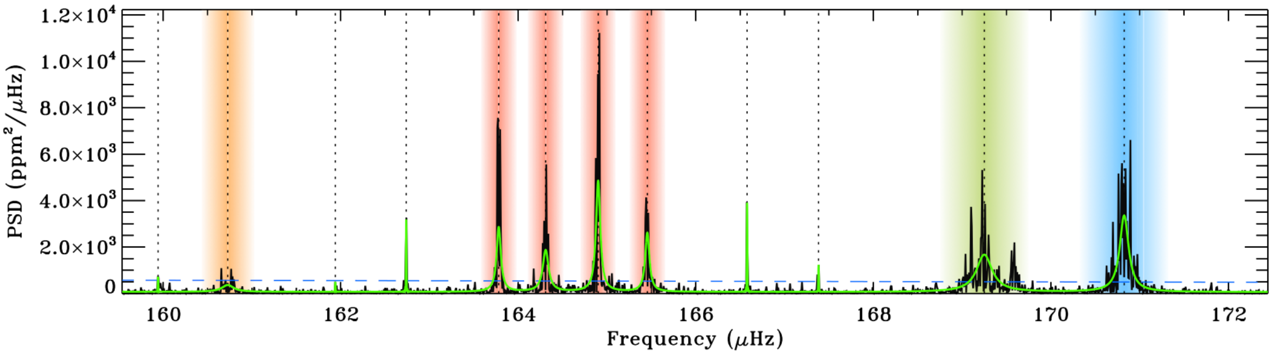
Hands on computer II

Fitting background and oscillations in a solar-like star using the Bayesian code

DIAMONDS™
BAYESIAN SOFTWARE

Enrico Corsaro

Space-Inn School on Astero/Helioseismology and Stellar/Solar Physics
27th October 2015



To do a fruitful hands on computer session on Tuesday 27th October 2015, you will need to have the DIAMONDS code already installed in your machine, as well as the exercises ready to be launched. I assume for simplicity that you have an OS X based system. UNIX systems are also fine.

1. Downloading and installing DIAMONDS

Before starting, I would suggest you to take a look at the original paper work of the code, which can be found at http://adsabs.harvard.edu/abs/2014A%26A...571A..71C
and to download the user guide manual available at
https://fys.kuleuven.be/ster/Software/Diamonds/diamonds_userguide/at_download/file
These documents will be essential helpful to assist you in understanding the different code parts, its working principle, troubleshooting, and for properly using it for our application. They will also be presented during our session.

Now please carefully follow the steps below. You find a detailed description in the official website of the code, under the section Installation Guide.

1) Go to the official website of the code https://fys.kuleuven.be/ster/Software/Diamonds/
2) Go to the Download section, fill in the form with your affiliation details and download the package available (v. 1.0).
3) After having downloaded the diamonds1.0.tar.gz file, untar and unzip it in a local folder. A folder named Diamonds will then appear. For the moment, ignore its content.
4) Before installing Diamonds, you have to make sure that you have installed the CMake compiler, which is a free software capable of compiling C, C++ source codes by recognizing the most suited compiler installed in your machine, depending on the platform you have got (normally clang for OS X, and gcc for Linux systems). I recommend you install the version 2.8.12.2, whose dmg is available for Mac OS X 10.6 or later from the link http://www.cmake.org/files/v2.8/cmake-2.8.12.2-Darwin64-universal.dmg.You can however find the other versions at http://www.cmake.org/download/ in case you are using another platform.
   Importantly: make sure you install CMake command line tool as well, since you need that to compile Diamonds via terminal. To do so, just open CMake app, then go to Tools/Install for Command Line Use.
5) Once you have installed CMake, we can proceed by compiling Diamonds. To do so, go to your Diamonds directory, then open the ASCII file CMakeLists.txt. Inside you have the first lines commented with what you have to do. Simply follow this guideline:
   from your Diamonds directory, via terminal type:

```
$ mkdir build
$ cd build
$ cmake ..
$ make -j 4
```

The 4 option takes into account the number of CPU cores you have available (e.g. 4 for a quad-core CPU). Change it according to this number.
Then the compilation and linking is executed and after a while you should display an ending message as

```
Linking CXX shared library libdiamonds.dylib
[100%] Built target diamonds
```

Then you have successfully installed Diamonds as a library in your computer.

2. Downloading and installing DIAMONDS extension for background fitting

The second step to accomplish is to set up DIAMONDS for our application. The first application that we will do regards the fitting of a background signal in the stellar power spectrum.

To download the background extension if DIAMONDS go to
https://fys.kuleuven.be/ster/Software/Diamonds/package/background_extension1.0.tar.gz

By unpacking the archive file you will find a folder labeled Background and with a structure similar to that of Diamonds. This extension you need to compile separately from Diamonds, but only after you have compiled Diamonds (see Section 1). Diamonds is used as a library for this extension. The compilation commands are the same as for Diamonds. Before proceeding with the compilation make sure you put the Background folder at the same level of that of Diamonds (BE AWARE: not the Diamonds folder itself, but the parent directory where you contain Diamonds since the Background extension has to be in a different path). Then, using any text editor, open the files

/build/multirun	(SCRIPT FILE)
/source/RedGiantBackground.cpp	(MAIN FILE FOR BACKGROUND FITTING)

and edit the local paths specified between **** to match those of your own machine.
Then in your build directory, via terminal type:

```
$ cmake ..
$ make -j 4
```

1) In this extension I have provided to you an example of format for data and configuring files for one *Kepler* red giant. It should work out of the box.
2) In the data folder you find the latest Q0-Q17 dataset provided by Dr. Rafael A. Garcia in an ASCII format (first column frequency, second column power spectral density).
3) In the results folder you find the folder with the same name of the data file (the KIC ID of the star) and inside three configuring files that you can edit when running your code (see presentation during hands on computer II). One file in particular provides the prior boundaries for the input parameters.
4) In the results folder of the star there is also a subfolder labeled 00 that contains results already computed and that you can read and plot. I will explain you how to do that during the hands on computer II session.
5) The background model adopted is given in the RedGiantBackgroundModel.cpp file (see presentation during hands on computer II).
6) In the folder idl you find a complete set of idl routines that are useful to plot the results, which will be described during the hands on computer II. If you don't have IDL installed in your machine, you can still follow the presentation where I will show how the routines work and how the results appear.

In this way everything is set up, and the code does not need to be recompiled every time you need to change input data and prior boundaries. All you have to do is to edit the ASCII files for the code configuration, which must be present in each star folder (they will change of course depending on the star you analyze).

3. Downloading and installing DIAMONDS extension for oscillation fitting

The second application that we will do concerns the fitting of the oscillation modes in the stellar power spectrum and their significance test. This requires you install a second extension of DIAMONDS, that for the peak bagging analysis. The procedure is perfectly analogous to that explained in Section 2.

To download the peak bagging extension if DIAMONDS go to
https://fys.kuleuven.be/ster/Software/Diamonds/package/peakbagging_extension1.0.tar.gz

By unpacking the archive file you will find a folder labeled PeakBagging and with a structure similar to that of Diamonds. This extension you need to compile separately from Diamonds, but only after you have compiled Diamonds (see Section 1). Diamonds is used as a library for this extension. The compilation commands are the same as for Diamonds and its Background extension. Before proceeding with the compilation make sure you put the PeakBagging folder at the same level of that of Diamonds (BE AWARE: not the Diamonds folder itself, but the parent directory where you contain Diamonds since the PeakBagging extension has to be in a different path). Then, using any text editor, open the files

build/multiple_run	(SCRIPT FILE)
idl/kic_evidence.pro	(IDL FILE FOR PEAK SIGNIFICANCE TEST)
idl/kic_plot_peakbagging.pro	(IDL FILE TO PLOT RESULTS)
source/RedGiantPeakBagging.pro	(MAIN FILE FOR PEAK BAGGING)

and edit the local paths specified between **** to match those of your own machine.
Then in your build directory, via terminal type:

```
$ cmake ..
$ make -j 4
```

1) In this extension I have provided to you an example of format for data and configuring files for the same *Kepler* red giant of Section 2. It should work out of the box.
2) In the data folder you find the latest Q0-Q17 dataset provided by Dr. Rafael A. Garcia in an ASCII format (first column frequency, second column power spectral density) but this time trimmed in frequency range around the region of the power excess.
3) In the results folder you find the folder with the same name of the data file (the KIC ID of the star) and inside two subfolders (run_4 and run_4A) and one configuring file containing the solution of the background fitting (see presentation during hands on computer II). One file in particular provides the prior boundaries for the input parameters.
4) In both the results folders of the star run_4 and run_4A there is a subfolder labeled 00 that contains results already computed and that you can read and plot, and all the files containing the configuring parameters of the code. The two folders will allow for a demonstration of the Bayesian model comparison. See the presentations for explanation.
5) The oscillation fitting model adopted is given in the LorentzianSincMixtureModel.cpp file (see presentation during hands on computer II).
6) In the folder idl you find a complete set of idl routines that are useful to plot the results, which will be described during the hands on computer II. If you don't have IDL installed in your machine, you can still follow the presentation where I will show how the routines work and how the results appear.

In this way everything is set up, and the code does not need to be recompiled every time you need to change input data and the oscillation peaks you want to fit. All you have to do is to edit the ASCII files for the code configuration, which must be present in each fitted chunk for each star folder (they will change of course depending on the star you analyze).