Yeast Showdown 1.5: Cell Segmentation

In Yeast Showdown part I, we analyzed microscope images of yeast cells bearing wildtype (RFP-labeled) and mutant (YFP-labeled) alleles of human autism genes. To compare the effect of the alleles on the cells, we fused a downstream promoter to CFP to serve as a readout of glycine metabolism (see *Yeast Showdown I* for experimental setup).

We used MATLAB to obtain the average CFP pixel intensity for each of the two types of cells, but here our analysis stalled because there was no obvious way to assign a significance to any difference we saw between wildtype and mutant. (*Why is this? What are some possible problems with comparing pixels? Is there another analysis that might be better?*)

A more natural way to quantify the data is to find the intensity of CFP fluorescence *per cell*. This requires segmenting, or determining the boundaries, of the yeast cells. Automatic segmentation is an extremely interesting—and difficult—research problem at the interface of computer science and biology. However, as you will see in this exercise, with just a basic bag of tricks you can get surprisingly far on segmentation problems that are by no means trivial.

We will start by incorporating functions into your code from part I; this will minimize code repeat and make it easier to read. Then we will determine the total CFP fluorescence of each cell in the images by a simple thresholding segmentation, and use the resulting fluorescence distributions to determine whether the mutant allele has quantitatively affected glycine metabolism in yeast.

1. **Write a function to normalize images, and use it in your code.** Remember how in part I you repeatedly needed to convert an image to double-precision numbers and normalize it before you could display it? Wouldn't be nice if you could simply do this in a single command? Here is your chance to make life easier for yourself. This function should take a raw image as an argument and return a fully double-converted, normalized (i.e. contain values from 0 to 1) image.

2. **Segment the images and obtain the CFP-fluorescence distributions for both wildtype (RFP-labeled) and mutant (YFP-labeled) cells.** There will be many different conceptual strategies for doing the segmentation, but the code you write will follow the same general outline: do some transformations on the image, convert it to binary (1 for cells, 0 for background), convert the binary image to a label matrix, and then use the label matrix to extract data from the raw CFP image. Before you write any code, think about what commands you will need for each part of the process, what data you will get out at the end, and how that data will help you answer the next question. Also keep in mind that segmentation is often imperfect; you may need to accept a small number of unsegmented (not counted) or oversegmented (counted as more than one) cells to be able to correctly segment most cells.

3. **How different is the CFP signal between wildtype and mutant cells? Use bootstrapping to determine a significance value for this difference.** What is your null hypothesis? How sensitive is this step to mistakes in segmentation from the last round?