Statistical Learning Example Examination Questions

1. Suppose we get some training data $(X_1, Y_1), \ldots, (X_N, Y_N)$ and we decide to fit the data using a support vector machine (SVM) with the RBF (radial basis function) kernel with fixed parameter $\gamma = 1$ and cost function $C = 1$.

   - Suppose we observe a good fit, say the 0/1-error on the training set is 0.01. How do you think the learned SVM will predict test data from the same source? Do you expect a lot of overfitting to take place? Explain your answer (HINT: see Section 12.3.2)
   - Now suppose we do not observe a good fit for $\gamma = 1$, but by playing around, we see that the fit is quite good for $\gamma = 0.02$. So we predict future data with $\gamma = 0.02$. What can we expect from the test error? Is there a sounder way to optimize γ?
   - Suppose again that the fit for $\gamma = 1$ is not very good. Rather than playing with γ, we decide to use a different kernel. People have tried many different kernels in the literature. Suppose we try 100 different kernels on the training data (without playing around with parameters). Suppose that one of them fits the data really well. So we take this one to predict future data. What can we expect from the test error?

2. *Semi-Supervised Learning* Consider training data for a classification problem, where the data consists of pairs (X_i, Y_i), where Y_i is the (categorical) class variable. In many practical settings, it is hard to obtain Y_i-values; either they are expensive or they are not available at all. On the other hand, obtaining unlabeled X_i-values may be very easy or cheap. Examples are spam filtering (X_i are email messages, $Y_i \in \{\text{SPAM}, \text{NO-SPAM}\}$): most Internet users have received 10000s of emails, so the amount of X-data available is large. However, the user has to specify by hand for each email whether or not it is spam, and he/she is not likely to be willing to do this for more than a few 100 emails. Another example is automatic identification of objects on pictures (encoded as vectors of pixel grey values). The internet is full of pictures but in order to be sure what is on them they usually need to be looked at by a human, which is expensive.

   Suppose we have data $(X_1, Y_1), \ldots, (X_N, Y_N), X_{N+1}, \ldots, X_M$ where $M \geq N$.
   The $X_{N+1}, \ldots, X_M$ are the "unsupervised" extra X-values. Y_i are the class values, $Y_i \in \{\text{ORANGE}, \text{GREEN}, \text{BLUE}\}$. The X_i are two-dimensional attributes $X_i = (X_{i,1}, X_{i,2})$, where $X_{i,1}$ and $X_{i,2}$ are both real numbers. Figure 4.2 (page 105) in the book is a graphical representation of a data set of this kind with $N = M$. Suppose that the data were changed so that we only had *one* labeled example for each category, i.e. $N = 3$, and we have X_1 is a point in the southwest of the blue cloud, and $Y_1 = \text{BLUE}$; X_2 is in the southwest of the orange cloud, and $Y_2 = \text{ORANGE}$; X_3 is in the southeast of the green cloud, and $Y_3 = \text{GREEN}$. For all other points, only the X-values are given, and the Y-data are hidden to us.

   We use this data to find the maximum likelihood parameters for the model underlying Linear Discriminant Analysis. That is, let $\hat{\theta}_{N,M}$ be the parameter vector that maximizes $\log p_\theta((X_1, Y_1), \ldots, (X_N, Y_N), X_{N+1}, \ldots, X_M)$, and let $\hat{\theta}_N$ be the vector that maximizes $\log p_\theta((X_1, Y_1), \ldots, (X_N, Y_N))$. where

   $$\theta = (\mu_{\text{GREEN}}, \mu_{\text{ORANGE}}, \mu_{\text{BLUE}}, \pi_{\text{GREEN}}, \pi_{\text{ORANGE}})$$

   is the parameter vector as defined in the book on page 108, Section 4.3. We set Σ equal to the identity matrix, so we will not estimate it from data. $N = 3$ and $X_1, Y_1, X_2, Y_2, X_3, Y_3$ are as above. We use the inferred $\hat{\theta}_{N,M}$ and $\hat{\theta}_N$ to make predictions of future Y given future X. Note that the $\hat{\mu}_k$ in $\hat{\theta}_{N,M}$ are not simply averages of data points any more; to actually compute $\hat{\mu}_k$ we would need to use numerical optimization.

a. How does the decision boundary look like according to $\hat{\theta}_N$? How do you think would the decision boundary look like according to $\hat{\theta}_{N,M}$? Argue that it helps to take the additional X-values into account, i.e. to predict using $\hat{\theta}_{N,M}$ rather than $\hat{\theta}_N$.

b. Now we are going to use the same data to find the maximum conditional likelihood parameters for the same model. These are just the maximum likelihood parameters for the logistic regression model. With this method of fitting, can the additional data help to get better predictions of future data? If so, why? If not, why not?