MAST90138 Assignment 3

Instructions:

- This assignment counts towards 15% of the final mark for the subject. If you LaTeXand `knitr` your assignment in a nice way, you will potentially get up to a maximum of 0.75% towards the final mark for the subject as extra credits. (You may use R Markdown as well.)
- Use tables, graphs and concise text explanations to support your answers. Unclear answers may not be marked at your own cost. All tables and graphs must be clearly commented and identified.
- No late submission is allowed.

Data: In the assignment you will analyze some rainfall data. The dataset is available in `.txt` format on the LMS website. To load the data into `R` you can use the function `read.table()` or any command of your choice. You may need to manipulate the data format (data frames or matrices) depending on the task. The data are separated in a training set and a test set. The training set contain $p = 365$ explanatory variables $X_1, \ldots, X_p$ and one class membership ($G = 0$ or 1) for $ntrain = 150$ individuals. The test set contains $p = 365$ explanatory variabless $X_1, \ldots, X_p$ and one class membership ($G = 0$ or 1) for $ntest = 41$ individuals.

In these data, for each individual, $X_1, \ldots, X_p$ correspond to the amount of rainfall at each of the $p = 365$ days in a year. Each individual in this case is a place in Australia coming either from the North ($G = 0$) or from the South ($G = 1$) of the country. Thus, the two classes (North and South) are coded by 0 and 1.

You will use the training data to fit your models or train classifiers. Once you have fitted your model or trained your classifiers with the training data, you will need to check how well the fitted models/trained classifiers work on the test data.

The test and training data are all placed in different text files: `XGtrainRain.txt`, which contains the training X data (values of the p explanatory X-variables) for $ntrain = 150$ individuals as well as their class (0 or 1) label, and `XGtestRain.txt`, which contains the test X data (values of the p explanatory X-variables) for $ntest = 41$ as well as their class (0 or 1) label. The test class membership is provided to you ONLY TO COMPUTE THE ERROR OF CLASSIFICATION of your classifier.

Please Include all the necessary R code to answer the questions, but not the superfluous R code that are not relevant. Marks may be taken off for R code that is poorly presented.

Problem 1 [60 marks]:

In this problem you will train quadratic discriminant (QDA) and logistic regression classifiers to predict the class label (0 or 1) in the test set.

(a) Using standard functions in R to train the QDA classifier and the logistic classifier , with all the p predictors in the training set. What happened? And why did it happen? Do you recommend using these two classifiers on the test set? (Hint: For the logistic classifier, use the `summary` function to take a look at the trained model object) [10]

(b) Use `prcomp` and the `plsr` (package `pls`) functions to obtain, respectively, the PCA and PLS components of the explanatory variables. Here, when considering the covariance maximization problem of PLS, we maximise the covariance between $X = (X_1, \dots, X_p)^T$ and $Y = 1\{G = 1\}$, the indicator variable that an individual belongs to group 1. For each case, you will need to use the "projection matrix" (i.e., Γ for PCA and Φ for PLS discussed in class) reported by the function to re-compute the components "manually" to check that you understand how the components are obtained. (Report your R outputs in a concise manner. Do not make it unnecessarily long) [10]

(c) Train a QDA classifier with the PLS components, and another one with the PCA components. In each case, pick the number of components to use based on leave-one-out cross validation (LOOCV); consider up to using 50 components. Plot the leave-one-out CV error against the number of components considered. Report the final chosen number of components. (Refer to the lab in Week 7 to get some ideas. The function `which.min` may be useful)

Do the exact same for the logistic classifier.

(You can expect LOOCV to be computationally intensive. If you want to pick your number of components based on methods other than LOOCV, please explain your choice in a clear and concise manner) [20]

(d) For each of the QDA and logistic classifiers, which version (PCA or PLS) do you prefer? Why? (Answer this question without any knowledge of the test-set results in the next sub-problem) [5]

(e) Apply your trained classifiers in (c) to the test set, and report the resulting classification error (test error). Be careful about how you should center the data in your test set to produce your prediction. The lab in Week 7 may give you some ideas again. [15]

Problem 2 [25 marks]:

In this problem you will train random forest (RF) classifiers to predict the class labels (0 or 1) in the test set.

(a) Using the `randomForest` package in R, construct a **random forest classifier** using all p predictor variables in the training set. When training the classifier, use the default value of m (the number of random candidate variables for each split), but justify your choice for the number of trees B using the out-of-bag (OOB) classification error. Plot a graph showing the OOB error against the number of trees used. [10]

(b) Show two graphs that illustrate the importance of the X_j variables, for both decrease in OOB prediction accuracy and decrease in node impurities measured by Gini index. Is there an explanation of why those particular X_j 's are the most important for classification in this rainfall example? [5]

(c) Apply the resulting trained classifier to the test data `XGtestRain`, and compute the resulting classification error. Try running your tree multiple times. Do you always get the same classification error? If yes, why? If not, why and what can you do to make the forest more stable and why? [10]

Problem 3 [15 marks]:

Compare the percentage of misclassification for each of the five classifiers (Logistic + PCA, Logistic + PLS, QDA + PCA, QDA + PLS, RF) considered in the previous problems. Identify the classifiers that worked the best, those which worked the worst, and comment on those results. Provide an explanation of the poorer/better performance of some of the classifiers. [15]