FOUNDATIONS OF MACHINE LEARNING

M.SC. IN DATA SCIENCES AND BUSINESS ANALYTICS

CENTRALESUPÉLEC

Assignment 2 – Kaggle Challenge

Instructor: Fragkiskos Malliaros
Competition responsible: Sagar Verma

Due: **November 28, 2021 at 23:00**

Description of the Competition

We often face the problem of searching meaningful emails among thousands of promotional emails. This challenge focuses on creating a *multi-class classifier* that can classify an email into one of the four classes based on the metadata extracted from the email. **More details about the task are presented in the accompanying presentation file posted on piazza**. The data of the challenge and the submission website can be found at the following Kaggle website:

`https://www.kaggle.com/c/centralesupelec-ml-course/`

Summary of the Pipeline

The pipeline that will be followed in the project is similar to the one followed in the labs. Next we briefly describe each part of the pipeline. The pipeline is contained in the `skeleton_code.py` python script. The script describes a simple baseline model.

- *Data preprocessing*: After loading the data, a preprocessing task should be done to transform the data into an appropriate format.
- *Feature engineering and dimensionality reduction*: The next step involves the feature engineering task, i.e., how to select a subset of the features that will be used in the learning task (feature selection) or how to create new features from the already existing ones (see also previous section). Moreover, it is possible to apply dimensionality reduction techniques in order to improve the performance of the algorithms.
- *Learning algorithm*: The next step of the pipeline involves the selection of the appropriate learning (i.e., classification) algorithm for the problem. Here, you can test the performance of different algorithms and choose the best one (e.g., logistic regression, SVMs, decision trees, neural networks). Additionally, you can follow an ensemble learning approach, combining many classification algorithms.
- *Evaluation*: In the next section, we describe in detail how the evaluating will be performed. Keep in mind that the challenge corresponds to a multi-class classification task.

Evaluation

You will build your classification model based on the training data contained in the `train.csv` file. To do this, you can apply *cross-validation* techniques[1]. Of course, having a good model that achieves good accuracy under cross validation does not guarantee that the same accuracy will also be achieved for the test data. Thus, the final evaluation of your model, will be done on the test dataset contained in the `test.csv` file. After having a model that performs well under cross-validation, you should train the model using the whole training dataset (i.e., all the instances of the `train.csv` file) and test it on the test dataset as described below.

How to evaluate your model on the test dataset?

For the test data (`test.csv`) we do not have information about the class labels (type of each email), and thus the final assessment will be done in the Kaggle platform. Note that, the same preprocessing tasks that have been applied on the training data should also be applied on the test data. The evaluation process can be summarized as follows:

1. Run your model on the test data (`test.csv`).
2. Get the predicted class labels for each instance of the test dataset.
3. Create a new file, called `sample_submission.csv`, that contains the predicted class label for each instance.
4. Create an account on Kaggle (the same for each member of your team) and make a new submission by simply uploading the `sample_submission.csv`. Then, Kaggle will evaluate your predictions, and the evaluation score as well as your position (with respect to the rest users) will appear in the Leaderboard. Note that, you can submit up to 10 entries per day. Your final score will be the best one that you have achieved.

Grading

Grading will be on 100 points total. Each team should deliver:

- A submission on the Kaggle competition webpage. **40 points** will be allocated based on raw performance only, provided that the results are reproducible. That is, using only your code and the data provided on the competition page, we should be able to train your final model and use it to generate the predictions you submitted for scoring.
- A 3-pages report (see details below). Please ensure that both your real names and the name of your Kaggle team appear on the report.

The 3-page report should include the following sections (in this specific order):

- *Section 1: Feature engineering* **[30 points]**. Regardless of the performance achieved, we will reward the research efforts and creativity put into the feature engineering step (e.g., creation of new features). You are expected to:
    1. Explain the motivation and intuition behind each feature. How did you come up with new feature? What is it intended to capture? Did you discard other features?

[1] Cross-validation in *scikit-learn*: `http://scikit-learn.org/stable/modules/cross_validation.html`.

2. Rigorously report your experiments about the impact of various combinations of features on predictive performance, and, depending on the classifier, how you tackled the task of feature selection.

- *Section 2: Model tuning and comparison* **[20 points].** You are expected to:

  1. Compare multiple classifiers (e.g., SVMs, Random Forest, Boosting, Logistic regression, Nearest neighbors).
  2. For each classifier, explain the procedure that was followed to tackle parameter tuning and prevent overfitting.
  3. Report the cross-validated performance (on the training data) of the models you have explored, as well as the score on the test set (given by Kaggle).
  4. Discuss about any additional models that you have tested but did not perform well.

Report and code **completeness, organization and readability will be worth 10 points**. Best submissions will (i) clearly deliver the solution, providing detailed explanations of each step, and (ii) provide clear, well organized and commented code.

How to Submit

Please complete the second assignment in groups of **3-4** students (preferably, the same team as in the project of the course). No late assignments will be accepted.

1. **Kaggle submission:** submission of your solution in the kaggle platform (team submission – pick also a name for your team).
2. **Report:** *typeset* your report (**PDF** file only). **In your report, you should mention the name of your team in Kaggle, and the names of all team members.** The submission of the report (max **3 pages**) should be made on **gradescope** (Assignment 2; Entry Code: D5455N).
3. **Code:** prepare a .zip file (`code_name_of_your_team.zip`) containing the code that is needed to reproduce your submission. The file should be sent by email to: *centralesupelec.fmlclass@gmail.com*. The subject of the email should be: "[DSBA-FML] - Assignment 2 Code - Name of Your Team".