**/extra/wayne1/preserve/stevenc9/Color-Gradient**

**Main.py:**

This is the driver code that combines everything from **inputFiles.py** and **createStructs.py** (along with some extra code for plotting which should probably be separated into its own file). The overall flow is:

1. Read in the TSV in the R waveband which holds info about the shape of the galaxy, most importantly info about the majorAxis, rotation radians, and minor/Major axis ratio.
2. Read in the clusMask for the first waveband (could be the second one, doesn’t matter for now). Gets all of the distinct arms, put their pixel positions in respective sets, within a containing dictionary
3. Look through this dictionary and gets the largest arm. This doesn’t get the “longest” arm, just the one that has the most pixels.
4. Creates a list of objects, where each object corresponds to a major axis length (in pixels), ranging from 5 pixels, to the semi-major axis (majorAxis/2). It calculates the corresponding semi-minor axis using the minor/major ratio and creates an ellipse from this resulting info, along with an initial rotation radians. From here it checks for overlap with the arm, and generates an arc, which will be used for all later plots.
5. Loops through the created list of “ellipse objects” from 4, first makes the ellipse had some overlap with the arm. Then it removes any duplicate data. For example, if thetas (24,25,26) result in the same pixel location, and therefor the same data point, it just takes the median theta, in this case 25. This happens in **remvSimThetas().**
6. These two use the resulting “cleaner” info, and open up the corresponding FITS images, and takes the respective info from the given theta/pixels of the arc.
7. The collected info on theta/pixels of the arcs are then used in the rest of code, which is just a bit of plotting and formatting the plots.