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

N. Vasconcelos & K. Kreutz-Delgado  
Winter 2012  

Due Tuesday, March 6, 2012  

1. Consider a two dimensional classification problem with two Gaussian classes

\[ P_{X|Y}(x|i) = \frac{1}{\sqrt{(2\pi)^2|\Sigma|}} e^{-\frac{1}{2}(x-\mu_i)^T\Sigma^{-1}(x-\mu_i)}, \ i \in \{0,1\} \]

of identical covariance \( \Sigma = \sigma^2 I \). For all parts of this problem assume the 0/1 loss function.

(a) If the classes have means

\[ \mu_0 = -\mu_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \]

and equal prior probabilities, \( P_Y(0) = P_Y(1) \), what is the Bayes decision rule for this problem?

(b) What are the marginal distributions for the features \( x_1 \) and \( x_2 \) for each class? In particular:

i. Derive expressions for the class-conditional densities \( P_{X_1|Y}(x_1|i) \) and \( P_{X_2|Y}(x_2|i) \) for \( i \in \{0,1\} \), where \( x = (x_1, x_2)^T \).

ii. Plot a sketch of the two densities associated with class \( Y = 0 \) and a sketch of the two densities associated with class \( Y = 1 \).

iii. Determine which feature is most discriminant.

(c) Suppose that a linear transformation of the form

\[ z = \Gamma x \]

is applied to the data, where \( \Gamma \) is a \( 2 \times 2 \) matrix. The decision boundary associated with the BDR is now the hyperplane of normal \( w = (1/\sqrt{2}, -1/\sqrt{2})^T \) which passes through the origin.

i. Determine the matrix \( \Gamma \)

ii. What would happen if the the prior probability of class 0 was increased after the transformation? Here it suffices to give a qualitative answer, i.e. simply say what would happen to the hyperplane.

iii. What is the distance in the original space (\( x \)) which is equivalent to the Euclidean distance in the transformed space (\( z \))?

2. Consider a binomial random variable \( X \) with parameters \( n \) and \( p \), i.e.\(^1\)

\[ P_X(x) = \binom{n}{x} p^x (1 - p)^{n-x} \]

Assuming that the parameter \( n \) is known, and given a sample \( D = \{x_1, \ldots, x_N\} \), what is the maximum likelihood estimate of the parameter \( p \)?

\(^1\)For example, \( X \) could be the number of heads which occur in \( n \) flips of a coin, where \( p \) is the probability of heads.
3. Consider an $m$-dimensional random vectors $Y$, such that:

$$ Y = Ax + N $$

where $A$ is a full rank matrix and $x$ is an unknown, but deterministic, $n$-dimensional vector, with $n < m$. The random vector $N$ is an $m$-dimensional Gaussian random vector of mean $0$ and full rank covariance matrix $\Sigma$.

a) What is the pdf for the random vector $Y$?

b) Show that given an observation $Y = y$, the maximum likelihood estimate of $x$ is

$$ \hat{x} = (A^T \Sigma^{-1} A)^{-1} A^T \Sigma^{-1} y. $$

c) What is the least squares problem whose solution is equivalent to b)? Assuming that $\Sigma$ is diagonal, what is the role of this matrix, i.e. how does it change the canonical (unweighted) least squares problem?

4. Here we consider a 0/1 loss function classification problem with two equal-covariance Gaussian classes:

$$ P_{X|Y}(x|i) = \mathcal{G}(x, \mu_i, \Sigma), \ i \in \{0, 1\} $$

of equal a priori probability

$$ P_Y(i) = 1/2. $$

In class, we discussed a practical way to approximately implement the BDR solution to this problem based on estimating the parameters of each of the Gaussian classes separately (say via MLE estimation) and then “plugging in” the parameter estimates into the BDR to obtain an approximate decision boundary. Here we will consider an alternative solution, that works directly on the class posteriors.

(a) Show that the posterior probability for class 1 is of the form (the posterior for class 0 is $1 - P_{X|Y}(1|x)$)

$$ P_{Y|X}(1|x) = \frac{1}{1 + e^{-w^T t}} $$

where $t^T = [x^T 1]$. What is the vector $w$?

(b) Show that an iid sample $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ has posterior probability

$$ P_{Y|X}(D_y|D_x) = \prod_{i=1}^{n} P_{Y|X}(y_i|x_i) $$

with

$$ P_{Y|X}(y_i|x_i) = \left( \frac{1}{1 + e^{-w^T t_i}} \right)^{y_i} \left( \frac{e^{-w^T t_i}}{1 + e^{-w^T t_i}} \right)^{1-y_i}, $$

where $D_y = \{y_1, \ldots, y_n\}$ and $D_x = \{x_1, \ldots, x_n\}$.

(c) Now learn the classification boundary, by directly learning the parameters, $w$, of posterior probabilities using standard (conditional) maximum likelihood estimation. I.e., solve for $\hat{w}$ such that

$$ \hat{w} = \text{arg max}_w P_{Y|X}(D_y|D_x). $$

---

Note that $Y$ and $N$ are both random vectors (which is why they are denoted by upper case letters, for clarification). For a given realization (aka “observation,” aka “instantiation,” aka “experimental outcome”) of the world, we have $Y = y$, $N = n$, and therefore $y = Ax + n$. 

---

2 Note that $Y$ and $N$ are both random vectors (which is why they are denoted by upper case letters, for clarification). For a given realization (aka “observation,” aka “instantiation,” aka “experimental outcome”) of the world, we have $Y = y$, $N = n$, and therefore $y = Ax + n$. 

---

2 Note that $Y$ and $N$ are both random vectors (which is why they are denoted by upper case letters, for clarification). For a given realization (aka “observation,” aka “instantiation,” aka “experimental outcome”) of the world, we have $Y = y$, $N = n$, and therefore $y = Ax + n$. 

---

2 Note that $Y$ and $N$ are both random vectors (which is why they are denoted by upper case letters, for clarification). For a given realization (aka “observation,” aka “instantiation,” aka “experimental outcome”) of the world, we have $Y = y$, $N = n$, and therefore $y = Ax + n$. 

---

2 Note that $Y$ and $N$ are both random vectors (which is why they are denoted by upper case letters, for clarification). For a given realization (aka “observation,” aka “instantiation,” aka “experimental outcome”) of the world, we have $Y = y$, $N = n$, and therefore $y = Ax + n$. 

---
5. Continuation of Computer Assignment.

In the last two computer problems, we saw how classification of hand written digits works. This time we will move to a more general and practical situation. For this problem, you are to use the data-set given with Homework 4.

We shall continue with the same training data used in the previous computer experiments, but with a new set of test data testImagesNew, which has been intentionally: 1) corrupted by noise and 2) re-scaled (attenuated) in amplitude.

This corresponds to the situation where we have clean training data (which presumably has been imaged using high-quality imaging systems in under optimal lighting), but poorer quality test data corresponding to the type of run-time digits images that would be produced using inexpensive image-scanners in poorly illuminated environments. (Poor illumination leads to contrast attenuation, and pixel-noise if the scanner is low-quality.) The image on the left of Fig.1 shows the case of an intensity-attenuated, noisy digital scan of the digit on the left. We want to classify such corrupted digits using a NN classifier.

We shall continue with the same training data used in the previous experiment, but with a new set of test data testImagesNew, which has been intentionally: 1) re-scaled in amplitude, and 2) corrupted by noise.

Using the Nearest Neighbor approach, we will find the training image that is nearest to an estimate of the uncorrupted test image. I.e., instead of the using the Euclidean distances between the training images and the corrupted test image, we will use the Euclidean distances between the training images and the ML estimate of the uncorrupted test image.

Toward this end, we assume that a corrupted test image $Y$ is a random vector of the form

$$Y = aX + N$$

i.e. the result of corrupting an unknown clean image $X$ through 1) amplitude re-scaling by the scalar $a$, and 2) addition of independent zero mean Gaussian noise with variance $v$, i.e. $N \sim G(0,vI)$.

The approach to be used is:

(a) Determine a maximum likelihood estimate (MLE), $\hat{a}_{ML}$, of the attenuation parameter $a$. The simplest way to do this is if we are given given the corrupted observation $y$ (the test image) corresponding to a known training image $x$. In which case we can compute the conditional MLE of $a$,

$$\hat{a}_{ML} = \arg \max_a P_{Y|X}(y|x; a,v).$$

(b) Given the test image $y$ and the MLE of the parameter $a$, determine the maximum likelihood estimate (MLE) $\hat{x}_{ML}$ of the unknown “clean image” $x$, assuming the model given in Eq. (4) (but with $a$ approximated by its maximum likelihood estimate).^5

(c) Determine the closest training image, $x$, to $\hat{x}_{ML}$. Once the closest training image has been found, its label is assigned to the test image $y$, thereby performing a classification of $y$. (I.e., the Euclidean distance between the scaled test image, $\hat{x}_{ML}$, and a training image, $x$, serves as the metric for the NN classifier.)

---

^3This will be the case here. If this were not the case, the problem would be much harder and we would have to be much cleverer on how to do this.

^4What about the variance parameter $v$?

^5Consistent with intuition, you should find that $\hat{x}_{ML}$ is a scaled version of the test image $y$. I.e., you should find that $\hat{x}_{ML} = c\cdot y$. What do you intuitively think the value of the scale parameter $c$ should be?
Perform the following:

(i) Using the two sample images sampletest.png and sampletrain.png, calculate the maximum likelihood estimate (MLE), \( \hat{a}_{\text{ML}} \), of the scale parameter \( a \) in Eq. (4).

(ii) Using the MLE \( \hat{a}_{\text{ML}} \) in place of \( a \) in Eq. (4), determine a formula which expresses \( \hat{x}_{\text{ML}} \) as a function of \( y \).

![Sample Images: Corrupted on the left. Clean on the right.](image)

Figure 1: Sample Images: Corrupted on the left. Clean on the right.

(iii) Now for the new testset testImagesNew, perform the task of classification using the least-squares distance metric applied to \( \hat{x}_{\text{ML}} \) (which is computed from \( y \)). As was done for our previous computer assignments, compute and plot the error rates for each class, and the total error rate.

(iv) Perform a NN classification on the new testset, using the simple algorithm of Computer Problem 1, i.e. using Euclidean distance metric applied to \( y \) (instead of to \( \hat{x}_{\text{ML}} \), as done in the previous step). Compare your results with the NN classification performed in the previous step.

**Notes:**
- Convert everything to double precision before calculating the MLE of \( a \).
- All the data needed for this assignment is uploaded on the website. Do not use data from the previous experiments.