

# Supernovas y la expansión acelerada del universo

Iván Villegas Pérez

Un trabajo presentado para la asignatura *Astrofísica (G69)*



Departamento de Física Moderna  
Universidad de Cantabria  
España

	Einstein-de Sitter	Concordancia $\Lambda$ CDM
$\Omega_\Lambda$	0	0.689
$\Omega_m$	1	0.311
$\Omega_r$	0	0.017
$\Omega_k$	0	-0.017

TABLE I: Valores de  $\Omega_\Lambda$ ,  $\Omega_m$ ,  $\Omega_r$  and  $\Omega_k$  en cada modelo.

## I. INTRODUCCIÓN

En 2011 el Premio Nobel de Física fue concedido a Saul Perlmutter, Brian P. Schmidt y Adam G. Reiss por el descubrimiento de la [expansión acelerada del universo](#)[2] mediante supernovas de tipo Ia (SnIa). La idea de este trabajo es reproducir estos resultados y comprobar que la expansión del universo es una expansión acelerada. Para ello, se han obtenido los datos medidos por los ganadores del Nobel y se ha obtenido el módulo de la distancia mediante la Ecuación 1.

$$\mu = m - M_v, \quad (1)$$

donde  $\mu$  es el módulo de la distancia,  $m$  es la magnitud aparente y  $M_v = -19.3$  es la magnitud absoluta en el espectro óptico.

Seguidamente, se han obtenido estimaciones teóricas de este módulo de la distancia mediante dos modelos diferentes; el modelo Einstein-de Sitter y la concordancia  $\Lambda$ CDM. Para ello, se ha utilizado la Ecuación 2.

$$\mu = 5 \cdot \log_{10}(d) - 5, \quad (2)$$

donde  $d$  es la distancia entre la supernova y la Tierra. El principal problema es la medida de la distancia, pues hay varios métodos, dando cada uno un resultados distintio. Para estas mediciones se ha optado por la medición de la distancia de luminosidad ( $D_L$ ), que viene dada por la Ecuación 3.

$$D_L = c \frac{1+z}{H_0} \int_0^z \frac{1}{E(z')} dz', \quad (3)$$

donde  $c$  es la velocidad de la luz,  $z$  es el corrimiento al rojo,  $H_0$  es la constante de Hubble en la actualidad ( $H_0 = 67.59$  km/s Mpc) y donde  $E(z)$  es la siguiente expresión:

$$E(z) = \sqrt{\Omega_m(1+z)^3 + \Omega_k(1+z)^2 + \Omega_r(1+z)^4 + \Omega_\Lambda}.$$

Para cada modelo, se tienen distintos valores de cada  $\Omega$ , los cuales se pueden observar en la Tabla I. Tanto los valores de  $H_0$  como de  $\Omega_\Lambda$  y  $\Omega_m$  han sido obtenidos del *Data Release de la colaboración Planck de 2015* ([PR2-2015](#)<sup>1</sup>). El motivo por el que se ha optado por la distancia de luminosidad y el uso de SnIa es por el hecho de que estas supernovas tienen todas la misma luminosidad absoluta (que no aparente) al tener todas el mismo origen; la explosión de una enana blanca de C-O en presencia de una compañera gigante, por lo que son una buena candela estándar. Sabiendo entonces su corrimiento al rojo y el tipo de universo en el que estamos, se puede obtener la distancia (como en nuestro caso) o utilizar la distancia y el corrimiento al rojo para saber cómo es nuestro universo.

## II. PROCEDIMIENTO

En primer lugar se han obtenido los datos de los grupos de Perlmutter, Schmidt y Reiss, almacenandolos en dos ficheros distintos “datos1.dat”, pertenecientes a las observaciones SCP SN<sub>E</sub> I<sub>A</sub> [1] y “datos2.dat” pertenecientes a

<sup>1</sup> <http://pla.esac.esa.int/pla/#cosmology>

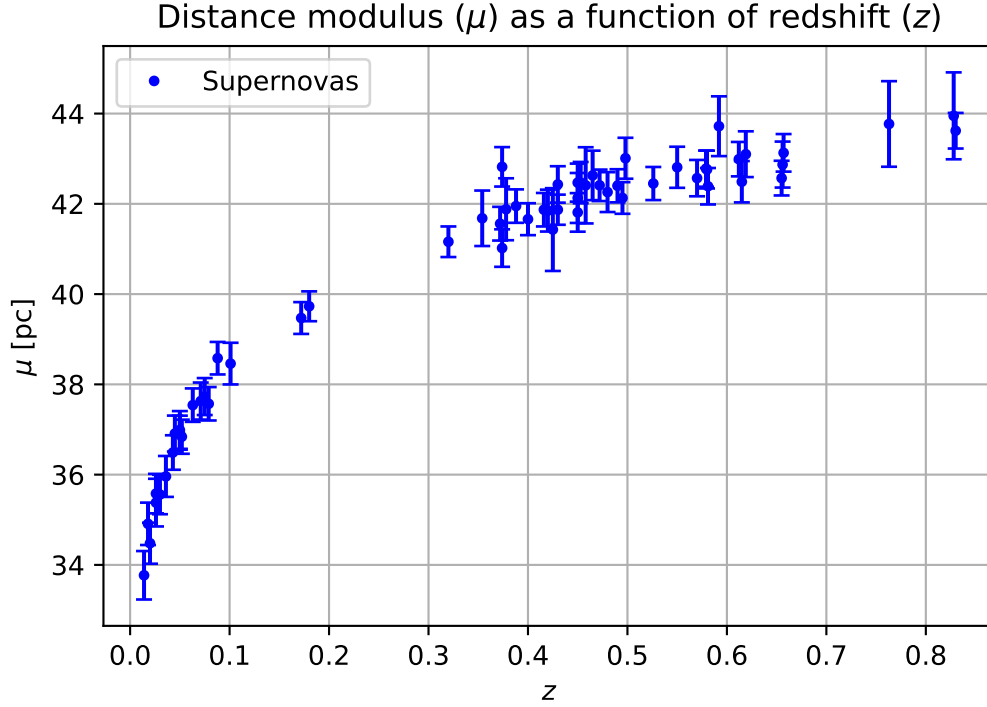


FIG. 1: Representación del módulo de la distancia ( $\mu$ ) frente al corrimiento al rojo ( $z$ ) de las observaciones de Perlmutter.

las observaciones CALÁN/TOLOLO SN<sub>E</sub> I<sub>A</sub>.<sup>[1]</sup> Estos ficheros pueden verse en la carpeta “Data” en mi repositorio público de [GitHub](#), el enlace se encuentra puede verse en la sección V.

Posteriormente, se han obtenido los valores de  $H_0$ ,  $\Omega_m$  y  $\Omega_\Lambda$  del PR2-2015. Al dar seis valores distintos, se ha optado por hacer la media de estos para cada variable, siendo finalmente el valor utilizado  $H_0 = 67.59$  km/s Mpc. Los valores de  $\Omega_m$  y  $\Omega_\Lambda$  se han utilizado para el modelo de la concordancia  $\Lambda$ CDM, pues en el modelo de Einstein-de Sitter, por definición,  $\Omega_m \equiv 1$ .

Para obtener la distancia de luminosidad en cada modelo se ha necesitado realizar una integral, la cual ha sido aproximada numéricamente mediante la función ‘`integrate(función, varibale)`’ del módulo ‘`scipy`’ de Python, que están disponible al en el Anexo (Sección V). Una vez realizada esta primera parte, se han añadido más datos de supernovas procedentes del “Supernova Cosmology Project” (SCP)<sup>[3]</sup>, que se encuentran en la carpeta “Data” de mi repositorio público de [GitHub](#) bajo el nombre “datos.dat” o en el siguiente [enlace](#)<sup>2</sup>.

### III. RESULTADOS

Una vez obtenidos los datos de “datos1.dat” y “datos2.dat” se ha realizado la Figura 1, mostrando los pares de datos obtenidos por Perlmutter y su grupo mediante la Ecuación 1. A continuación, se ha obtenido los valores teóricos del módulo de la distancia mediante la ecuación 2, pudiendo verse los modelos Einstein-de Sitter y el modelo de concordancia  $\Lambda$ CDM en la Figura 2. Como se puede ver, todo parece indicar que el modelo de concordancia  $\Lambda$ CDM tiene mayor similitud con los puntos observados. Para salir de dudas, se han añadido más datos (Figura 3) y otro modelo (Figura 4), confirmandonos el resultado.

<sup>2</sup> [https://supernova.lbl.gov/Union/figures/SCPUnion2.1\\_mu\\_vs\\_z.txt](https://supernova.lbl.gov/Union/figures/SCPUnion2.1_mu_vs_z.txt)

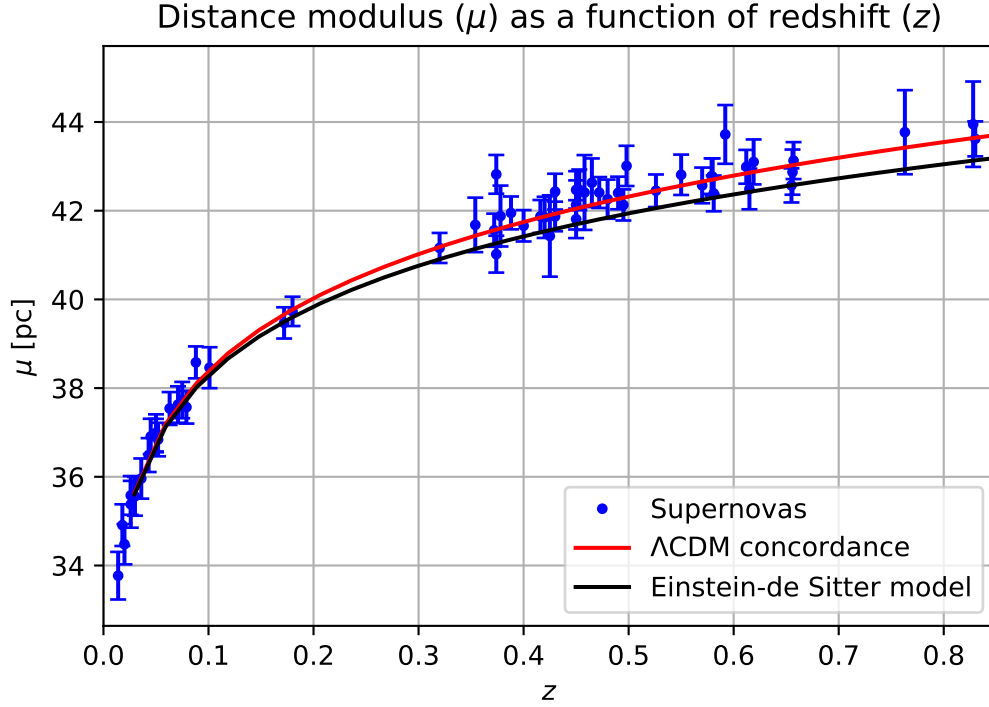


FIG. 2: Representación del módulo de la distancia ( $\mu$ ) frente al corrimiento al rojo ( $z$ ) de las observaciones de Perlmutter. Superpuesto, pueden verse los modelos de Einstein-de Sitter y de concordancia  $\Lambda$ CDM

Se ha optado por introducir otro modelo con  $\Omega_m = \Omega_\Lambda = 0.5$  una vez se ha visto que el término  $\Omega_\Lambda$  tiene un peso mucho mayor que  $\Omega_r$  u  $\Omega_k$ . Aunque también tiene mayor peso que  $\Omega_m$ , la diferencia entre este modelo y el de concordancia  $\Lambda$ CDM no es tanta. También se ha realizado la comprobación con otro modelo con  $\Omega_\Lambda = 0.7$  y  $\Omega_m = 0.3$ , pero no resultaba diferenciable del modelo de concordancia  $\Lambda$ CDM.

#### IV. CONCLUSIONES

Una vez representados los datos y comparados con distintos modelos, es trivial llegar a la conclusión de que el modelo que mejor se ajusta a los datos observacionales es el modelos de concordancia  $\Lambda$ CDM, lo que implicaría que el universo se encuentra en expansión acelerada

#### V. ANEXO

El código utilizado puede encontrarse a continuación, y también en mi repositorio público de [GitHub](https://github.com/ivanvillegas7/Supernovas-and-the-accelerated-expansion-of-the-universe) junto a todos los archivos creados. En el siguiente enlace: <https://github.com/ivanvillegas7/Supernovas-and-the-accelerated-expansion-of-the-universe>

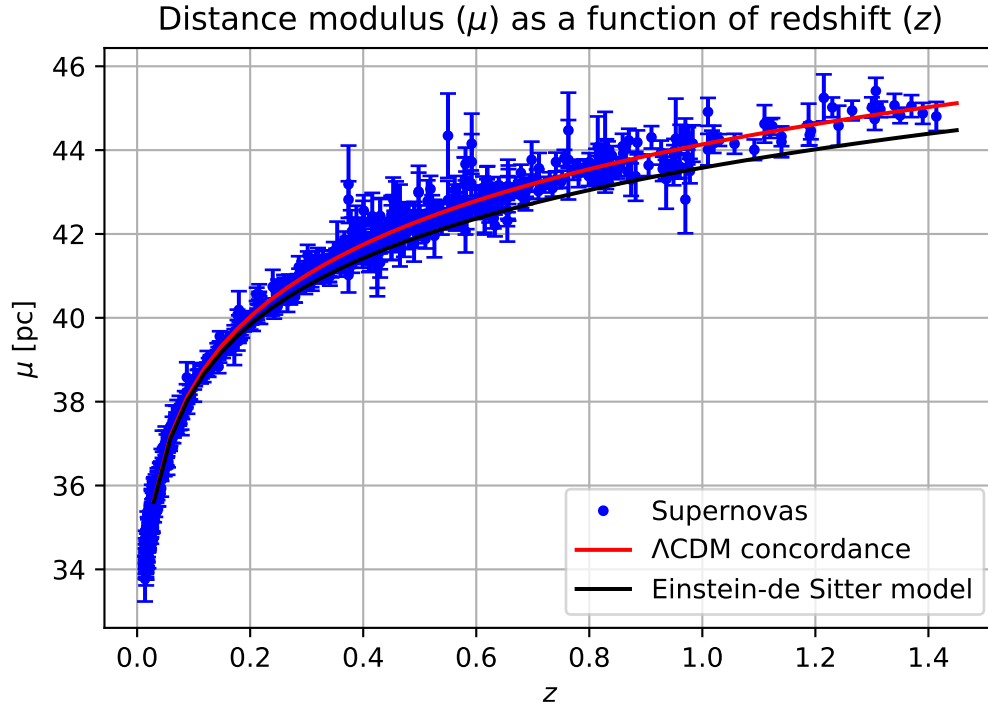


FIG. 3: Representación del módulo de la distancia ( $\mu$ ) frente al corrimiento al rojo ( $z$ ) de las observaciones de Perlmutter y las observaciones del SCP. Superpuesto, pueden verse los modelos de Einstein-de Sitter y de concordancia  $\Lambda$ CDM

```
# -*- coding: utf-8 -*-
"""
Created on Sun Nov 20 15:04:05 2022

@author: Iván
"""
#%% Importamos los módulos que se van a utilizar

from typing import List

import numpy as np

import matplotlib.pyplot as plt

import urllib.request

import pandas as pd

import scipy.integrate as integrate

#%% Importamos los datos

M_v: float = -19.3

j: int = 0

with open("datos1.dat", "r") as infile:

    lines = infile.readlines()

    z1: List[str] = []
```

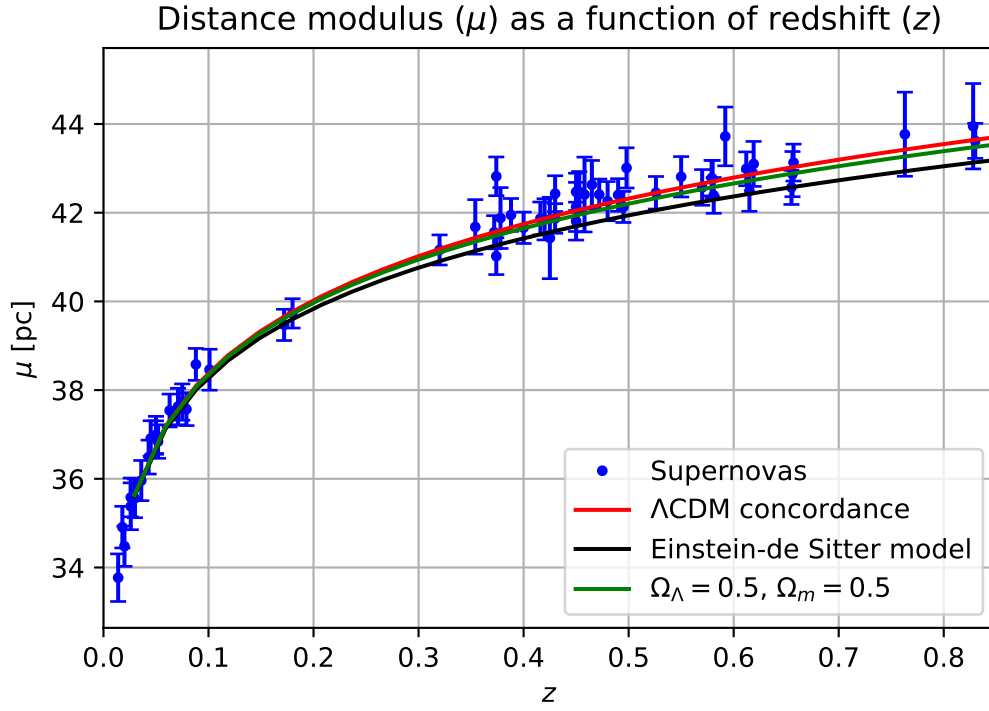


FIG. 4: Representación del módulo de la distancia ( $\mu$ ) frente al corrimiento al rojo ( $z$ ) de las observaciones de Perlmutter. Superpuesto, pueden verse los modelos de Einstein-de Sitter, de concordancia  $\Lambda$ CDM y un tercero con  $\Omega_{\Lambda} = \Omega_m = 0.5$ .

```

r1: List[float] = []

r_err1: List[float] = []

for line in lines:

    #Nos saltamos la primera fila del archivo

    if j==0:

        j+=1

    else:

        vals = line.split()

        z1.append(float(vals[1]))

        r1.append(float(vals[8])-M_v)

        r_err1.append((float(vals[8])-M_v)*float(vals[9])/float(vals[8]))

k: int = 0

with open("datos2.dat", "r") as infile:

    lines = infile.readlines()

    z2: List[str] = []

    r2: List[float] = []

    r_err2: List[float] = []

```

```

for line in lines:

    #Nos saltamos la primera fila del archivo

    if k==0:

        k+=1

    else:

        vals = line.split()

        z2.append(float(vals[1]))

        r2.append(float(vals[8])-M_v)

        r_err2.append((float(vals[8])-M_v)*float(vals[9])/float(vals[8]))

targetURL = "https://supernova.lbl.gov/Union/figures/SCPUnion2.1_mu_vs_z.txt"

i = 0

z = []

r = []

r_er = []

for line in urllib.request.urlopen(targetURL):

    #Nos saltamos las 5 primeras filas del archivo

    if i <= 5:

        column_names = line

        i+=1

    else:

        z.append(float(line.split()[1]))

        r.append(float(line.split()[2]))

        r_er.append(float(line.split()[3]))

output_df = pd.DataFrame({column_names[:17]: z, column_names[27:-1]: r,\
                          column_names[27:-1]: r_er})

output_df.to_csv('datos.dat')

##Tarea1

plt.figure()

plt.plot(z1, r1, '.', label='Supernovas', color='blue')

for i in range(len(r1)):

    plt.errorbar(z1[i],r1[i], yerr = r_err1[i], capsize = 3, color='blue')

plt.plot(z1, r1, '.', color='blue')

for i in range(len(r2)):

    plt.errorbar(z2[i],r2[i], yerr = r_err2[i], capsize = 3, color='blue')

plt.xlabel(r'$z$')

```

```

plt.ylabel(r'$\mu$ [pc]')
plt.title(r'Distance modulus ($\mu$) as a function of redshift ($z$)')
plt.legend()
plt.grid(True)

plt.savefig('r frente a z - sin modelos.pdf')

#%%Tarea2

H0: float = (67.31+67.81+67.9+67.27+67.51+67.74)*1e-3/6 # km/s Mpc -> m/s pc

c: float = 3e8 #m/s

Vacío: float = (0.685+0.692+0.6935+0.6844+0.6879+0.6911)/6

Materia: float = (0.315+0.308+0.3065+0.3156+0.3121+0.3089)/6

Curvatura: float = (0.0008-0.052-0.005-0.0001-0.04-0.004)/6

Radiación: float = 1-Vacío-Materia-Curvatura

CDM: List[float] = []

def E_CDM(x: np.array)-> np.array:

    return 1/np.sqrt(Materia*(1+x)**3+Vacío+Radiación*(1+x)**4+Curvatura*(1+x)**2)

EdS: List[float] = []

def E_EdS(x: np.array)-> np.array:

    return 1/np.sqrt((1+x)**3)

C_C: List[float] = []

def E_C_C(x: np.array)-> np.array:

    return 1/np.sqrt(0.5*(1+x)**3+0.5)

z_o: np.array = np.linspace(0, 1.45)

for i in range(len(z_o)):

    x: np.array = np.array(np.linspace(0, z_o[i]))

    CDM.append(5*np.log10(((c/H0)*(1+z_o[i])*integrate.simpson(E_CDM(x), x)))-5)

    EdS.append(5*np.log10(((c/H0)*(1+z_o[i])*integrate.simpson(E_EdS(x), x)))-5)

    C_C.append(5*np.log10(((c/H0)*(1+z_o[i])*integrate.simpson(E_C_C(x), x)))-5)

plt.figure()

plt.plot(z1, r1, '.', label='Supernovas', color='blue')

for i in range(len(r1)):

    plt.errorbar(z1[i], r1[i], yerr = r_err1[i], capsize = 3, color='blue')

plt.plot(z1, r1, '.', color='blue')

for i in range(len(r2)):

```



```

plt.errorbar(z2[i],r2[i], yerr = r_err2[i], capsize = 3, color='blue')

plt.plot(z_o, CDM, color='red', label='$\Lambda$CDM concordance')
plt.plot(z_o, EdS, color='black', label=r'Einstein-de Sitter model')
plt.xlabel(r'$z$')
plt.ylabel(r'$\mu$ [pc]')
plt.title(r'Distance modulus ($\mu$) as a function of redshift ($z$)')
plt.legend()
plt.xlim(0, 0.85)
plt.grid(True)

plt.savefig('r frente a z.pdf')
###Objetivos avanzados

plt.figure()

plt.plot(z1, r1, '.', label='Supernovas', color='blue')
for i in range(len(r1)):
    plt.errorbar(z1[i],r1[i], yerr = r_err1[i], capsize = 3, color='blue')
plt.plot(z1, r1, '.', color='blue')
for i in range(len(r2)):
    plt.errorbar(z2[i],r2[i], yerr = r_err2[i], capsize = 3, color='blue')
plt.plot(z, r, '.', color='blue')
for i in range(len(r)):
    plt.errorbar(z[i],r[i], yerr = r_er[i], capsize = 3, color='blue')
plt.plot(z_o, CDM, color='red', label='$\Lambda$CDM concordance')
plt.plot(z_o, EdS, color='black', label=r'Einstein-de Sitter model')
plt.xlabel(r'$z$')
plt.ylabel(r'$\mu$ [pc]')
plt.title(r'Distance modulus ($\mu$) as a function of redshift ($z$)')
plt.legend()
plt.grid(True)

plt.savefig('r frente a z extra datos.pdf')
###Objetivos avanzados 2

plt.figure()

plt.plot(z1, r1, '.', label='Supernovas', color='blue')
for i in range(len(r1)):

```

```

plt.errorbar(z1[i],r1[i], yerr = r_err1[i], capsize = 3, color='blue')
plt.plot(z1, r1, '.', color='blue')
for i in range(len(r2)):
    plt.errorbar(z2[i],r2[i], yerr = r_err2[i], capsize = 3, color='blue')
plt.plot(z_o, CDM, color='red', label=r'\Lambda$CDM concordance')
plt.plot(z_o, EdS, color='black', label=r'Einstein-de Sitter model')
plt.plot(z_o, C_C, color='green', label=r'\Omega_\Lambda=0.5$, \Omega_m=0.5$')
plt.xlabel(r'$z$')
plt.ylabel(r'$\mu$ [pc]')
plt.title(r'Distance modulus ($\mu$) as a function of redshift ($z$)')
plt.legend()
plt.xlim(0, 0.85)
plt.grid(True)

plt.savefig('r frente a z extra modelos.pdf')

```

## BIBLIOGRAPHY

- [1] S. Perlmutter et al. “Measurements of  $\Omega$  and  $\Omega_0$  from 42 High-Redshift Supernovae”. In: *The Astrophysical Journal* 517.2 (June 1999), p. 565. DOI: [10.1086/307221](https://doi.org/10.1086/307221). URL: <https://dx.doi.org/10.1086/307221>.
- [2] Royal Swedish Academy of Sciences. “THE ACCELERATING UNIVERSE”. In: (2011). URL: <https://www.nobelprize.org/uploads/2018/06/advanced-physicsprize2011.pdf>.
- [3] Supernova Cosmology Project. *Introduction to the SCP Union2.1 Compilation*. URL: <https://supernova.lbl.gov/Union/>. (accessed: 20.11.2022).