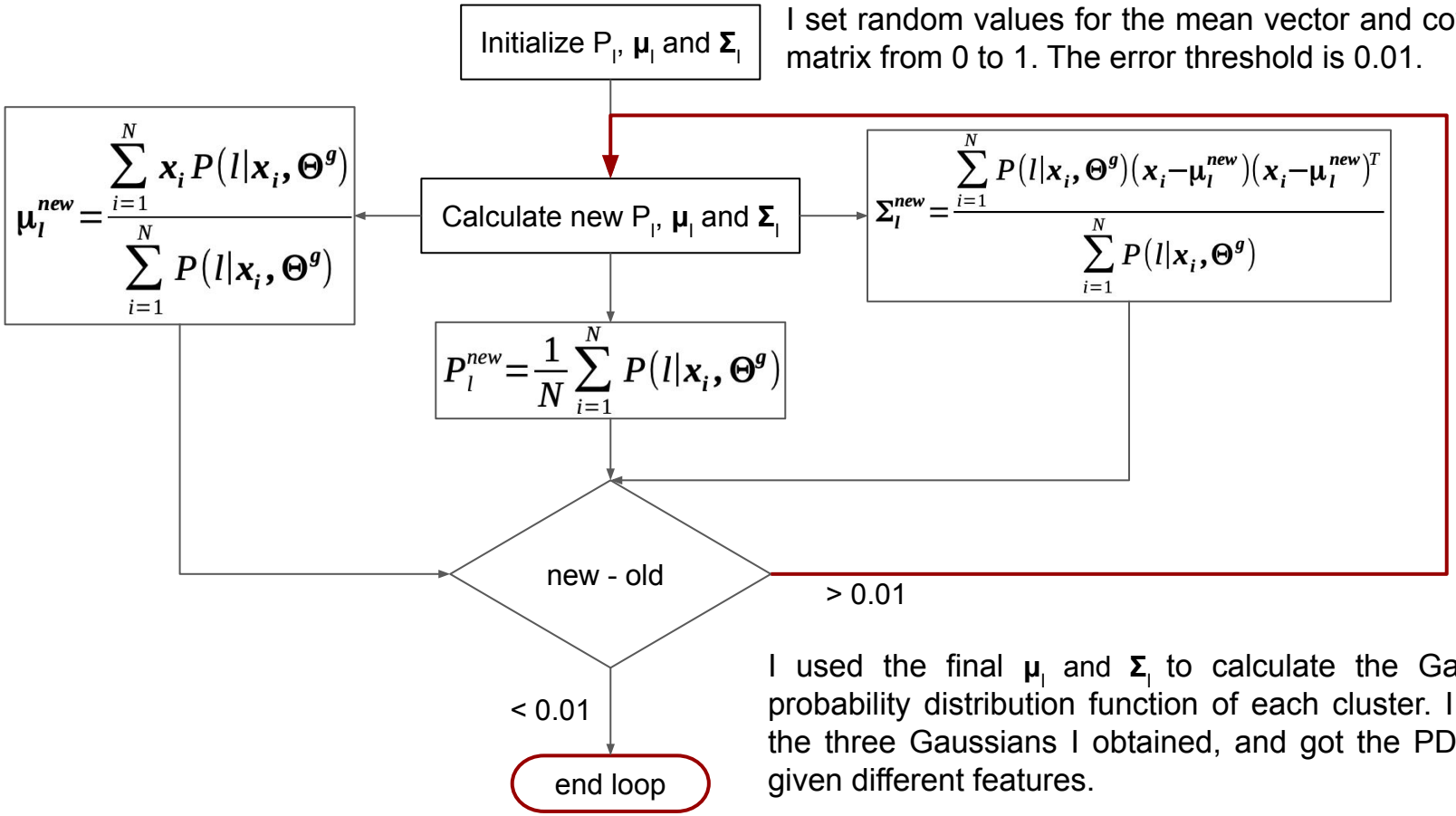
Activity 15 - Expectation Maximization

de Castro, Crizzia Mielle | 2015-08076

Expectation Maximization Algorithm

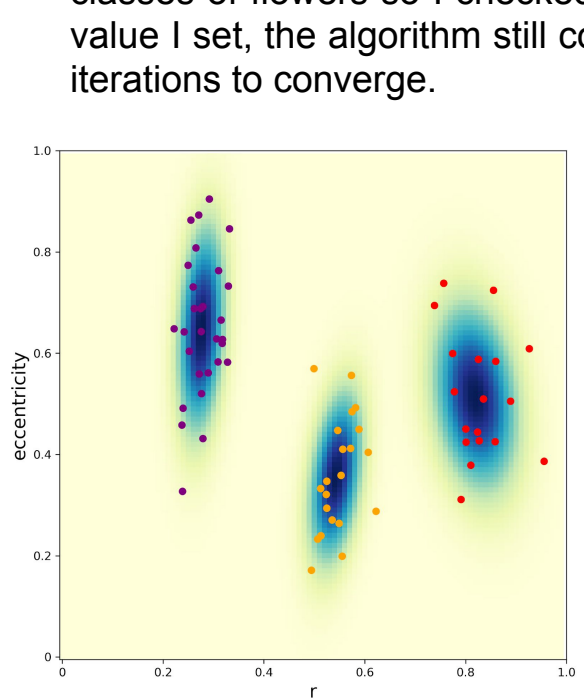


I set random values for the mean vector and covariance matrix from 0 to 1. The error threshold is 0.01.

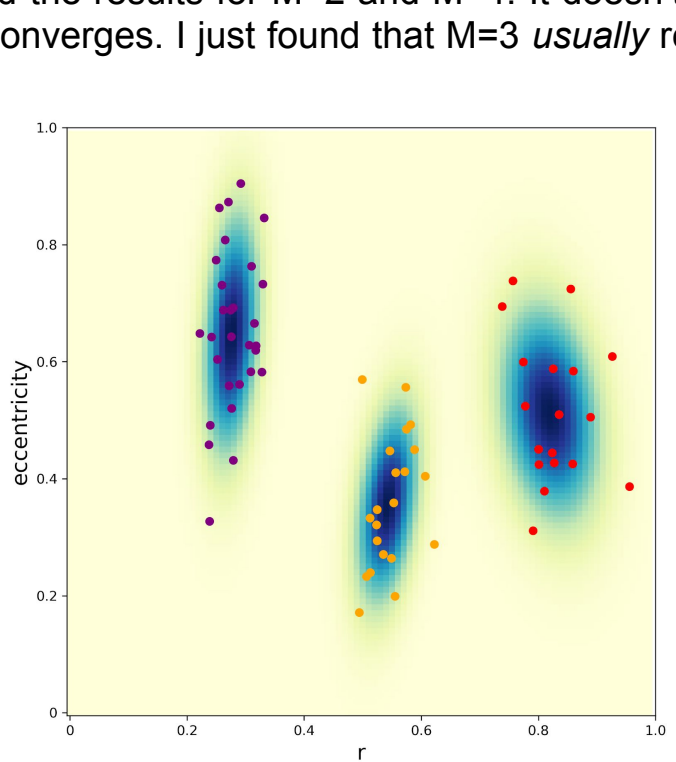
I used the final μ_l and Σ_l to calculate the Gaussian probability distribution function of each cluster. I added the three Gaussians I obtained, and got the PDF plots given different features.

Probability Distribution Function: eccentricity vs r

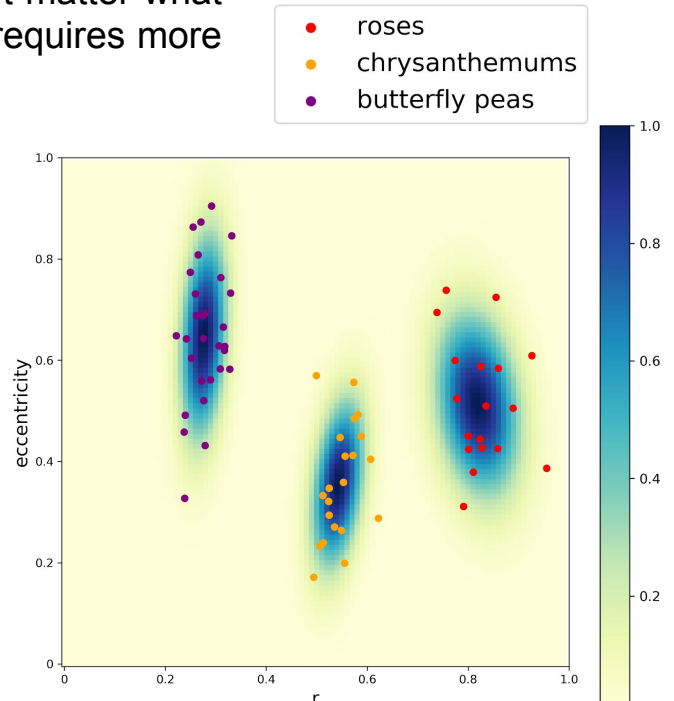
I tried to check what would happen for different prior probabilities 1/M. I have three classes of flowers so I checked the results for M=2 and M=4. It doesn't matter what value I set, the algorithm still converges. I just found that M=3 *usually* requires more iterations to converge.



M = 2
iterations = 25



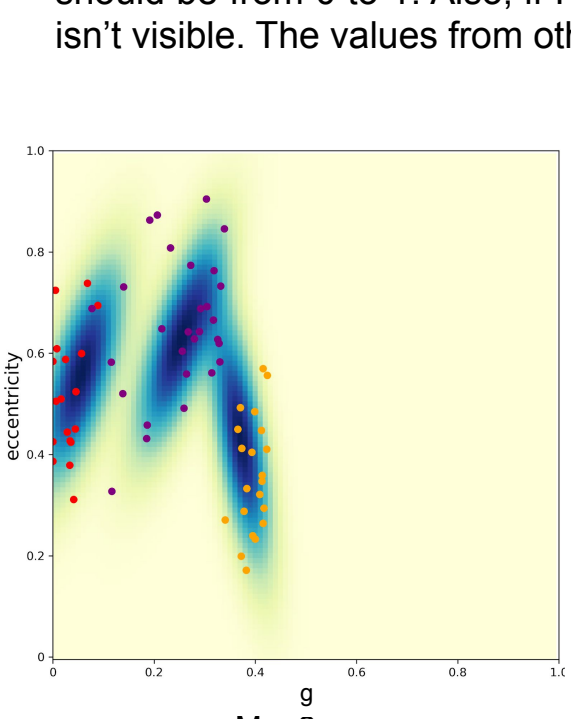
M = 3
iterations = 72



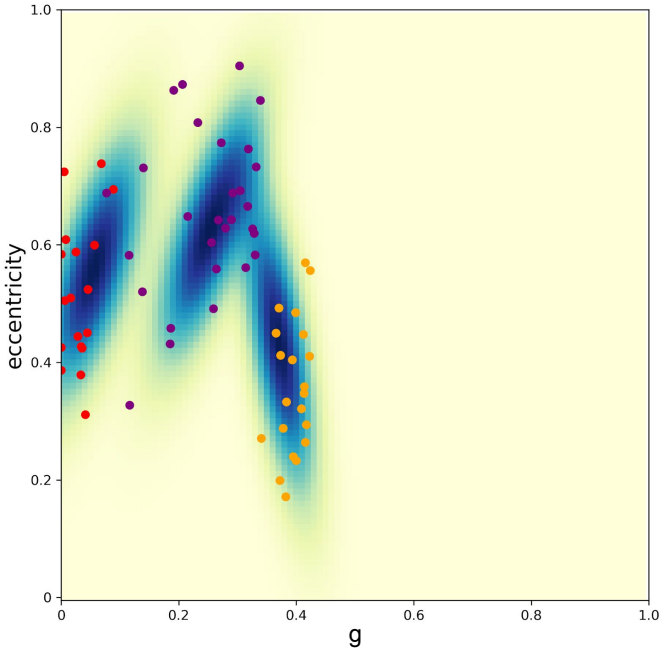
M = 4
iterations = 34

Probability Distribution Function: eccentricity vs g

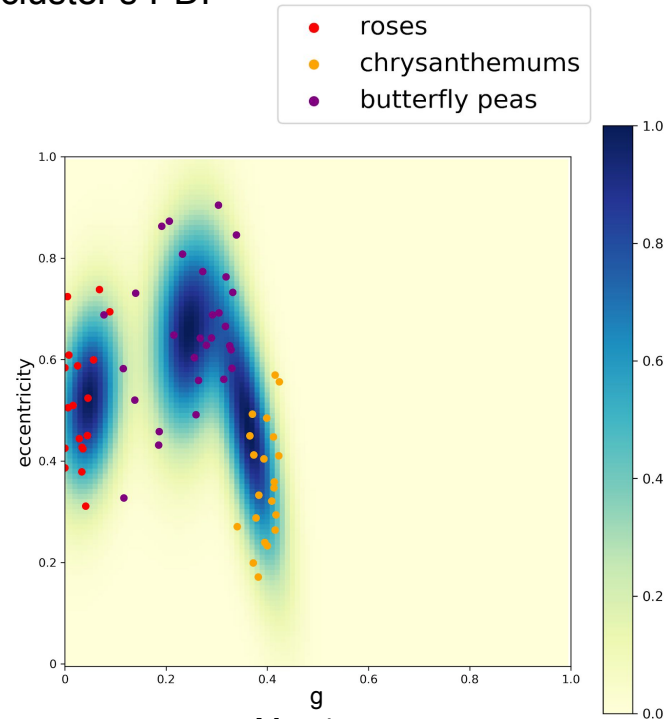
I also normalized each cluster's PDF with respect to itself, since the probabilities should be from 0 to 1. Also, if I don't normalize the PDFs, some of the cluster's PDF isn't visible. The values from other clusters is relatively much higher.



M = 2
iterations = 160



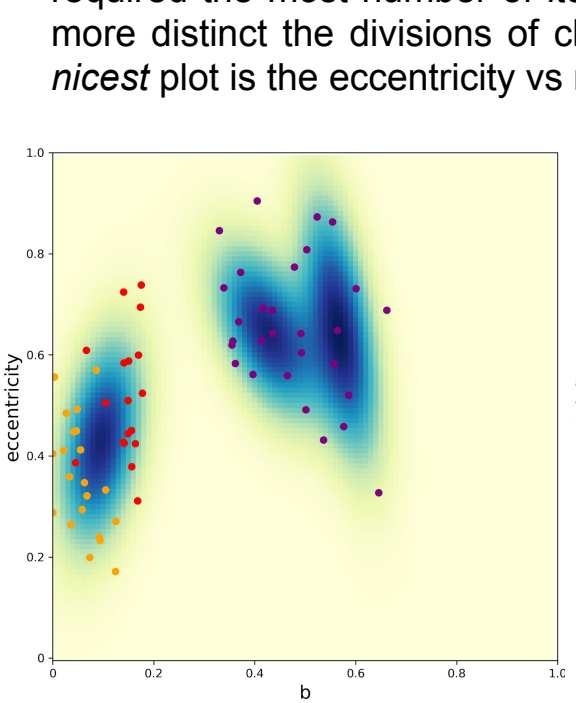
M = 3
iterations = 162



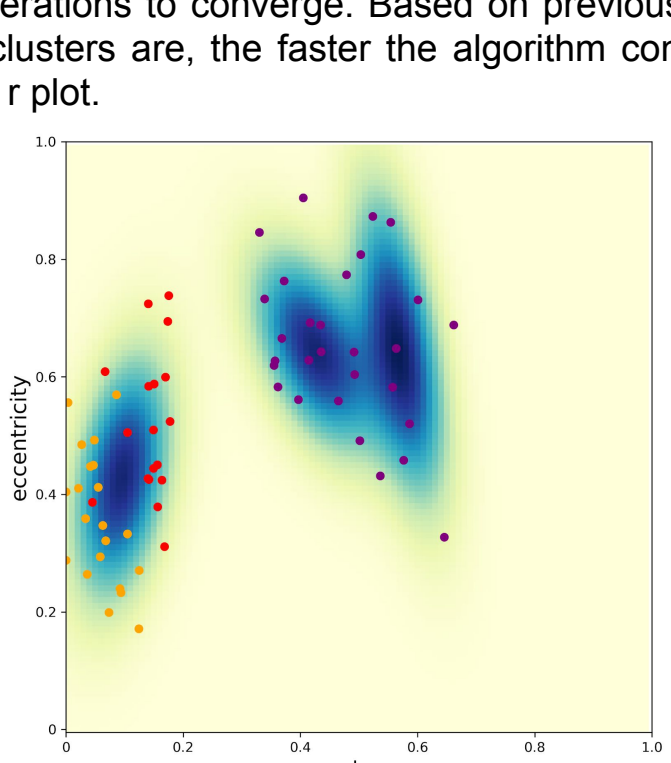
M = 4
iterations = 127

Probability Distribution Function: eccentricity vs b

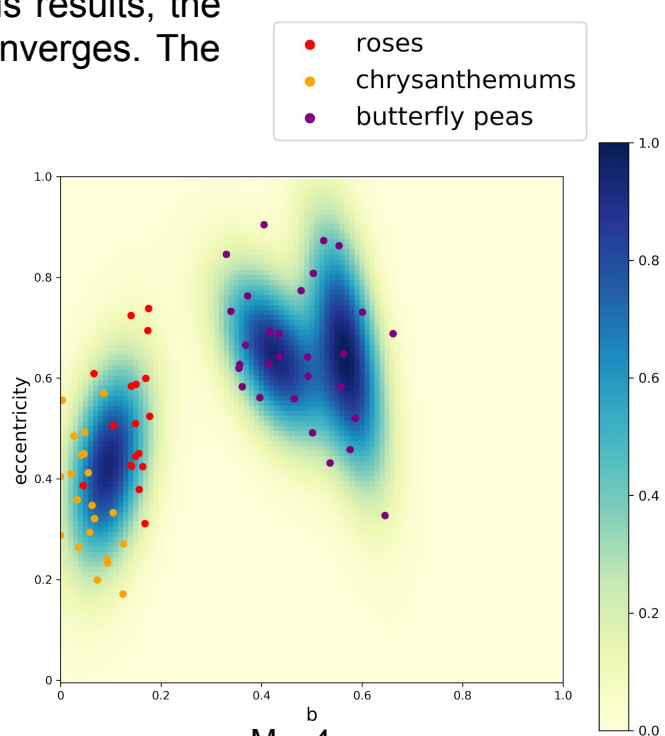
The algorithm was not able to cluster together properly for eccentricity vs b. It also required the most number of iterations to converge. Based on previous results, the more distinct the divisions of clusters are, the faster the algorithm converges. The *nicest* plot is the eccentricity vs r plot.



M = 2
iterations = 218



M = 3
iterations = 214



M = 4
iterations = 216