Homework Set Five Solutions

ECE 175
Department of Electrical and Computer Engineering
University of California, San Diego

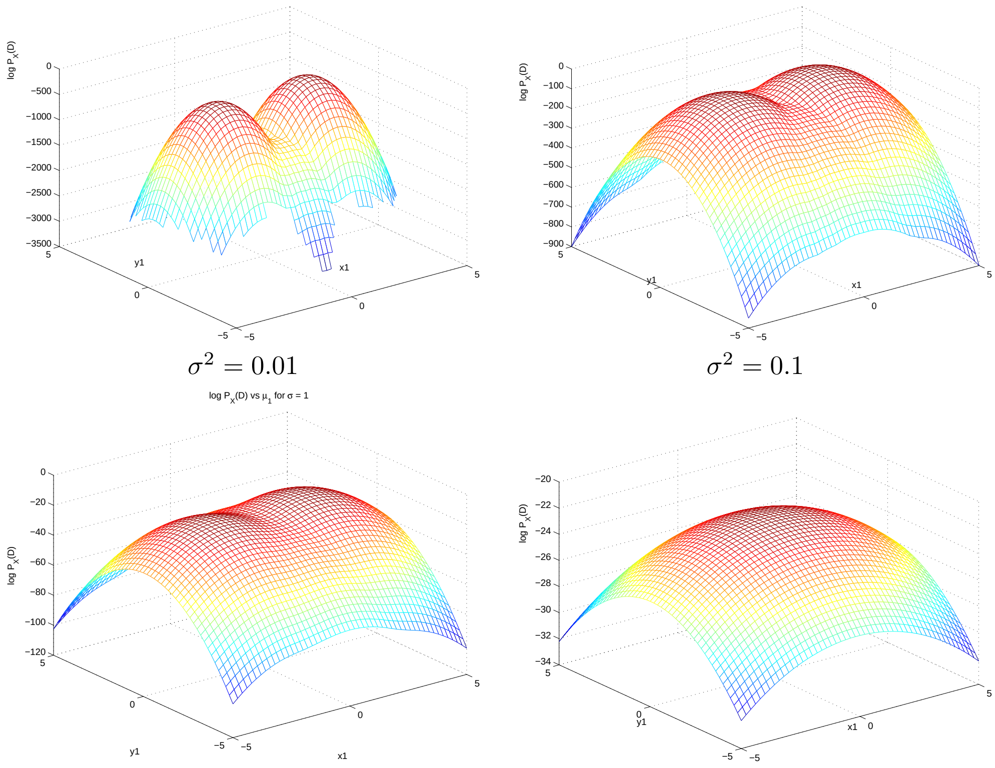
Nuno Vasconcelos — Spring 2008

1. This follows from

$$
\begin{aligned}
E_{\mathbf{X}}[\mathbf{x}] &= E_Y\{E_{\mathbf{X}|Y}[\mathbf{x}|i]\} = \sum_{i=1}^{C} \pi_i \mu_i \\
E_{\mathbf{X}}[\mathbf{x}\mathbf{x}^T] &= E_Y\{E_{\mathbf{X}|Y}[\mathbf{x}\mathbf{x}^T|i]\} = \sum_{i=1}^{C} \pi_i (\mathbf{\Sigma}_i + \mu_i \mu_i^T) \\
\mathbf{\Sigma}_x &= E_Y\{E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_x)(\mathbf{x}-\mu_x)^T|i]\} \\
&= E_Y\{E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_i+\mu_i-\mu_x)(\mathbf{x}-\mu_i+\mu_i-\mu_x)^T|i]\} \\
&= E_Y\{E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_i)(\mathbf{x}-\mu_i)^T|i] + (\mu_i-\mu_x)E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_i)^T|i] \\
&\quad + E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_i)|i](\mu_i-\mu_x)^T + (\mu_i-\mu_x)(\mu_i-\mu_x)^T\} \\
&= E_Y\{E_{\mathbf{X}|Y}[(\mathbf{x}-\mu_i)(\mathbf{x}-\mu_i)^T|i] + (\mu_i-\mu_x)(\mu_i-\mu_x)^T\} \\
&= E_Y\{\mathbf{\Sigma}_i + (\mu_i-\mu_x)(\mu_i-\mu_x)^T\} \\
&= \sum_{i=1}^{C} \pi_i[\mathbf{\Sigma}_i + (\mu_i-\mu_x)(\mu_i-\mu_x)^T]
\end{aligned}
$$

2.

a) The surface plots are presented in the figure below. Note that there are two local maxima and the likelihood function is symmetric. This is a characteristic of all parameter estimation problems involving mixtures. In this case, $\{\mu_1, \Sigma_1, \pi_1\}$ can be the parameters of either of the Gaussians, calling one component 1 and the other component 2 is really just a question of convention. This is reflected in the likelihood surface that presents two identical local maxima for μ_1, co-located with the means of the two Gaussians. As far as ML is concerned, the two are equally good solutions. We thus see that a problem with C components will have at least $C!$ equivalent solutions and a likelihood surface with $C!$-fold symmetry. It turns out that this is not a big problem, since these solutions are indeed equivalent unless we care about a specific ordering of the mixture components. Typically we don't. There are usually also other local maxima or saddle points, and those can create problems. One such point is visible for the smaller values of σ, in between the two main bumps. Algorithms like EM can, and usually do, get trapped in such local optima.

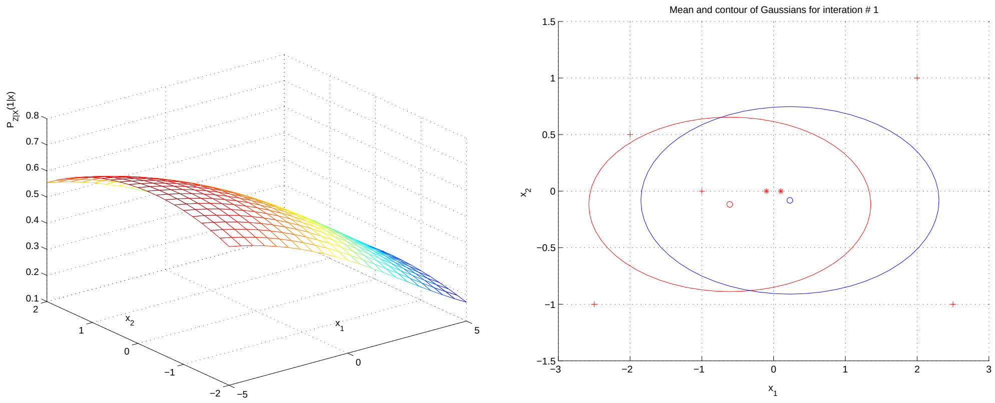


$\sigma^2 = 0.01$ $\sigma^2 = 0.1$

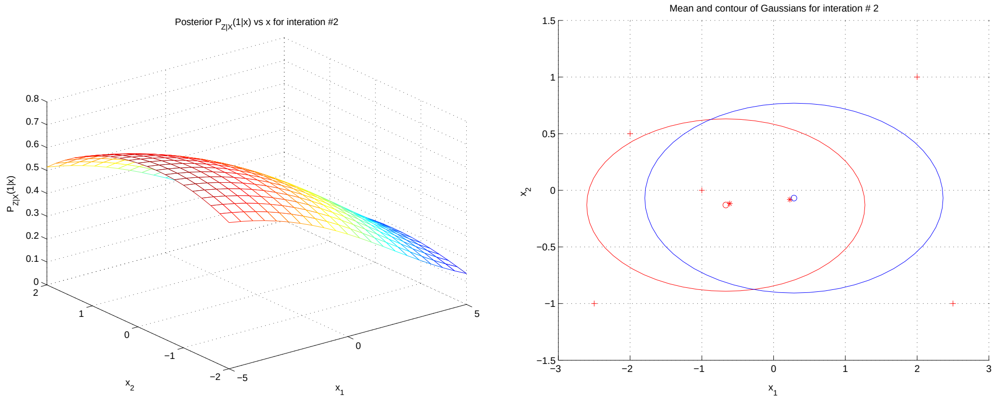
$\sigma^2 = 1$ $\sigma^2 = 10$

Regarding the role of σ, we see that it basically controls the smoothness of the likelihood function. For very small σ's the function is quite bumpy, becoming much smoother as σ increases. Finally, we see that convergence to a good solution is going to depend a lot on the initialization. If, for example, we had started from the location of the saddle point (visible when σ is small) an algorithm like EM might just get stuck in that initial point. In general there might be multiple local optima that are not close to the global optimum, and a good initialization can be critical.

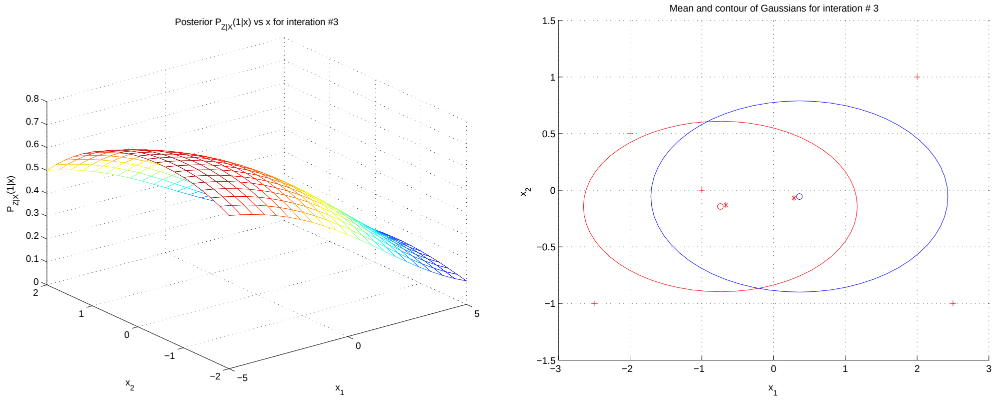
b) The plots of the posterior surface and the Gaussian estimates for the first three steps as well as after convergence (13 iterations) are shown in th next page. The surface plots are shown on the left



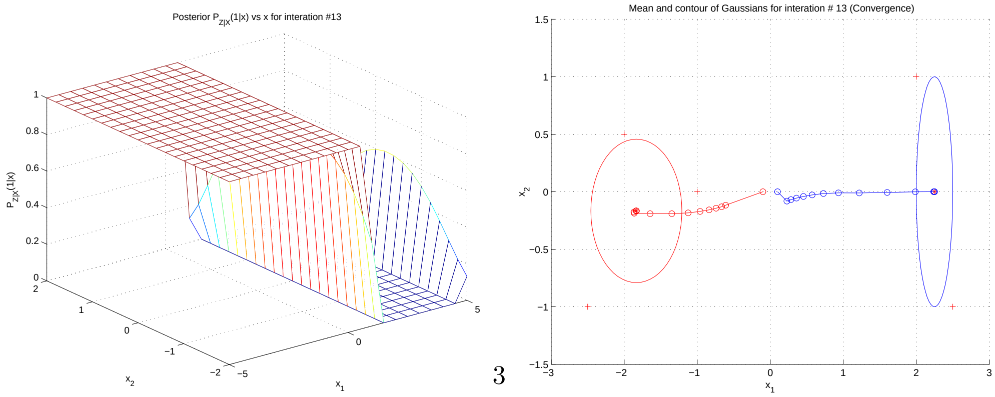
Iteration 1



Iteration 2



Iteration 3



Iteration 13

column, while the Gaussian estimates are shown on the right. In the latter plots, the starting parameter estimates (for the iteration) are shown as ’*’, while the final parameter estimates are shown as ’o’. The last plot also shows the path traced by the mean estimates across iterations. A few observations can be made from these plots. First, the class assignments start very soft, becoming increasing harder, and are quite hard at convergence. The assignments are soft in the regions where the posterior is not 0 or 1, which means that there is a reasonable probability that points in those regions will be assigned to either of the classes. In the first three iterations this is true for the entire region covered by the plots. On the other hand, only points very close to the class boundary have soft assignments after convergence. Notice that this is very different from the sequence of assignments made by a greedy algorithm that assigns each point to only one of the classes at each iteration. In this case, because the classes are very separated, the result of such an algorithm would not be very different from the EM solution. However, when there is significant overlap between classes, greedy can be highly suboptimal.

A second interesting point is the fact that convergence seems quite slow. In fact, after the first iteration, the progress within each iteration is quite small. This is a consequence of the soft assignments: because a point from the Gaussian on the right has a non-trivial probability under the Gaussian on the left, it also has a non-trivial contribution to its parameter updates (remember that e.g. the new mean is a weighted mean of all points, each point weighted by the its probability under the corresponding Gaussian). Hence, the points from the Gaussian on the right pull the parameters of the Gaussian on the left away from their true values, and vice-versa. The result is a slower convergence than that of a greedy algorithm based on hard assignments. This is the price to pay for the optimality of the EM solution.

3.

1. There are high chances that even in the first iteration there were empty clusters. One way to tackle this is to initialize the means ’intelligently’ as we do for the subsequent parts. Another possible solutions is to remove the mean corresponding to the empty cluster and reinitialize the mean with a perturbed version of that of the most populated cluster.

2. Randomly chosen images as means : Image ids [4191, 98, 3406, 1897, 4159, 2514, 3547, 2144, 1523, 948]

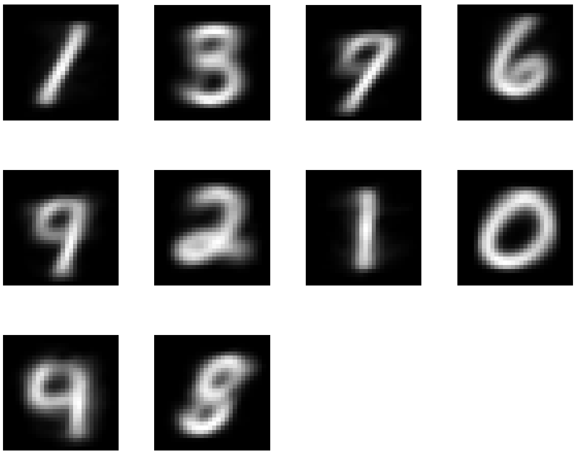


Figure 1: Class means for a random set of initializations

3. We notice that the Gaussian classifier performs badly when we choose the final kmeans classes as

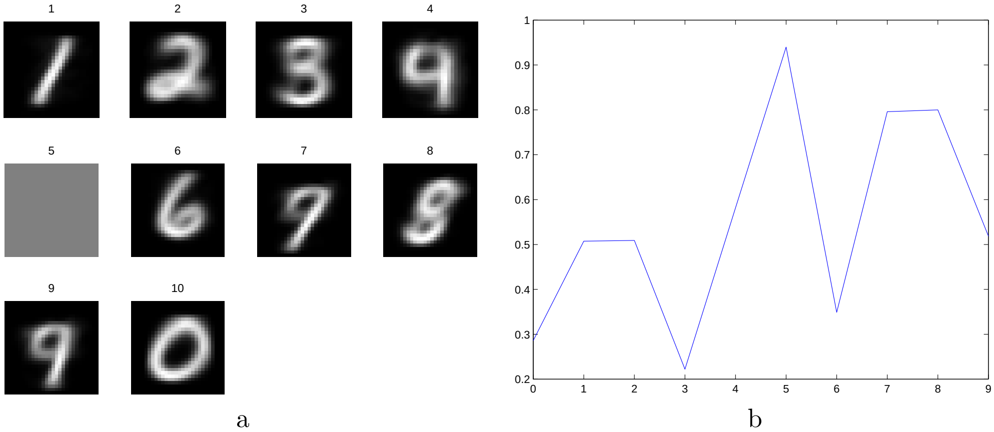


Figure 2: a)Assumed Gaussian means b)Error rates

the original class means. Remember that if we do not have any labels to begin with, this is much better than doing it at random which will on average be wrong 90% of the times. Total error rate is 0.55.

Table 1: Probability of error for each class

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0.28	**0.50**	**0.50**	**0.22**	**0.58**	**X**	**0.34**	**0.79**	**0.80**	**0.51**

4. The final class means are sensitive to the initializations. The kmeans finds the local maxima and hence proper care should be taken while doing the initialization step. The second random initialization gave us two zeros as means. Image ids: [3636, 1546, 4192, 2840, 1852, 3514, 2733, 2224, 3473, 3107]

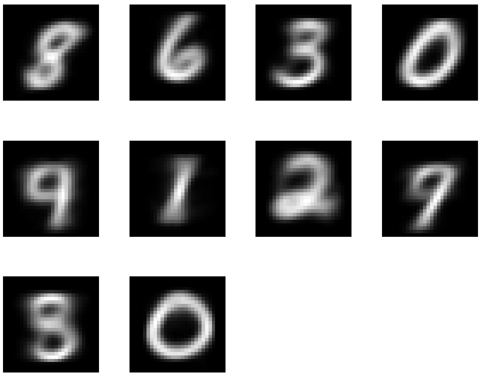


Figure 3: Class means for another random set of initializations