Temperature Map Pipeline

Carter Rhea

October 15, 2019

1 Temperature Maps

If you would like to simply run the program, please see README.md for details!

Before we can get to the temperature map, we first have to actually use the data we got from the Weighted Voronoi Tessellation Algorithm; we need the pixels binning information. Currently the user can choose to initially apply a WVT binning algorithm.

1.1 Intermediate Step: Binning the Data

As this subsection's title suggests (look up), we must first go through the process of binning the data (our fits file) based of the recommendation of our WVT algorithm.

Instead of describing this in great detail, I will simply include an algorithm outlining the process and point the reader toward the python file (*Bin_data.py*) for more in depth analysis.

```
Data: WVT Bins and Fits File
Result: Binned PHA files
initialization −− > read in WVT bin information;
for bin do
    for pixel in bin do
        Create fits file for pixel;
        Generate PI/PHA file;
    end
    Combine Pixel's PI/PHA files;
end
```

Algorithm 1: Binning Algorithm

Clearly the implementation is a little more complicated and uses several *CIAO* tools such as *specextract* and *dmextract*. For more information on these wonderful tools, check out the *CIAO* website:

http://cxc.harvard.edu/ciao/

1.2 Generating Temperature Maps

We can now (FINALLY) use our combined pixels to generate a temperature map assuming you have already created a binning of some sort – I would suggest my Weighted Voronoi Tesselations algorithm[1]. We will be employing *XSPEC*[2] for our modeling needs.

[1] https://github.com/crhea93/WVT

[2] https://heasarc.gsfc.nasa.gov/xanadu/xspec/

2 The Pipeline

Data: Binning Map
Result: Temperature Map and Graphic
Step 1: Create binned spectra using *binned_spectra.py*. ;
Step 2: Run XSPEC Model on binned data by "simply"[a] running *Temperature_Fits.py*. ;
Step 3: Plot the temperature map with *Temperature_Plot.py*.;

Algorithm 2: Temperature Map Pipeline

[a] Like fitting a model is simple..... But seriously this is where a mistake could be made resulting in incorrect Temperature Profiles so play around with the model on a *single* observation to ensure it is a decent fit...

The algorithm itself is fairly succinct because all of the mechanics are intentionally under to hood. For instance, we actually create what I deem as super-pixels in the creation of the spectra for each bin since `specextract` runs faster with less regions. The super-pixel algorithm groups as many pixels together to create a box-region to reduce computational time[3]

To run this code, we simply need to supply an input file to `Temperature_Maps.py`
Below is an example input file

```
#----------INPUT DATA------------#
base_dir = /home/user/Documents/AstronomyTools/Tests
Name = NGC4636
ObsIDs = 323,324
WVT_data = Merged_unbinned/WVT_data.txt
source_file = center
output_dir = binned/
#----------FIT INFO--------------#
redshift = 0.003129
n_H = 1.91e-2
Temp_guess = 1.0
#----------CHOICES---------------#
bin_spec = True
num_bins = 0 #Unnecessary if bin_spec == True
fit_spec = True
plot = True
Colormap = jet
```

[3] The mechanics for doing this are well commented in the *binned_spectra.py* file.

3 Dealing with Merged Data Set

How do we create temperature maps for merged data sets? As documented on the `CIAO` site [4], we are not able to create spectra from merged observations. Thus we have to be very careful in the creation of temperature maps for merged data sets.

The basic steps are as follows:

1. Merge Observations (Don't worry! We are just going to use the event file and exposure map).

2. Create a bin map using your favorite binning algorithm; I suggest using Weighted Voronoi Tesselations.

3. For each individual observation do the following:

   (a) Create individual event files for each bin

   (b) Extract spectrum for each bin

4. Run fitting software on each bin by fitting all datasets simultaneously

In following these steps, we are able to reconstruct a temperauture map for a merged observation while avoiding the inherent drawbacks of merging data.

[4] http://cxc.harvard.edu/ciao/caveats/merged_events.html

4 NGC 4636

For an example, I will continue with the cluster `NGC 4636` which I used to demonstrate the efficacy of my Weighted Voronoi Tessellations algorithm except this time we will be using the entire CCD on which the cluster is located and wont be doing any background/exposure subtraction.

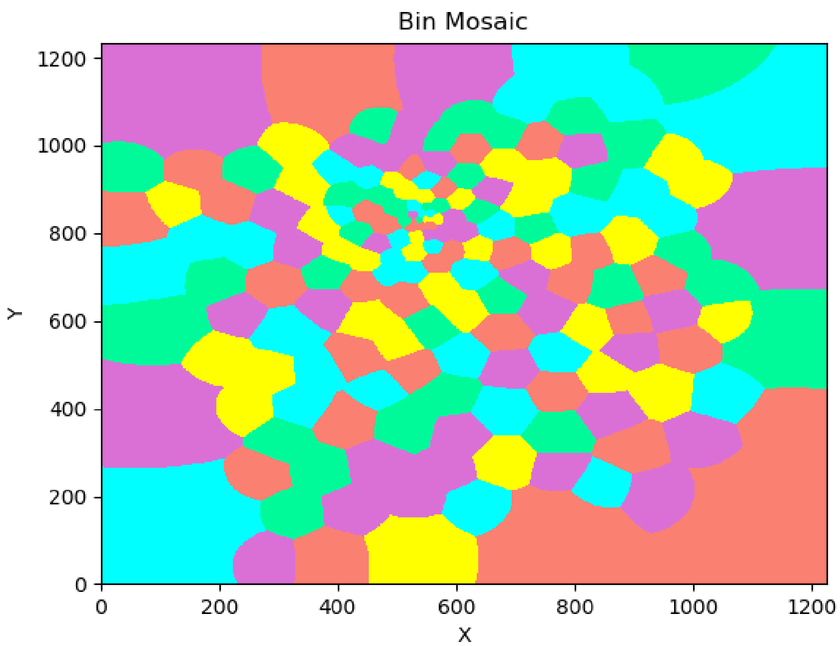


Figure 1: The final WVT map for `NGC 4636` with 177 bins in image units.

The next step involves running the creation of spectra for each bin. I have included the input parameters in case anyone wants to reproduce the results :)

Finally we are able to fit the data! I opted to simply used an absorbed apec model from *Xspec* through *Sherpa*. The model was fit simultaneously to both observation Ids (323 and 324) with:

$$set\_source(1, xsphabs.abs1 * (xsapec.apec1))$$

$$set\_source(2, abs1 * (xsapec.apec2))$$

and the column density nH of the absorber was set to $1.91e^{20}cm^3$ which was taken from the NASA HEASARC NH Tool[5].

And now we get to plot our temperature map!!

In comparing this temperature map with that from Diehl and Statler 2005[6], we can see that the map is well-recreated!

[5] https://heasarc.gsfc.nasa.gov/cgi-bin/Tools/w3nh/w3nh.pl

[6] https://arxiv.org/abs/astro-ph/0512074

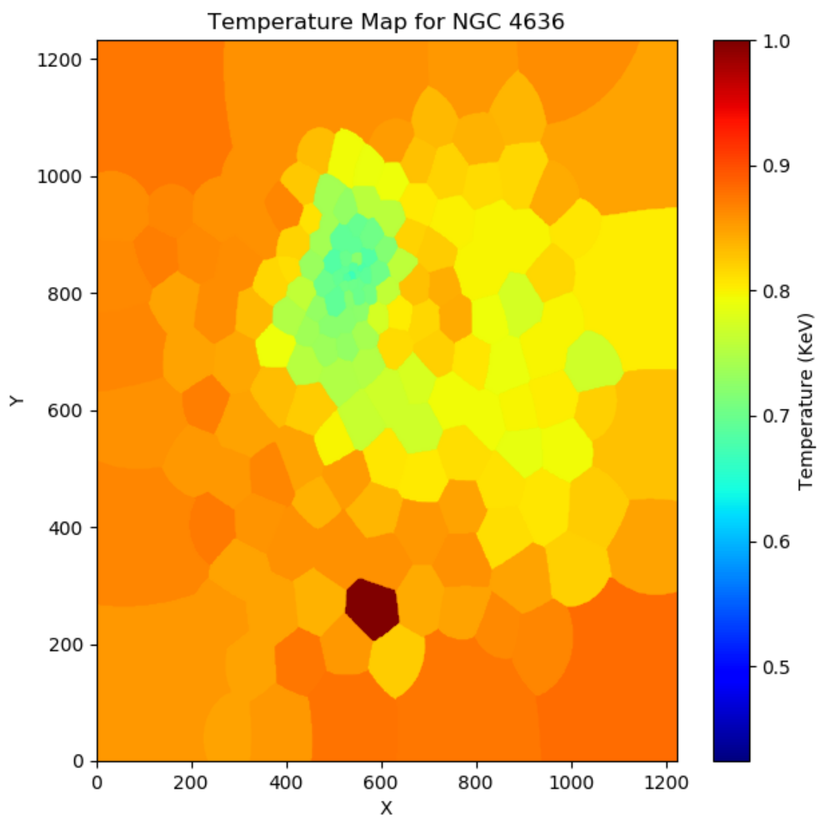
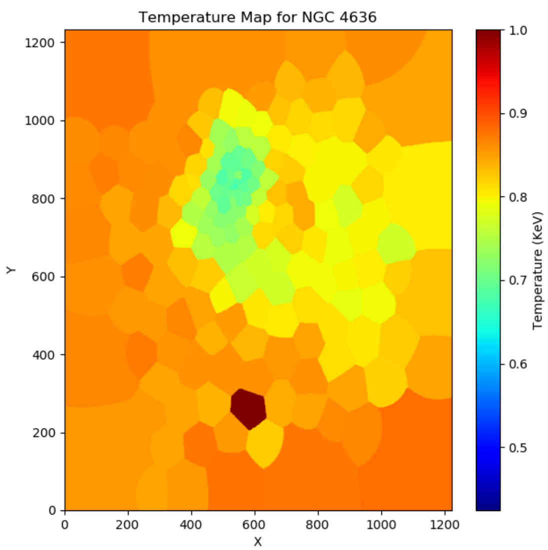
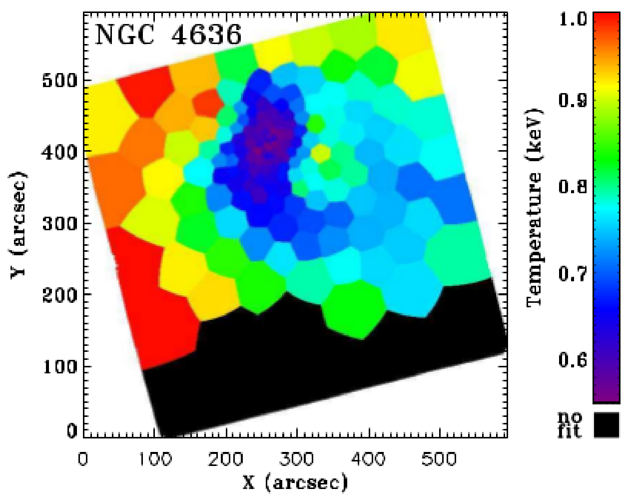


Figure 2: Final Temperature map for NGC4636 again in image units.



(a) New Temperature Map



(b) Diehl and Statler 2005 Temperature Map

Figure 3: Comparison of the two temperature maps