Tutorials hands-on understanding stellar evolution of massive stars using MESA stellar evolution code data.

Objectives:

1. Understand the evolution of the star.
2. Learn about the structure of the star.

Preparations:

1. MESA data (history and profile data)
2. Evolution and profile script (provided)

We are going to use computed data from MESA stellar evolution code to plot the evolution and structure of 20 $M_{\odot}$ model. The python script is available from https://github.com/lizayusof/VSOA_2024

We shall follow the description from
http://user.astro.wisc.edu/~townsend/resource/teaching/astro-310-F21/python-lab/mesa-web-history.html#exploring-a-history-file

but we are going to modify it for massive star model.

For this exercise, we shall produce:

a) For the evolution

1. HR diagram
2. Temperature vs density diagram
3. Evolution of mass with respect to star age
4. Evolution of stellar abundance with respect to the star age

b) For the structure

1. Radius vs mass
2. Radius vs luminosity
3. Radius vs density
4. Radius vs stellar abundance

Steps to start the exercise:

1. Create a folder and renamed it as VSOA8
2. Go to https://github.com/lizayusof/VSOA_2024
3. Download `mesa_web` Python module to your folder
4. Open jupyter-notebook (it will open in your browser) and save the notebook as history_notebook.ipynb
5. We shall use this notebook to plot and understand the evolution of the massive stars.

From your notebook, write the following command to read the libraries and module

```
[1]: # Import the numpy module to provide numerical functionality
import numpy as np

# Import the matplotlib.pyplot module to provide plotting functionality
import matplotlib.pyplot as plt

# Tell matplotlib.pyplot to do inline plots
%matplotlib inline

# Import the mesa_web module to simplify reading MESA-Web files
import mesa_web as mw
```

Read the data and print the list to check the list of history data

```
[2]: #read the data from the directory
hist_data=mw.read_history('20p0msun_data/LOGS/history.data')
```

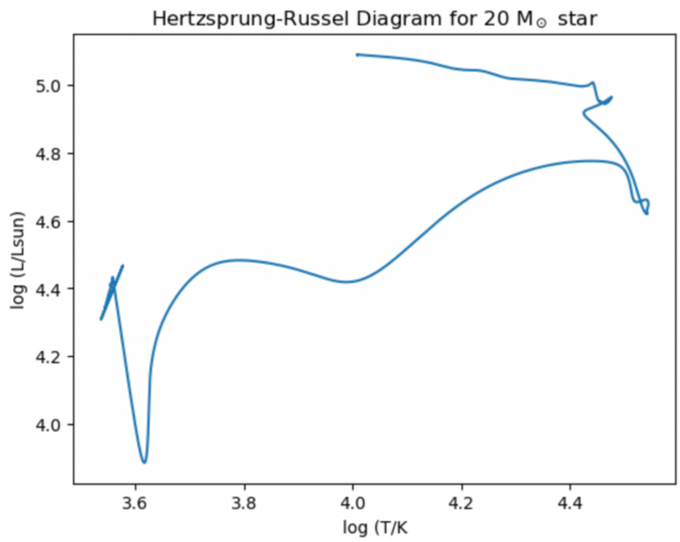
```
[3]: #check the list of data
print(type(hist_data))
print(hist_data.keys())
```

For more details, type help

```
[4]: help(mw.read_history)
```

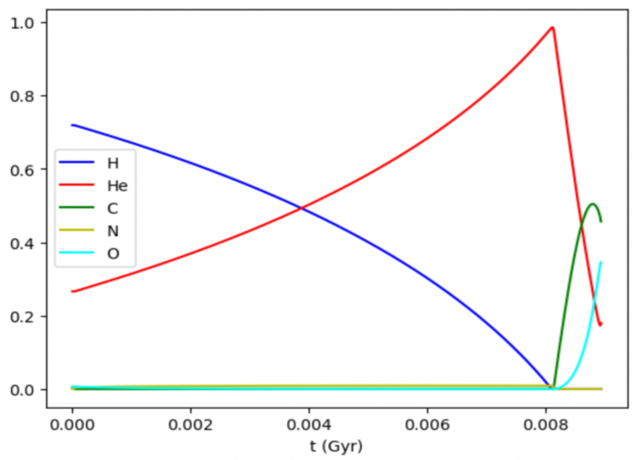
```
# Extract data from hist_data, using dict indexing

log_Teff = hist_data['log_Teff'] # log(Teff/K)
log_L = hist_data['log_L']       # log(L/Lsun)

# Plot the HR diagram

plt.figure()

plt.plot(log_Teff, log_L)

#plt.xlim(4.0, 3.5)
#plt.ylim(-1,4)

plt.xlabel('log (T/K')
plt.ylabel('log (L/Lsun)')

plt.title('Hertzsprung-Russel Diagram for 20 M$_\odot$ star')
```

The result:



We extract the central abundances vs time (you can plot all the abundances in the history data).

```
# Plot central  abundances versus time (measured
# since the start of the calculation)

X_c = hist_data['center_h1']
Y_c = hist_data['center_he4']
C_c = hist_data['center_c12']
N_c = hist_data['center_n14']
O_c = hist_data['center_o16']
t = hist_data['star_age']

plt.figure()

plt.plot(t/1E9, X_c, color='b', label='H')
plt.plot(t/1E9, Y_c, color='r', label='He')
plt.plot(t/1E9, C_c, color='g', label='C')
plt.plot(t/1E9, N_c, color='y', label='N')
plt.plot(t/1E9, O_c, color='cyan', label='O')

plt.xlabel('t (Gyr)')

plt.legend()
```



We can also plot the evolution of mass with time and observed the mass loss during the evolution

```
# Plot time versus model number

t = hist_data['star_age']
mass = hist_data['star_mass']

plt.figure()

plt.plot(t/1E9, mass, color='b')

plt.ylabel('Mass')
plt.xlabel('t (Gyr)')
```