In [9]: #Einstein-de Sitter
CDM = Class()
CDM.set({'Omega_cdm':0.95,'Omega_b':0.05})
CDM.compute()

# Just to cross-check that Omega_Lambda is negligible
# (but not exactly zero because we neglected radiation)
Omega_lambda = LCDM.get_current_derived_parameters(['Omega0_lambda'])
Omega_lambda
```

```
Out[9]: {'Omega0_lambda': 0.6999083286434546}
```

```
In [5]: #Get background quantities and recover their names:
baLCDM = LCDM.get_background()
baCDM = CDM.get_background()
```


In [6]: *#Get background quantities and recover their names:*

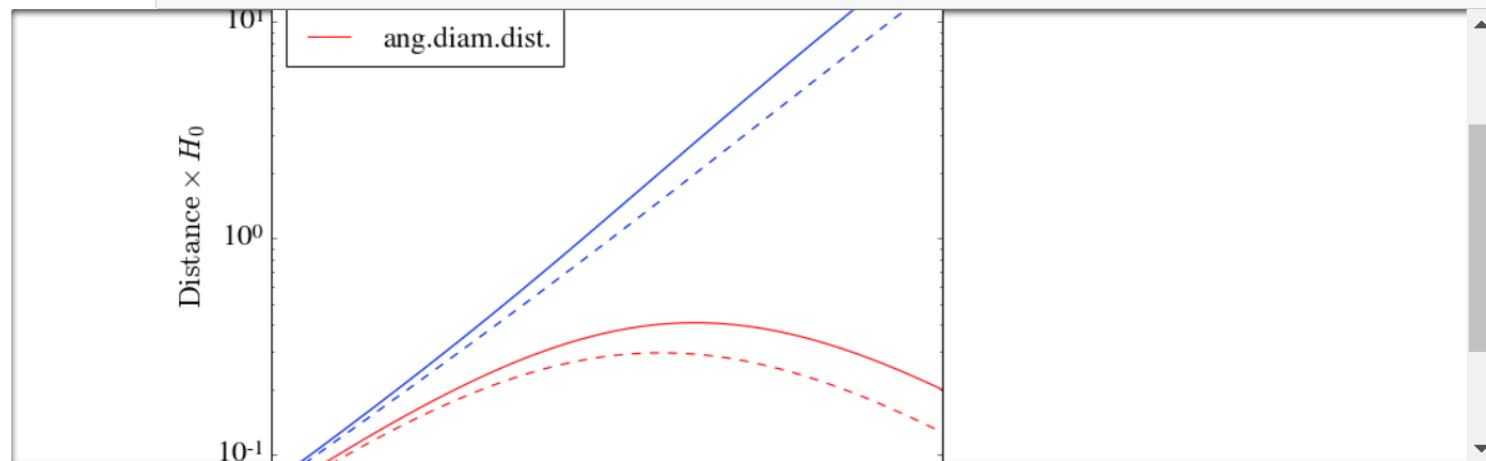
```
baLCDM = LCDM.get_background()
baCDM = CDM.get_background()
```

In [7]: *#Get H_0 in order to plot the distances in this unit*

```
fLCDM = LCDM.Hubble(0)
fCDM = CDM.Hubble(0)
```

```
In [12]: #namelist = ['lum. dist.', 'comov. dist.', 'ang.diam.dist.']
namelist = ['lum. dist.', 'ang.diam.dist.']
colours = ['b', 'r']
for name in namelist:
    idx = namelist.index(name)
    plt.loglog(baLCDM['z'], fLCDM*baLCDM[name], colours[idx]+'-')
plt.legend(namelist, loc='upper left')
for name in namelist:
    idx = namelist.index(name)
    plt.loglog(baCDM['z'], fCDM*baCDM[name], colours[idx]+'--')
plt.xlim([0.07, 10])
plt.ylim([0.08, 20])

plt.xlabel(r"$z$")
plt.ylabel(r"$\mathrm{Distance} \times H_0$")
```



In []: `plt.savefig('distances.pdf')`

USO BÁSICO

1. Importar el módulo Class
2. Crear un “caso” (instance)
3. Crear un diccionario con los parametros que uno quiere variar
4. Inicializar las estructuras con “compute”
5. Jugar con las funciones de Class!
Resolver el background, resolver perturbaciones, CMB, etc.
6. Limpiar las estructuras con “struct_cleanup”
7. Borrar parametros con “empty”
8. Empezar con un nuevo caso en 2.

DOS OPCIONES:

1. Crear un script de .py
2. Crear un notebook de jupyter

