Fall 2021

COSC 522 – Homework #5 (Due 11/09)

Problem 1: (30 pts.) On classifier fusion. Assume there are two classifiers (L1, L2), performing classification tasks on three objects (ω_1, ω_2, ω_3). Assume there are 30 samples in the training data for each category. Following are the confusion matrices generated from the training data.

L1	ω_1	ω_2	ω_3
ω_1	20	5	5
ω_2	3	24	3
ω_3	0	9	21

L2	ω_1	ω_2	ω_3
ω_1	25	2	3
ω_2	3	22	5
ω_3	5	6	19

In the confusion matrix, each row represents the ground truth label and each column represents the label given by the classifier.

(a) (20 pts.) Derive a lookup table that includes the fused result from all possible combinations of labels from the two classifiers using Naïve Baysian (show details and justify your selection of fused labels)

L_1, L_2	Fused label
1, 1	
1, 2	
1, 3	
2, 1	
2, 2	

2, 3	
3, 1	
3, 2	
3, 3	

(b) (10 pts.) How to generate the lookup table using BKS from the above confusion matrices, if possible at all? If not possible, then what additional information would you need to generate the lookup table for BKS? What is the difference between Naïve Baysian and BKS?

Problem 2: (20 pts) Decision tree. Assume 100 samples are assigned to node N, among which 90 samples actually belong to class 1 and 10 samples actually belong to class 2. For the following two split candidates resulted in two different queries being conducted, which one is a better query according to the Gini impurity?

- Option 1: 70 class 1 samples go to "left", 20 class 1 samples and 10 class 2 samples go to "right"
- Option 2: 80 class 1 samples go to "left", 10 class 1 samples and 10 class 2 samples go to "right"

Need to show details.

Problem 3: (50 pts.) Using SVM for XOR. The four training samples are [[-1, -1], [-1, 1], [1, -1], [1, 1]], with the corresponding label being [-1, 1, 1, -1]. Obviously, the decision boundary would not be linear. We use the kernel trick to solve this problem. Kernel tricks use a kernel function to project the data from the original space (in this case, 2-d space) to a higher-dimensional space where a linear boundary can be found to perfectly separate the samples.

(a) (10 pts) Suppose the kernel function is a 2nd-degree polynomial, i.e., $K(\mathbf{x}, \mathbf{y}) = (x_1y_1 + x_2y_2 + C)^2$, where $\mathbf{x} = [x_1\ x_2]^T$, $\mathbf{y} = [y_1\ y_2]^T$. Derive the basis functions $\phi(x)$, that is, $K(\mathbf{x}, \mathbf{y}) = \phi^T(\mathbf{x}).\phi(\mathbf{y})$

(b) (10 pts) Using the derived basis function, what is the higher-dimensional space that the 2-d sample should be mapped to? Provide the higher-dimensional counterpart to the four 2-d samples.

(c) (15 pts) Apply Perceptron on the higher-dimensional samples and see if you can find a linear decision boundary using the higher-dimensional samples. Output the weights learned. Project the learned hyperplane onto 2-D space and show the decision boundary.

(d) (15 pts) What would be the support vectors in the higher-dimensional space? Manually find the decision boundary if using SVM. Provide parameters that determine the decision boundary. Project the decision boundary onto 2-D space.