Assignment 3 (12 Marks)
Deadline: October 21 2018, 11:59pm

1. PROBLEM DESCRIPTION

In this assignment, we will practice executing ensemble methods in R and learn how those methods could help us improve the prediction performance.

The data is a simulated dataset with one binary label and 15 features. There are 2000 records in the training data (with label values) and 2000 records in the test data (without label values). For Task 1 and Task 2, please use the first 1500 records in A3_train.csv for training and the last 500 records in A3_train.csv for computing performance.

2. TASKS

Task 1: Write your own code of Random Forest of a post-pruned rpart by modifying the R code template uploaded to IVLE A3 folder. (3 marks)

- You might find the swirl exercise (BT5152 Tutorial 1 - Decision Trees) from week 3 helpful if you need to refresh your memory on post-prune of an rpart decision tree.
- Performance metric is simple accuracy for binary labels.
- This part is for practice purpose to help you check your understanding about Random Forest algorithm. In practice, most packages implements with using a fully grown tree.
- Grading of this part is about the correctness of your code and checking your understanding about random forest. Prediction performance won't be graded.

Task 2: Stacking of three algorithms: C50 with default parameter values, KNN with k=3, and your random forest in Task 1. The output of level-0 is a binary label (not predicted probability). Logistic regression is used for the level-1 algorithm. The learning objective is to help you check your understanding about Stacking. (4 marks)

- The final output is a binary label and the performance metric is simple accuracy.
- You may use the same classification problem dataset provided in the template for Task 1. Make sure your implementation is able to report the prediction accuracy on the test dataset.
- Same as Task 1: grading of this part is about the correctness of your code. Prediction performance won't be graded.
- In this question, you need to code the details of Stacking. In other words, you are not allowed to use caretEnsemble or caretStack. You are allowed and encouraged to use these packages in Task 3.

- Bonus (up to 1 mark): You may include additional code and a half page discussion comparing your stacking implementation and any of the level-0 models. You may also consider generalizing your stacking implantation such that it can be used on any classification dataset.

Task 3: Toy Data Competition. Now you try your best to predict the true label of the 2000 rows in the test set file (the file without true label). **The performance metric is AUC**. In other words, you are required to submit predicted probabilities. (5 marks)

- Grading of this task is based on **your prediction performance and reproducibility of your prediction results**. You need to submit your predicted values and also the code to generate predicted values for verification purpose. If your AUC is around median AUC of this class, your expected mark is 2.5 out of 5 in this assignment.
- To alleviate the workload of TA, your training code must complete within 5 minutes.
  - You can grid-search by Caret and only submit the code to build your final model with the chosen parameters. On my 3-year old normal desktop, xgBoost takes less than 1 second to train on this dataset.
- You are allowed to use any R packages for algorithms covered in our lectures, the required textbook, and tutorials before week 7 (including Week 7). Packages for algorithms not covered so far are NOT allowed.
  - Only R is allowed. Python is not allowed in this exercise.
  - LightGBM is not allowed. GBM or XGBoost in R is allowed.
  - At the same time, you are allowed to try different settings of any of the R packages covered. You do not need to stick to the (default) parameter settings used in the sample codes from tutorials. For example, you can change the parameter settings of neuralnet or nnet packages in any way that you like.
  - Using randomForest package in R or Caret is allowed. No need to use the hand-coded version of Random Forest.
  - caretEnsemble or caretStack is allowed.
  - You can choose to use Caret or not.
- You are allowed and encouraged to create new features based on raw data. Any function for features engineering is allowed.
- You are allowed to drop features if you believe it helps the performance. Using R packages to help you execute features selection methods or dimension reduction methods is allowed.

Submissions and Grading

- You can submit up to three files:
    1. a *.R (or *.Rmd) file [required]
    2. a *.csv for Q3 [required]
    3. a *.PDF (or *.html generated by your rmarkdown) file of you results and answers. [optional]
- **Name all files by your student number (e.g., A0123456X.R, A0123456X.html, A0123456X.csv) and upload to IVLE workbin submission folder "A3". <u>Do not zip your submissions.</u>**
- In your R script you can assume that dataset files are in the same directory as the R script, e.g. train_data <- read.csv("A3_train.csv")
- The page limit of the pdf file is maximum 2 pages including everything. The formatting is A4, default margin, 12 font size, single-spacing. There is no need to try to fill 2 pages. Correct answers are much more important than the length of your answers for grading.
- You may revise and submit as many times before deadline. Make sure to remove any old version that you don't wish to be graded.
- If you have questions about the assignment, feel free to email TA and cc me. Later, if you have questions about grading of the assignment, then you can email TA and cc me because TA (not me) will grade your assignment by following my grading rules listed below.

Grading Rules

- All suspected plagiarism cases will receive 0 mark.
- Every day of late submission will result in 3 marks deducted, i.e. 4 days late = 0 mark.
- For Q1, you should use the provided R code template without major modification. The completed R code should be correct.
- For Q2, you should have correct R code that can be used for prediction of at least the test classification dataset in Q1 and produce the test prediction accuracy.
- For Q3, R code that can reproduce the exact same prediction results as your *.csv submission. 0 will be given for Q3 if executing the R script/markdown doesn't produce the same csv file. -2.5 marks if execution time on the TA's 2GHz i7, 8GM memory MacBook Air is more than 10 minutes.
- Submissions without a runnable R/Rmd file will receive a failing grade. Make sure all the dependency packages are imported e.g. library(C50)
- TA can judge the quality of your code and deduct up to 2 marks. For example, if you have unnecessary/repeated code, meaningless variable names, excessive comments, marks may be deducted.