Homework 4

CSE 802 - Pattern Recognition and Analysis
Instructor: Dr. Arun Ross
Points: 100 (+10 bonus points)
Due Date: April 29, 2015 (5:00pm)

Please read the following instructions carefully:

1. You are permitted to discuss the following questions with others in the class. However, you must write up your own solutions to these questions. Any indication to the contrary will be considered an act of academic dishonesty.

2. A hard-copy of this assignment must be submitted by April 29, 5:00pm. Late submissions will not be graded. In this copy, please include the names of individuals you discussed this homework with and the list of external resources (e.g., websites, other books, articles, etc.) that you used to complete the assignment (if any).

3. Code developed as part of this assignment should be placed in a zip file and sent to rossarun at cse.msu.edu with the subject line “CSE 802: Homework 4”.

1. [20 points] Generate 100 random training points from each of the following two distributions: N(20,5) and N(35,5). Write a program that employs the Parzen window technique with a Gaussian kernel to estimate the density, \( \hat{p}(x) \), using all 200 points. Note that this density conforms to a single bimodal distribution (a Gaussian Mixture Model).

   (a) Plot the estimated density function for each of the following window widths: \( h = 0.01, 0.1, 1, 10 \). [Note: You can estimate the density at discrete values of \( x \) in the \([0,55]\) interval with a step-size of 1.]

   (b) Repeat the above after generating (i) 500 training points from each of the two distributions, (ii) 1,000 training points from each of the two distributions; and (iii) 2,000 training points from each of the two distribution.

   (c) Discuss how the estimated density changes as a function of the window width and the number of training points.

2. [30 points] Consider the dataset available here. It consists of two-dimensional patterns, \( x = [x_1, x_2] \), pertaining to 3 classes (\( \omega_1, \omega_2, \omega_3 \)). The feature values are indicated in the first two columns while the class labels are specified in the last column. The priors of all 3 classes are the same and a 0-1 loss function is assumed. Partition this dataset into a training set (the first 250 patterns of each class) and a test set (the remaining 250 patterns of each class).
(a) Let
\[
p([x_1, x_2]^t | \omega_1) \sim N([0, 0]^t, 4I),
p([x_1, x_2]^t | \omega_2) \sim N([10, 0]^t, 4I),
p([x_1, x_2]^t | \omega_3) \sim N([5, 5]^t, 5I),
\]
where \( I \) is the \( 2 \times 2 \) identity matrix. What is the error rate on the test set when the Bayesian decision rule is employed for classification? Report the confusion matrix as well.

(b) Suppose \( p([x_1, x_2]^t | \omega_i) \sim N(\mu_i, \Sigma_i), i = 1, 2, 3, \) where the \( \mu_i \)'s and \( \Sigma_i \)'s are unknown. Use the training set to compute the MLE of the \( \mu_i \)'s and the \( \Sigma_i \)'s. What is the error rate on the test set when the Bayes decision rule using the estimated parameters is employed for classification? Report the confusion matrix as well.

(c) Suppose the form of the distributions of \( p([x_1, x_2]^t | \omega_i), i = 1, 2, 3 \) is unknown. Assume that the training dataset can be used to estimate the density at a point using the Parzen window technique (a spherical Gaussian kernel with \( h = 1 \)). What is the error rate on the test set when the Bayes decision rule is employed for classification? Report the confusion matrix as well.

(d) Suppose the 1-nearest neighbor method is used for classifying the patterns in the test set. What is the error rate of this approach? Report the confusion matrix as well.

3. [5 points] Consider a tangent-distance based classifier based on \( n \) prototypes, each representing a \( k \times k \) pixel pattern of a handwritten character. Suppose there are \( r \) invariances we believe characterize the problem. What is the storage requirement (space complexity) of such a tangent-based classifier? Explain clearly.

4. [10 points] Determine whether the four properties of a metric are obeyed by the Tanimoto distance.

5. [15 points] Consider a set of two-dimensional points pertaining to two classes that are linearly separable. The data can be accessed here. Note that the class labels (+1 or -1) are indicated at the end of every pattern.

Write a program to compute the linear decision boundary for this two-class problem by implementing the fixed-increment single-sample perceptron learning algorithm. Report the linear decision boundary computed by the learning algorithm and the number of iterations taken by the algorithm to converge to this boundary when the weight vector is initialized as follows: (i) \((-1, -1, -1)^t\), (ii) \((1, 1, 1)^t\), (iii) \((5, 5, 5)^t\), (iv) \((-30, 1, 1)^t\), and (v) \((10, 10, 10)^t\). In each case plot (i) the two dimensional-points, (ii) the initial decision boundary and (iii) the final decision boundary.

6. [5 points] Based on the notations used in the SFS/SFFS algorithm discussed in class, write down the algorithm for Sequential Floating Backward Selection.
7. [15 points] Consider a classification problem involving four features - \( x_1, x_2, x_3, x_4 \). The accuracy of a particular classifier for all possible feature subsets is given in the following table.

<table>
<thead>
<tr>
<th>Feature Subset</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>{x_1}</td>
<td>70%</td>
</tr>
<tr>
<td>{x_2}</td>
<td>80%</td>
</tr>
<tr>
<td>{x_3}</td>
<td>75%</td>
</tr>
<tr>
<td>{x_4}</td>
<td>60%</td>
</tr>
<tr>
<td>{x_1, x_2}</td>
<td>88%</td>
</tr>
<tr>
<td>{x_1, x_3}</td>
<td>85%</td>
</tr>
<tr>
<td>{x_1, x_4}</td>
<td>88%</td>
</tr>
<tr>
<td>{x_2, x_3}</td>
<td>90%</td>
</tr>
<tr>
<td>{x_2, x_4}</td>
<td>60%</td>
</tr>
<tr>
<td>{x_3, x_4}</td>
<td>70%</td>
</tr>
<tr>
<td>{x_1, x_2, x_3}</td>
<td>91%</td>
</tr>
<tr>
<td>{x_1, x_3, x_4}</td>
<td>95%</td>
</tr>
<tr>
<td>{x_1, x_2, x_4}</td>
<td>88%</td>
</tr>
<tr>
<td>{x_2, x_3, x_4}</td>
<td>93%</td>
</tr>
<tr>
<td>{x_1, x_2, x_3, x_4}</td>
<td>84%</td>
</tr>
</tbody>
</table>

(a) According to the SFS algorithm, what would be the “best” feature subset of size 3? How does it compare with the “optimal” feature subset of size 3?

(b) According to the SBS algorithm, what would be the “best” feature subset of size 2? How does it compare with the “optimal” feature subset of size 2?

(c) Give two reasons why a feature subset of size 3 may have a higher accuracy than a set comprising of all 4 features.

Bonus Question:

1. [10 points] Consider a 2-category classification problem involving a single feature \( x \). Assume equal class priors and a 0-1 loss function. The class-conditional densities are as follows:

\[
p(x | \omega_1) = \begin{cases} 
2x & \text{for } 0 \leq x \leq 1 \\
0 & \text{otherwise},
\end{cases}
\]

\[
p(x | \omega_2) = \begin{cases} 
2 - 2x & \text{for } 0 \leq x \leq 1 \\
0 & \text{otherwise}.
\end{cases}
\]

Suppose we randomly select a single point from \( \omega_1 \) and a single point from \( \omega_2 \), and create a 1-nearest-neighbor classifier. Suppose too we select a test point from one of the categories (\( \omega_1 \) for definiteness). Integrate to find the expected error rate \( P_1(e) \).