E1 213 Pattern Recognition and Neural Networks

Practice Problems: Set 3

1. Consider a two class problem with one dimensional feature space. Suppose we have six training samples: x_1, x_2, x_3 from one class and x_4, x_5, x_6 from the other class. Suppose we want to estimate the class conditional densities nonparametrically through a Parzen window estimate with Gaussian window with width parameter σ. Write an expression for the Bayes classifier (under 0–1 loss function) which uses these estimated densities.

Hint: This problem is just to make sure you got the idea of nonparametric estimate. From what is given, the estimate of $f_1(x)$ and $f_2(x)$, the class conditional densities are

$$f_1(x) = \frac{1}{3}\sum_{i=1}^{3} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-x_i)^2}{2\sigma^2}}; \quad f_2(x) = \frac{1}{3}\sum_{i=4}^{6} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-x_i)^2}{2\sigma^2}}$$

Now we implement Bayes classifier: decide on class-1 if $f_1(x) > f_2(x)$ and class-2 otherwise (assuming equal priors).

2. Consider the kernel density estimate given by

$$\hat{f}_n(x) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{h_n} \phi\left(\frac{x - x_i}{h_n}\right)$$

Let the function ϕ be given by $\phi(x) = \exp(-x)$ for $x > 0$ and it is zero for $x \leq 0$. Suppose the true density (from which samples are drawn) is uniform over $[0, a]$. Show that the expectation of the density estimate is given by

$$E\hat{f}_n(x) = \begin{cases} 0 & \text{for } x < 0 \\ \frac{1}{a}\left(1 - e^{-x/h_n}\right) & \text{for } 0 \leq x \leq a \\ \frac{1}{a}\left(e^{a/h_n} - 1\right) e^{-x/h_n} & \text{for } x \geq a \end{cases}$$

Is this a good approximation to uniform density? Explain.

Hint: The x_i in the expression for $\hat{f}_n(x)$ are iid and hence each term in the summation has the same expectation (as explained in class). Taking

Z as a random variable that has the same distribution as x_i, we can write the expectation as

$$E\hat{f}_n(x) = \int_{-\infty}^{\infty} \frac{1}{h_n}\, \phi\left(\frac{x-z}{h_n}\right)\, f_Z(z)\, dz$$

We are given that the samples are drawn from a uniform distribution and thus $f_Z(z) = 1/a$ if $z \in [0,\ a]$ and is zero otherwise. Also, $\phi(t)$ is zero for $t \leq 0$. Hence, the limits for z in the above integral would be 0 to x when $0 \leq x \leq a$ and they will be 0 to a if $x > a$. Now evaluating the integral would give you the expectation as needed. To say whether it is a good approximation, you need to consider limit as $h_n \to 0$. It would be the uniform density (as it should be because you know kernel density estimate is consistent)

3. Consider 2-class PR problems with n Boolean features. Consider two specific classification tasks specified by the following: (i) a feature vector X should be in Class-I if the integer represented by it is divisible by 4, otherwise it should be in Class-II; (ii) a feature vector X should be in Class-I if it has odd number of 1's in it, otherwise it is in Class-II. In each of these two cases, state whether the classifier can be represented by a Perceptron; and, if so, show the Perceptron corresponding to it; if not, give reasons why it cannot be represented by a Perceptron.

Hint: A Boolean number is divisible by 4 iff the least two significant bits are zero. Suppose x_1 is the least significant bit and so on. Then a perceptron with $w_1 = w_2 = 1$, $w_i = 0$, $3 \leq i \leq n$ and $w_0 = -0.5$ would be able to do this classification.
For the second part, in $n = 2$ case it is the Boolean function XOR and that is clearly not linearly separable. For the general case. Suppose odd number of 1's is negative class. Suppose there is a Perceptron, W, for this problem. Take a X with odd number of 1's. If we change one of the 0's to 1 then $W^T X$ has to increase becuase now X is positive class. Which means the w_i corresponding to the bit we changed has to be positive. But this means all w_i have to be positive. Then, if we start with a X with even number of 1's and change a zero to one, $W^T X$ cannot decrease.

4. Consider the incremental version of the Perceptron algorithm. The algorithm is: at iteration k, if $W(k)^T X(k) \leq 0$ and thus we misclassified

the next pattern then we correct the weight vector as: $W(k+1) = W(k) + X(k)$.

(i). By going over the proof presented in class, convince yourself that if we change the algorithm to $W(k+1) = W(k) + \eta X(k)$ for any positive step-size η, the proof is still valid.

(ii). In the perceptron algorithm, when we misclassify a pattern and hence correct the weight vector, the algorithm does not necessarily ensure that $W(k+1)$ will classify $X(k)$ correctly. Suppose we want to change the algorithm so that when we misclassify a pattern, we change the weight vector by an amount that ensures that after the correction, the weight vector correctly classifies this pattern. While this may seem like just a matter of choosing a 'step-size', note that if we want to choose η so that the above is ensured at every k then, the 'step-size' may have to vary from iteration to iteration and it may be a function of the feature vector. Hence, the earlier proof may not go through. Design a simple modified version of the Perceptron algorithm which effectively ensures the above property and for which the same convergence proof holds.

Hint: Hint is needed only for part (ii).

We can ensure that we correct W enough while using a step-size that is an integer (which would be a function of X). That is, we add to $W(k)$ either $X(k)$ or $2X(k)$ or $3X(k)$ or $\cdots$ so that $W(k+1)^T X(k)$ becomes positive. Now, this can be viewed as follows. When we misclassify $X(k)$, we correct $W(k)$ (by the standard perceptron algorithm) and then present the same example again. We will not move to the next example in the input sequence till we classify the current example correctly. This only changes the order of presentation of examples but ensures all examples are presented again and again (because we any way cycle through all examples). Hence the same proof works.

5. Consider the joint density of X, Y given by

$$f_{XY}(x,y) = \frac{1}{2\pi\sigma^2\sqrt{1-\rho^2}} \exp\left(\frac{-1}{2\sigma^2(1-\rho^2)}(x^2+y^2-2\rho xy)\right), \ -\infty < x, y < \infty$$

Find a function g that minimizes $E\left[(g(X)-Y)^2\right]$.

Hint: We know that the best g is conditional expectation. By simple integration (or from the properties of jointly Gaussian rv's) one can show that

the marginal is Gaussian with mean zero and variance σ^2. By dividing the joint density by the marginal density you can see that the conditional density of Y given X is Gaussian with mean ρx and variance $\sigma^2(1-\rho^2)$. Hence you conclude that the conditional expectation is ρX. So, when X, Y are jointly Gaussian the optimal regression function is linear.
(You may have done this in your probability course).

6. Suppose we have $y = \mathbf{a}^T X + \xi$ where ξ is a zero-mean random variable with variance σ^2. Under this model we have calculated in the class the variance of the least squares solution, W^*. Calculate the expected value of the least squares solution.

Hint: Intuitively, we should have $EW^* = \mathbf{a}$ in this case. Given the actual relationship between X and y, we see that $Ey_i = \mathbf{a}^T X_i = X_i^T \mathbf{a}$. Since rows of matrix A are X_i^T and Y is vector with components y_i, we have $EY = A\mathbf{a}$. Least squares solution is $W^* = (A^T A)^{-1} A^T Y$. Taking expectations, $EW^* = (A^T A)^{-1} A^T EY = (A^T A)^{-1} A^T A\mathbf{a} = \mathbf{a}$.

7. We can pose the problem of learning a linear classifier as minimizing

$$J(W) = \sum_{i=1}^{n} L(W^T X_i, y_i)$$

where L is a loss function. For least squares criterion, we take $L(a, b) = (a - b)^2$. If, instead we want to minimize absolute value of error, we can take $L(a, b) = |a - b|$. Show that logistic regression (in the 2-class case) can also be put in this framework with $L(W^T X, y) = \ln(1 + \exp(-yW^T X))$, where we assume that the class labels are $+1$ and -1. What would be the loss function corresponding to mult-class logistic regression?

Hint: In logistic regression, we model y as Bernoulli with parameter $1/(1 + \exp(-W^T X))$ and then find ML estimate for W. We want to take $y \in \{-1, +1\}$. So, the model is

$$P[y = 1|X] = \frac{1}{1 + \exp(-W^T X)}; \;\; P[y = -1|X] = 1 - \frac{1}{1 + \exp(-W^T X)} = \frac{1}{1 + \exp(W^T X)}$$

So, the conditional density can be written as

$$f(y|X) = \frac{1}{1 + \exp(-yW^T X)}, \;\; y \in \{-1, +1\}$$

We want to maximize $\ln\left(\prod_i f(y_i|X_i)\right)$ which is same as minimizing $\sum_i \ln(1+\exp(-y_i W^T X_i))$ which can be viewed as the empirical risk if we take the loss function as $L(W^T X, y) = \ln(1+\exp(-yW^T X))$

In the multi-class case, we take y to be a one-hot vector with K components and our model would have K weight vectors as parameters. To figure out the loss function in multi-class case, look at the log likelihood given in the lecture. (Note that in the multiclass case we have to write the loss as $L(y, g(W, X))$ where you should remember that both arguments are K-dimensional vectors; W actually involves K weight vectors and y is a one-hot vector). Hope you can complete this now. To check your answer, consider the multi-class loss function for the special case of $K = 2$. This would have as parameters two weight vectors: W_1, W_2. By defining W as some suitable function of W_1, W_2, the multi-class loss function for the case $K = 2$ should become same as the 2-class loss function given above.

8. Consider a classification problem with K classes: $C_1, \cdots, C_K$. We say that the training set is linearly separable if there are K functions: $g_j(X) = W_j^T X + w_{j0},\ j = 1, \cdots, K$, such that we have $g_i(X) \geq g_j(X), \forall j$, whenever $X \in C_i$. We say that a set of examples is totally linearly separable if given any C_i, there is a hyperplane that separates examples of C_i from the set of examples of all other claases. Show that totally linearly separable implies linearly separable but the converse need not be true.

Hint: If each class is linearly separable from the rest, then, for each class C_j, there is a W_j, w_{j0} such that $W_j^T X + w_{j0} > 0$ if X is in C_j and it is negative if X is in any other class. Thus, with $g_j(X) = W_j^T X + w_{j0}$, we satisfy the condition given. This shows that totally linearly separable implies linearly separable. To show that linearly separable does not imply totally linearly separable, all you need is a counter-example. You can try to construct an example as follows. Take $X \in \Re^2$. We can take $||X_i|| = 1$ and also $||W|| = 1$. Then $W^T X_i$ is just cos of the angle. Now what you are asking for is this: Can we have three such vectors, W_1, W_2, W_3 so that when we define the class of any X by the W_i that it is closest to (in terms of angle), there is at least one class that is not linearly separable from the other two. I hope you can now construct the example.