CS 422: Data Mining

Department of Computer Science
Illinois Institute of Technology
Vijay K. Gurbani, Ph.D.

Spring 2022: Homework 5 (10 points)

Due date: Sun, March 21 2022, 11:59:59 PM Chicago Time

Please read all of the parts of the homework carefully before attempting any question. If you detect any ambiguities in the instructions, please let me know right away instead of waiting until after the homework has been graded.

Remember: This is an individual assignment. Sharing of code is strictly prohibited, and affected students will loose points if code is determined to have been shared.

1. Questions

1.1 (3 points) Tan, Ch. 5 (Association Analysis)

Question 1, 2, 9(a), 9(b)

2. Problems

2.1 (7 points) Grid search and Random Forests

A grid search is a hyperparameter optimization method; it chooses a set of optimal hyperparameters for a learning algorithm. For example, if you wanted to tune two parameters, **a** and **b**, and **a** can take the values between 1-3, and **b** can take the values between 4-6, you will do a grid search like so:

```
for a = 1 to 3 {
  for b = 4 to 6 {
    model = train_ML_model(..., a, b)
    result = predict(model, ...)
    save result
  }
}
```

At the end of the loops, you examine result (the collection data structure to which you append your results in each iteration) and see which value of **a** and **b** leads to a model you will consider to be best.

To help you guide your programming, please see Problem-2-1-pseudocode.r file uploaded in Blackboard. Please feel free to use this file to guide the structure of your code.

You will use the same dataset in HW 4. In HW 4, Problem 2.1 you created a decision tree model, you will now train a RandomForest model to predict the income. Use the same training and test datasets from HW4.

You will conduct a grid search across two Random Forest : ntree (number of trees in the forest) and mtry (randomly chosen attributes for each split). Run the grid search programmatically, i.e., using loops, instead of manually building nine models.

Use the following values for the ntree: 250, 500, 750 (the 750 value for ntree may take some time, so make sure you let your computer run until it is done).

Use the following values for the mtry parameter: $\lfloor\sqrt{n}\rfloor$, $\lfloor\sqrt{n}\rfloor + 1$, $\lfloor\sqrt{n}\rfloor + 2$ (where n is the number of predictors in your base model of Q 2.1).

At the end of each iteration of the grid search, save the following items in a collection data structure:

1) The OOB error estimate obtained during training. (Hint: examine the err.rate field of the model to determine the mean OOB estimate. This field will contain as many rows as the value of ntree under consideration. The first column in the err.rate field is the OOB error, and the i-th element is the OOB error for all trees up to the i-th tree.)

2) The confusion matrix that results from fitting the model on the held-out test dataset.

At the end of the grid search, examine your collection data structure in which you saved your results and:

(a) Determine the **best** model by examining balanced accuracy, sensitivity, and specificity as shown in the confusion matrix from the held-out test dataset, and picking the model that shows the **maximum** balanced accuracy, sensitivity and specificity. Print out the best model as shown below:

Grid search resulted in the best model at ntree = XXX and mtry = XXX.
Accuracy = ...
Balanced Accuracy = ...
Sensitivity = ...
Specificity = ...

(b) Determine the **best** model bu examining the lowest (minimum) OOB error rate. Print out the best model as shown below:

Grid search resulted in the best model for OOB at ntree = XXX and mtry = XXX.
OOB = ...

(c) Is the best model as determined by (a) the same model as determined by (b). Justify your answer.