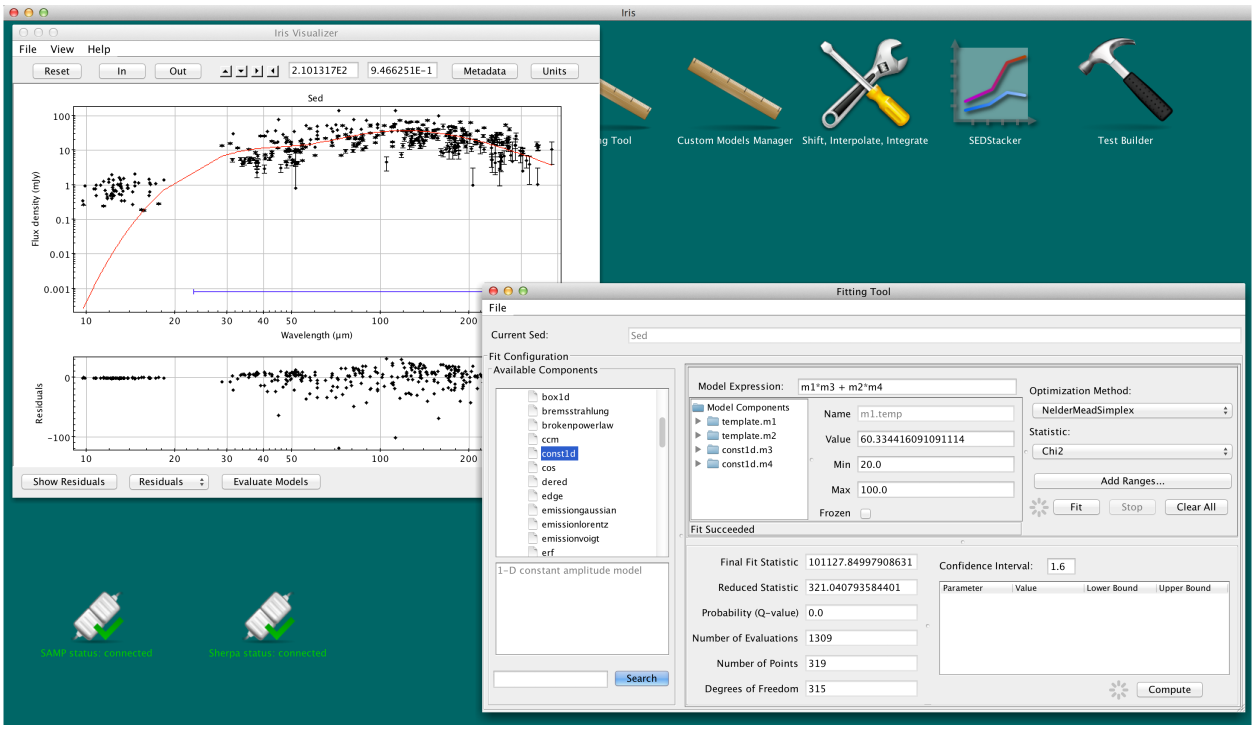
Template Fitting Worksheet

This worksheet shows how to import and use template libraries with other libraries and models in a fitting session.



Note: the `<path-to-iris>` refers to the full path to the directory in which the Iris demo material was downloaded. If you used Github to download the sources, `<path-to-iris>` will be `<path-to>/aas229iris`. If you downloaded the data from the thumb drive, it'll be `<path-to>/iris`.

Launch Iris if it is not already open.

```
$ source activate iris-workshop
(iris-workshop) $ iris
```

PART 1: Importing the custom template library

Description and background

Users can import their own Python functions and templates into Iris via the **Custom Models Manager**.

Templates must be in ASCII format, with at least two columns. Only the first two columns are read in. The first column is the spectral axis, which must be in Angstroms, and the second is the flux.

There is an example template library included in the **demo directory** Iris package:

```
$ cd <path-to-iris>/worksheets/template_fitting/
$ head -4 \ modified-blackbody-templates/modified-blackbody-template_20K.dat
```

```
# col0 col1
10000.0 0.0
10069.3863148 0.0
10139.2540756 0.0
```

Template libraries require a *template model index file* which maps the grid of template parameter values to the corresponding template file on disk. The index file follows the same format as in Sherpa.

```
$ cat modified-blackbody-templates.txt
# TEMP  REFER     BETA MODELFLAG FILENAME
20      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_20K.dat
25      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_25K.dat
30      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_30K.dat
35      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_35K.dat
40      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_40K.dat
45      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_45K.dat
50      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_50K.dat
55      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_55K.dat
60      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_60K.dat
65      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_65K.dat
70      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_70K.dat
75      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_75K.dat
80      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_80K.dat
85      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_85K.dat
90      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_90K.dat
95      1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_95K.dat
100     1000000   1.5  1   modified-blackbody-templates/modified-blackbody-template_100K.dat
```

Each row in the index file corresponds to a template. The first columns contain the template parameter values, which the user defines. In this example, `TEMP`, `REFER`, and `BETA` are the template parameters, and only `TEMP` changes between the templates. The second-to-last column, “`MODELFLAG`,” is mandatory, and specifies if the corresponding template should be used (`MODELFLAG = 1`) or not (`MODELFLAG = 0`). The last column shows the **full path location** of the template on disk.

The parameters of this modified blackbody library are the temperature `TEMP`, the reference wavelength `REFER`, and the dust emissivity `BETA`. Dust emissivity is degenerate with temperature, and so we choose to keep beta constant.

Note: Templates and table models follow the same file format. The difference between importing a table model (which is a single template) and a template library (a list of templates), is the template model index file.

Step by step: importing a library of templates into Iris

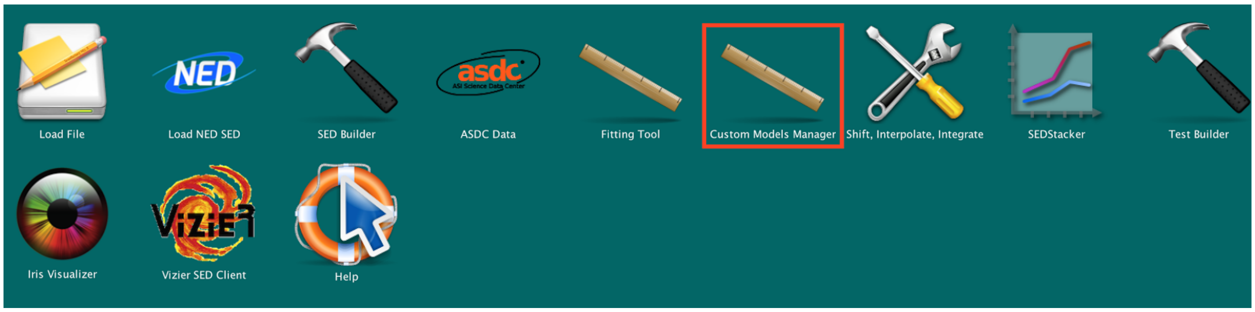
1. First, you will need to **edit the template model index file** in `modified-blackbody-templates.txt`, so that the **template paths reflect the path on your system**. For example, if my templates are located in `/Users/jbudynk/aas229iris/worksheets/template_fitting/modified-blackbody-templates`

   I would have

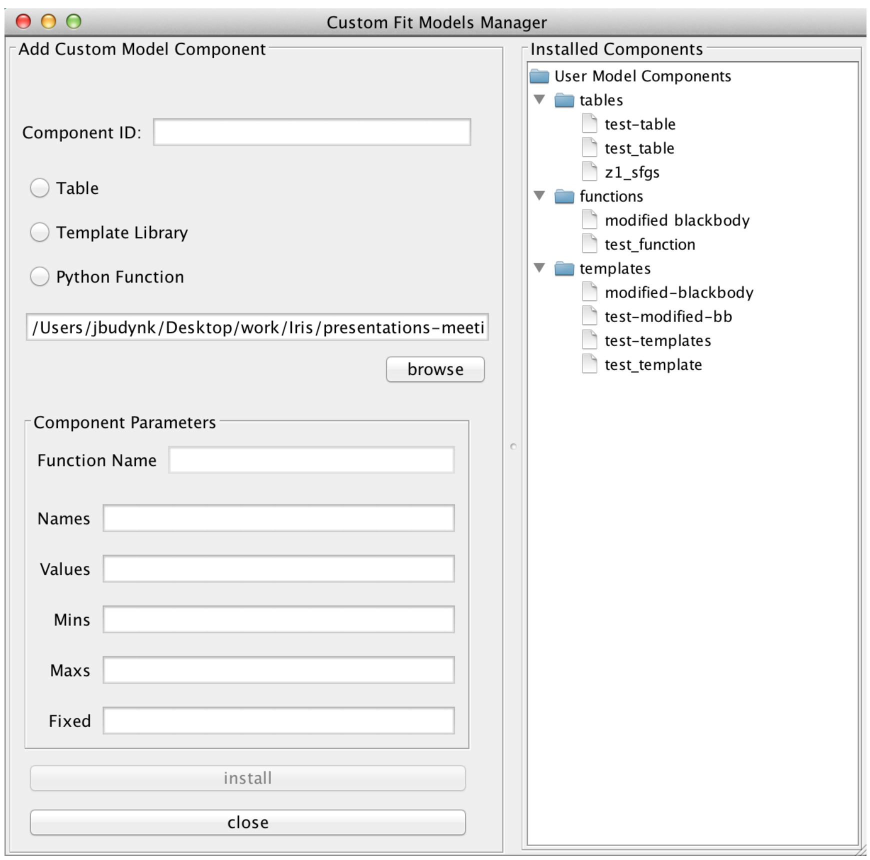
```
$ head -3 modified-blackbody-templates.txt
# TEMP  REFER     BETA MODELFLAG FILENAME
20      1000000   1.5  1
/Users/jbudynk/aas229iris/worksheets/template_fitting/modified-blackbody-templates/modified-black
body-template_20K.dat
```

```
25      1000000   1.5  1
/Users/jbudynk/aas229iris/worksheets/template_fitting/modified-blackbody-templates/modified-black
body-template_25K.dat
```

2. Click on the **Custom Models Manager** icon.



This brings up the **Custom Model Manager**.



We'll fill in the information needed to import the template libraries from top to bottom.

2. Choose a "**Component ID**". This is the name as it appears in the list of Custom Models in the Fitting Tool. This example uses "`modified-blackbody`".

3. Select the "**Template Library**" radio button.

4. Enter the location of the template model index file. You can click "**Browse**" to look for it through your system's file browser, or type in the full path to the file. If you downloaded the workshop repository, the file is located in

```
<path-to-iris>/worksheets/template_fitting/modified-blackbody-template.t
xt.
```

5. Now enter information about the parameters. Each term must be separated by a comma, and must be entered in the same order as they appear in the template model index.

    a. "**Names**" : Add the parameter names. These must match the names of the parameters specified in the index file.

    ```
    temp,refer,beta
    ```

    b. "**Values**" : Add the default initial guesses for each parameter. These must match the values in the template model index file.

    ```
    50,1000000,1.5
    ```

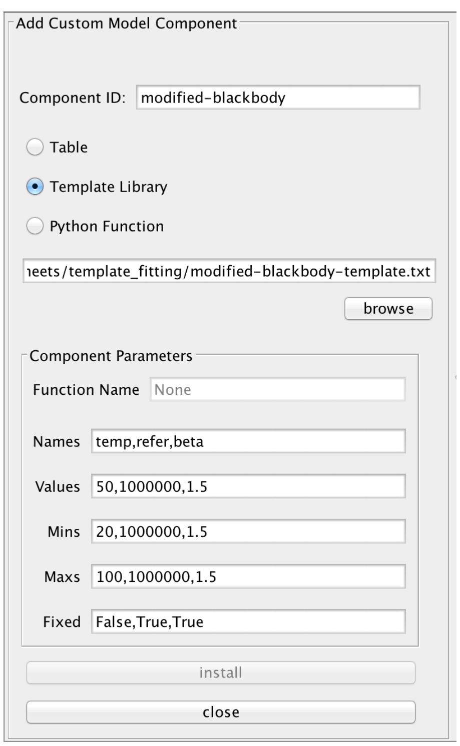
    c. "**Mins / Maxs**" : Add the minimum and maximum values each parameter can be. Again, make sure these are consistent with the values in the template model index file.

    ```
    Mins:  20,1000000,1.5
    Maxs:  100,1000000,1.5
    ```

    d. "**Fixed**" : Finally, state whether the parameters are initially fixed (frozen) or allowed to vary (thawed).

    ```
    False,True,True
    ```

In the end, your import form should look like this:



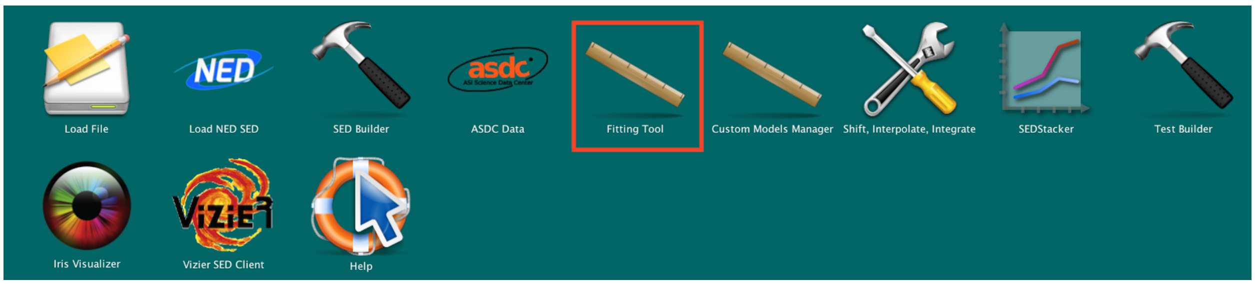
6. Click "**Install Model Component**" when you are finished. This adds the template model library to Iris so you can use it at any other time.

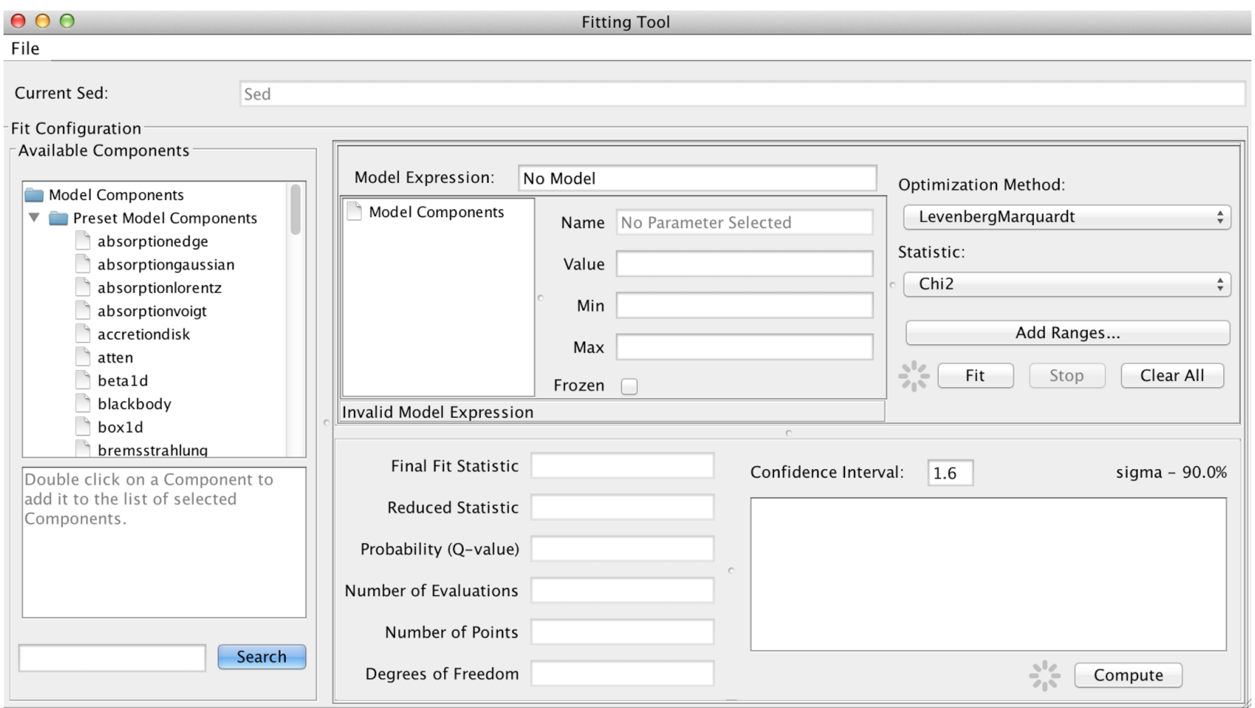
PART 2: Fitting

You will use a dataset of star forming galaxies near redshift z~1 published in [Kirkpatrick et al. (2012)](), which contains galaxies observed from near IR to submm wavelengths in the GOODS-N and ECDFS fields. These sources are modeled well by two modified blackbodies, with the physical interpretation that the cold dust is due to the diffuse interstellar medium heated by the underlying stellar population, and the warm dust is from the star forming regions.

You will use the template library imported in **Part 1** to model this set of star forming galaxies at redshift z~1.

1. Start by opening the **Fitting Tool**.





2. Under "**Available Components**," scroll down to "**Custom Model Components**," and double click on the **modified-blackbody** template library name to add it to the "**Model Components**" (middle panel in the Fitting Tool).

   Because we will model two dust populations, add another modified blackbody component. You should have two components, `m1` and `m2`.

3. Set initial guesses for the temperatures in each model by expanding the model components, highlighting the `temp` parameter, and typing in the initial value in the "**Value**:" field.

   For the warm dust component **`m1`, set `temp=70`**.
   For the cold dust, component **`m2`, set `temp=30`**.

   As temperature is the only variable parameter in the library, we won't modify the other parameters.

4. We also want to measure the contribution or strength of each component, so we multiply each modified black body by a constant model.

   In the list of "**Preset Model Components**", double click on `const1d` twice to add two constant models. You should have two `const1d` models: `m3` and `m4`.

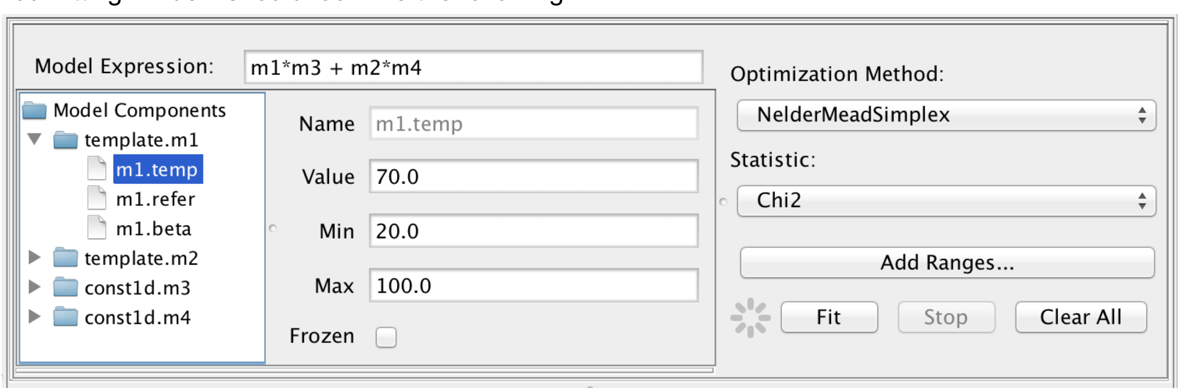
5. You may have noticed that the **Model Expression** field -- the model to be fit to the SED -- automatically linearly combines each model as it is added to your fitting session. This works well when your model is only a linear combination of other terms. In this case, you will need to update the model expression.

   In the model expression field, update the model expression to express a linear combination of the modified blackbodies times a constant model:

   ```
   m1*m3 + m2*m4
   ```

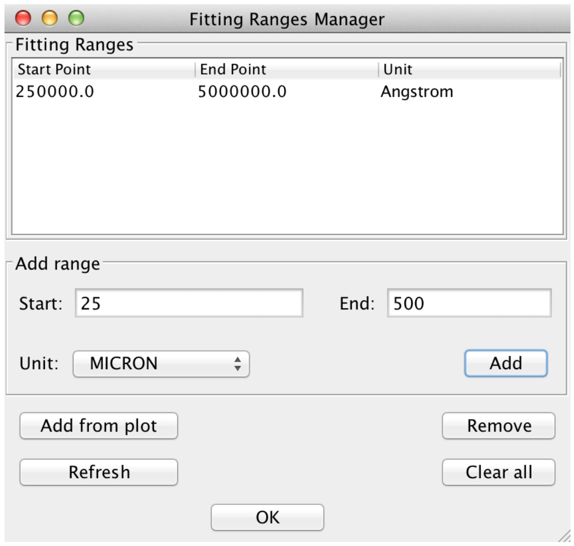
6. With your model expression defined, choose your **optimization** and **fit statistics**. Try using Nelder-Mead optimization (**NelderMeadSimplex**) with the vanilla chi-squared statistics (**Chi2**).

   Your fitting window should look like the following:



7. Next, define a fitting range from about 25 to 500 microns. This will only fit the model to the data within the fitting range.

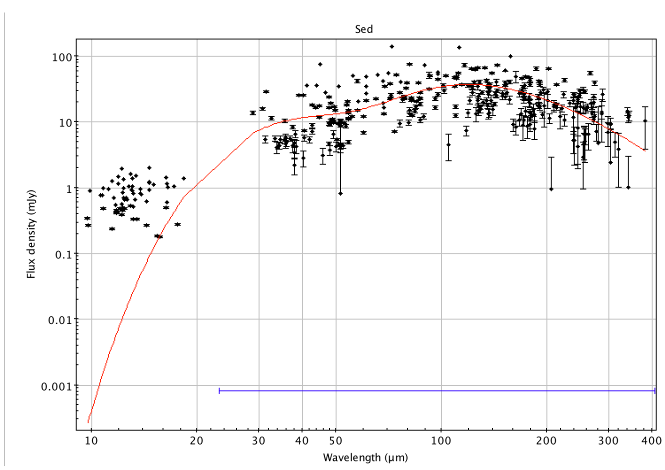
   Click on the "**Add Ranges…**" button, below the "**Fit Statistic**" and "**Optimization**" options. This will open a new window for managing fitting ranges.



You can enter a fitting range via two methods: (1) by specifying the “**Start**” and “**End**” ranges, or (2) by selecting “**Add from plot**” and clicking on the start and end points on the plot itself. The image above specified the start and end ranges with method (1).

8. Click “**Fit**” to fit the SED.

   A red line representing the fitted model is overplotted on the data in the plotter. The model parameter values update to the best fit values, and the fit statistics are shown in the bottom half of the window. The fit results aren’t that great, but there’s lots of scatter in the data, so it is to be expected. However, the results match well with the literature.



9. You can save the results to a human-readable text file and to a JSON file which can be re-read into Iris and evaluated or fit on other SEDs. Go to “**File → Save Json…**” to save the current model in JSON format, and “**File → Save Text…**” to save a human-readable version of the fit results. For example, take a look at the saved JSON and text files `<path-to-iris>/worksheets/template_fitting/z1sf_kirkpatrick-fit_results*`

```
$ cat z1sf_kirkpatrick-fit_results.txt
Iris Fitting Tool - Fit Summary
SED ID: Sed (Segments: 1)

Model Expression: m1*m3 + m2*m4
```

```
Components:
        template.m1
                                    m1.temp =  6.03344E+01
                                   m1.refer =  1.00000E+06 Frozen
                                    m1.beta =  1.50000E+00 Frozen
        template.m2
                                    m2.temp =  2.19262E+01
                                   m2.refer =  1.00000E+06 Frozen
                                    m2.beta =  1.50000E+00 Frozen
        const1d.m3
                                      m3.c0 =  8.49525E-14
        const1d.m4
                                      m4.c0 =  9.41256E-13

Fit Results:
                        Final Fit Statistic =  1.01128E+05
                          Reduced Statistic =  3.21041E+02
                      Probability (Q-value) =  0.00000E+00
                         Degrees of Freedom = 315
                                Data Points = 319
                       Function Evaluations = 1165

                                  Optimizer = NelderMeadSimplex
                  Statistic (Cost function) = Chi2
```