MLCS – fall 2024
Assignment 1

Instructor: Hans P. Reiser

Submission Deadline: Tuesday, Sept 03, 2024 – 23:59

1 Experiments with spam classification

As a basis for this assignment, you can use your code from the first practice assignment or the sample solution. The goal of this task is to compare different supervised learning approaches for the classification of spam messages and experimentally study the influence of hyperparameters on the quality of a classification.

- Use the preprocessed and presplit (training data and test data) data set from `https://www.dropbox.com/s/yjiplngoa430rid/ling-spam.zip`
- Remove the word "Subject:" from the subject line, remove all non-alphabetic characters, and remove all single-character words.
- Your goal is to compare a Naïve Bayes classifier, KNN, and Logisitic Regression.
- For each of the following experiments, you should obtain accuracy, precision, recall, F1-score, and the area under the ROC curve (scikit: `roc_auc_score`) as metrics, and plot the ROC curve.
- You should describe / implement your experiments such that they are *reproducible* by others (you do not have to include the data set in your submission).
- Submit your source code (Python) and a short report with the main results and any explanations you consider relevant.

1.1 Influence of the dictionary size (2 pt)

Using the Naïve Bayes classifier, experimentally check the influence of the dictionary size on the results, using a size $s \in \{100, 500, 1000, 2000, 3000\}$.

Documentation: Results (metrics) and a single graph with all 5 ROC curves

On this basis, select the best value for the dictionary size to be used in all following experiments.

1.2 Influence of the k parameter on KNN (2 pt)

Using the KNN classifier, experimentally check the influence of the value K, using values $K \in \{4, 6, 8, 10, 15, 20\}$

Documentation: Results (metrics) and a single graph with all 5 ROC curves.

On this basis, select the best value for K for the comparison in part 1.5.

1.3 Influence of hyper-parameters on Logistic Regression (2 pt)

This is a very complex topic that we can address only very superficially. There are many hyperparameters in LR, such as which solver is used, a “penalty” parameter, a maximum interation number. We limit our experiments to a regularization parameter (“C” in scikit-learn). This parameter ’C’ controls the strength of regularization in logistic regression, which helps prevent overfitting by adding a penalty term to the loss function based on the magnitude of the model’s coefficients.

Use the default values for all other parameters, and consider the influence of the “C” parameter with five significantly different values of your choice.

Documentation: Results (metrics) and a single graph with all 5 ROC curves.

On this basis, select the “best” value for C for the comparison in part 1.5.

1.4 Influence of hyper-parameters on Logistic Regression (II) (2 pt)

For the experiments with Logistic Regression, in addition to the performance metrics on the test data set, look at the performance parameters for the training set and for the test set (if a model performs well on the training set but bad on the test set, this is an indication of over-fitting/lack of generalization)

Documentation: A table comparing the recall and precision metric between test set and training set, for all 5 values of the “C” parameter.

1.5 Comparison (2 pt)

Compare the three approaches using the metrics and by plotting a single graph with the three ROC curves

1.6 Optional bonus points

- 1 point for voluntarily presenting your solution (this bonus point will be available for every assignment, but only once for each student)
- 2 points for adding another learning approach of your choice to the comparison (including presenting it in class)