Machine Learning
Programming Assignment II-b

Ujjwal Sharma and dr. Stevan Rudinac

The following assignments will test your understanding of topics covered in the first four weeks of the course. These assignments **will count towards your grade** and should be submitted through Canvas by **28.11.2019 at 12:59 (CET)**. You can choose to work individually or in pairs. You can get at most 4 points for these assignments, which is 4% of your final grade.

Submission

You can only submit a Jupyter Notebook (*.ipynb) for Assignments 3A and 3B. All exercises in this assignment should be submitted with assignment 3A in the same notebook. To test the code we will use Anaconda Python 3.6. Please state the names and student ID's of the authors (at most two) at the top of the submitted file.

1 Implementation Details

In this assignment, we will be extending our coverage of classifiers to include two additional models you have seen this week, the *Decision Tree Classifier* and the *Random Forest Classifier.* Tree-based models have a distinct set of strengths and weaknesses and are uniquely suited for some problems. As before, you will be using the sklearn `Pipeline` functionality to encapsulate your preprocessing transformations as well as classification models into a single estimator. In the following assignments, you should perform preprocessing, model fitting and prediction operations only with a `Pipeline` estimator.

If you are asked to analyze an effect, we expect a written textual block summarizing your analysis. This can include plots if they strengthen your argument.

Any grid search should also be performed on the `Pipeline`, not on standalone estimators or transforms.

2 Data

For this assignment, you will continue to use the same dataset that was provided with assignment 3A. If you are attempting this assignment without attempting part 3A, we would advise you to go back and look at the previous assignment for more information on the data.

3 Models

The model structure introduced in Assignment 3A uses a `Pipeline` to wrap a *scaler*, an *imputer* and a *classifier.* For this assignment, you will use the same setup with two new tree-based classifiers. As such, you are only required to modify the classifier component of the pipeline.

In this assignment, you will be required to use the `DecisionTreeClassifier` and `RandomForestClassifier` as the primary classifiers. For these models, you must perform the following experiments:

✉ u.sharma@uva.nl, s.rudinac@uva.nl

1. Initialize a `DecisionTreeClassifier` model and perform the following tasks:
   (a) Fit your classifier on scaled as well as unscaled versions of the data and report the model scores.
   (b) Analyze the effects of data preprocessing operations like scaling on the classification performance of a decision tree classifier.
   (c) Provide a textual justification for why these models can overfit so easily if models parameters are not carefully selected. Supplement your explanation with suitable figures/tables if necessary.
   (d) Perform a grid search on the decision tree model parameters `['max_depth', 'max_features', 'min_samples_leaf', 'min_samples_split']` to evaluate the optimal values for these parameters. Report model parameters for your best classifier model.
   (e) Report all evaluation metrics described in the Model Evaluation section below.

   In no more than 50 words, explain your observations and your assessment of the classifier. You can use a text box (i.e. Markdown Cell) in Jupyter to write down your analysis.

2. Initialize a `RandomForestClassifier` model and perform the following tasks:
   (a) Fit your classifier on scaled as well as unscaled versions of the data and report the model scores.
   (b) Analyze the effects of data preprocessing operations like scaling on the classification performance of a random forest classifier and comment on its ability to overfit to data.
   (c) Both decision tree classifiers and random forest classifiers use decision trees to classify data. In less than 100 words, answer the following questions:
      i. How are decision tree classifiers different from random forests on a structural level?
      ii. Discuss their advantages/disadvantages over each other and where would you choose one over the other?
   (d) Perform a grid search on the random forest model parameters `'max_depth', 'max_features', 'min_samples_leaf', 'min_samples_split', 'n_estimators'` to evaluate the optimal values for these parameters. Report model parameters for your best classifier model.
   (e) Report all evaluation metrics described in the Model Evaluation section below.

   In no more than 50 words, explain your observations and your assessment of the classifier. You can use a text box (i.e. Markdown Cell) in Jupyter to write down your analysis.

4 Model Evaluation

For all models in 3A and 3B, you must report:

1. Classification metrics like accuracy, recall, F1 and micro/macro/weighted averaged precision/recall/F1 statistics.
2. ROC curves for each of the models.
3. Answer the following questions (with plots if needed):
   (a) Is accuracy well suited to this classifications task? Justify in less than 20 words.
   (b) Can you select an optimal model for this task from the ROC curves. Justify in less than 20 words.

These metrics will be discussed at the beginning of Week 5.

5 Grading

Component	Points
Decision Trees + Analysis	2
Random Forests + Analysis	2