NEAT

Nebular empirical analysis tool

Documentation last updated 5 April 2011

1 Introduction

NEAT (Nebular empirical analysis tool) is a code designed for the quick analysis of emission line spectra of photoionised nebulae. It is designed to be very simple but to return robust and meaningful results.

2 Installation

The fortran 90 source code is available from http://github.com/worldtraveller/NEAT. If you have git installed you can checkout the source code, or otherwise you can download a tar.gz file.

You will also need the relevant atomic data, available from http://github.com/worldtraveller/Atom data. This should be placed in a directory called atomic_data06 in the directory you put NEAT into. If you type ‘ls’ in this directory, you should see:

00compile 00compilegfortran 01compile atomic_data06 examples manual README source

Once you’ve got the source code and the atomic data, you should be able to compile the code by typing ./00compilegfortran. If you don’t have gfortran installed then any modern fortran compiler should work as well.

3 Running the code

You can test that the code is working by typing

./abundances.exe 1 examples/6543_proc_errors

You should see a lot of output to the terminal, ending with a section titled ”Abundance Discrepancy Factors”.

3.1 Inputs

The code is run by typing

./abundances.exe integer filename

The filename should be a plain text file, containing three columns. The first is a rest wavelength, which the code will use to identify the line. The wavelengths should correspond exactly to those listed in source/Ilines_levs. The second column should be a flux per unit wavelength (any units are fine), and the third column should be the uncertainty on the flux, given in the same units.

The integer is the number of iterations to carry out. If this number is 1, then the code simply calculates all of the temperatures, densities, ionic and total abundances that it can, using the line list provided. If it is greater than 1, then the code will also calculate uncertainties using a Monte Carlo technique, in which the nebular parameters are calculated repeatedly, each time drawing the line flux randomly from a Gaussian distribution, centred on the quoted flux and with a standard deviation equal to the quoted uncertainty.

The higher the number of iterations, the better sampled the uncertainty distribution of the output parameters will be. Around 10,000 iterations should be sufficient in most cases.

3.2 Outputs

At the moment, the code simply outputs its results to the terminal.