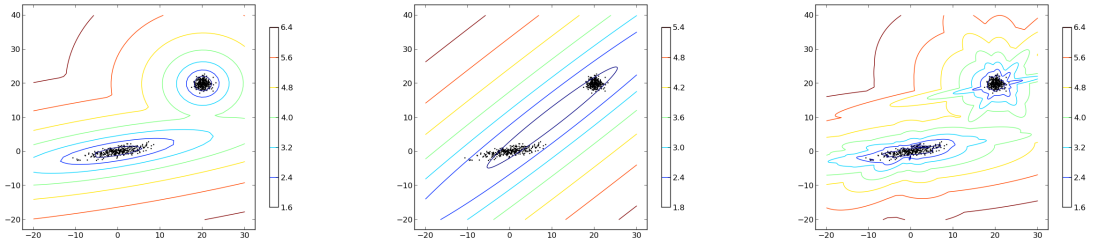
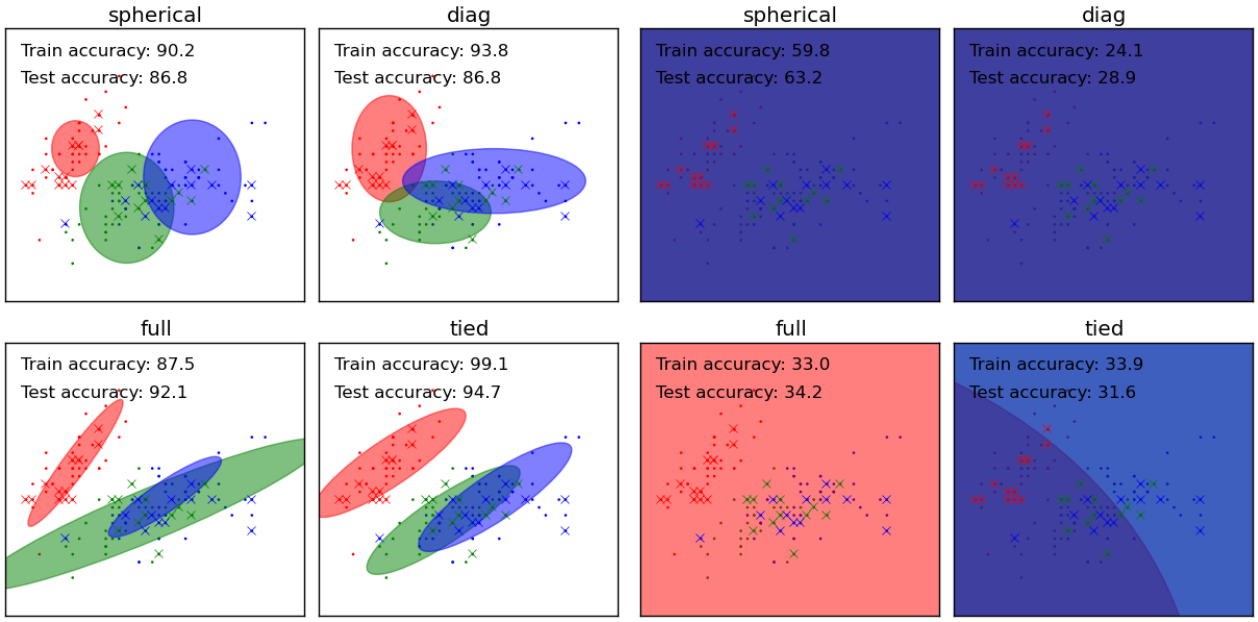


(a) Original figure, 2 components (b) only 1 component (c) 30 components for fun

Figure 1: Density estimation for a mixture of Gaussians (2, 1, 30 components)

1 Summary of 4.1 Guassian mixture models

- The first example generates random data with a mixture of gaussians at two different points. It's pretty simple, and most of my confusion comes from numpy and array shaping / concatenating things. See Figure 1 for examples with a few different number of components for model.
- Next, I followed the Gaussian Mixture Model (GMM), using the Expectation-Minimization (EM) method on the iris data set (using all 4 dimensions, showing only two of those dimensions. I was confused by the method at first, because it seemed supervised, and then I realized it is partially supervised. The code for testing the four different covariance constraints, along with making the ellipses is quite complicated. I learned a lot, but still wouldn't be comfortable writing that from scratch. Maybe I'll never have to, because there are so many examples to work from. See Figure 2a for replication of the tutorial.
- For fun, I attempted to use the DPGMM model on the iris dataset. This sort of prooved the above point, that it's very easy to adapt the example code. Although, for many different alpha choices, nothing could come close. This is either because I'm using DPGMM incorrectly, or, I think more likely, because unsupervised DPGMM will never work on the iris data set. See Figure 2b for the plot of failed attempt.



(a) GMM classification, partially supervised

(b) DPGMM attempt (failed)

Figure 2: Partially-supervised GMM of iris test data and failed attempt at using Dirichlet Process (DP) GMM on the iris data. This may make sense, since iris data are so overlapped?