Fall 2024

COSC522 – Homework #5 (Due 11/26)

Problem 1: (35 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?

(c) (5 pts.) How about the other way around? If given the training table as shown in Lecture 13 (slide 12), would you be able to derive the lookup table for Naïve Bayes?

Problem 2: (45 pts) Read through Pass and Howard's How to explain gradient boosting. A toy example was given regarding a regression problem where you are supposed to predict the rent price given the square footage of the apartment. For the dataset given from the reading, answer the following questions:

sqfeet	rent
750	1160
800	1200
850	1280
900	1450
950	2000

(a) (10 pts) Apply simple linear regression and derive the model. Compare model performance with those of L2Boost and L1Boost.

(b) (10 pts) Why is it called "Gradient" boosting? What is boosting?

(c) (25 pts) Apply L2Boost on Problem 3 of Homework 4. Compare performance with that of simple regression.

Problem 3: (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?

(a) Option 1: 70 class 1 samples go to "left", 20 class 1 samples and 10 class 2 samples go to "right"

(b) 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.

Bonus: (+10 pts) One way to populate the ROC curve of a classifier is to change the prior probability. However, in kNN, the prior probability is fixed at n_k/n. Suggest potential modifications to kNN such that different prior probabilities can be incorporated.