Project 3: Machine Learning
COSC423/523, Fall 2021

Due: Nov 2, 11:59pm

I. Overview and Instructions

Machine learning models are computational methods that attempt to automatically find patterns in large data sources and re-use them for future decision-making. This project requires that you (1) implement and evaluate machine learning models and (2) produce a written report that details your evaluation approach and your observations.

Learning Objectives

The learning objectives of this project are geared to improving your ability to:

1. Write programs in the Python programming language.
2. Manage data in Python with Pandas dataframes.
3. Implement machine learning models with modern software libraries.
4. Understand documentation of machine learning libraries in Python.
5. Articulate observations from machine learning models.

This project is research-oriented. You will be evaluated on the implementation of your machine learning models, their evaluation, and what you have learned from the evaluation. In our lectures, you have been introduced to the process of building and evaluating machine learning models. This assignment serves as an opportunity to demonstrate your mastery of that knowledge.

II. General Information

In this project, you will implement two types of machine learning models: (1) Support Vector Machines and (2) Neural Networks. You will use one particular dataset for your implementation You should employ cross-validation. The settings and datasets are as follows:

Mushroom Dataset. This dataset involves a binary classification problem that asks you to classify mushrooms as "edible" or "poisonous". The first attribute is the target label ("p" or "e") and the other 22 are features. There are 8,124 instances in the data. Some instances have invalid data (e.g., blanks, question-marks, etc), making them inappropriate for use in a machine learnign context. For that reason, they should be removed. The dataset is available on Canvas.

III. Project Requirements

This project has several requirements related to (1) Language Version and Libraries, (2) Model Requirements, (3) Evaluation Requirements, (4) Program Structure, and (5) Report Requirements. The requirements for this project are as follows:

1. Language and Library Requirements

   - Your implementation must be written in Python 3.6 or higher.
   - You must use a compliant version of Pandas to manage your data
   - You must use a compliant version of Scikit-Learn to implement and evaluate SVMs.
   - You must use a compliant version of Keras to build and evaluate Neural Networks.

2. Model Requirements

   - Your must implement and evaluate SVMs and Neural Nets for classifying mushrooms.
   - You must explore all appropriate hyperparameters for both model types.
   - You must identify the best-performing set of hyperparameters for both types of models through a coarse grid search and a fine grid search.
   - You must use k-Fold cross-validation to evaluate your models. You must use a reasonable value for k.

3. Evaluation Requirements

   - You must present an evaluation of SVMs with Scikit-learn and Neural Nets with Keras.
   - Your evaluation should clearly report three metrics: Accuracy, Precision, and Recall.
   - You should create a Precision-Recall plot for your best performing SVM model.
   - You should create a Precision-Recall plot for your best performing NN model.

4. Program Requirements

   - Your program should be split across multiple files. Specifically, you should include the following:
     - **svm_search.py**: Implements and executes the grid search process for SVMs. Prints the accuracy of the best-performing SVM model followed by the accuracy of all explored model configurations.
     - **svm_best.py**: A static-implementation of the best-performing SVM model. Demonstrates the model's usage. Prints the accuracy of the model.
     - **nn_search.py**: Implements and executes the grid search process for NNs. Prints the accuracy of the best-performing NN model followed by the accuracy of all explored model configurations.
     - **nn_best.py**: A static-implementation of the best-performing NN model. Demonstrates the model's usage. Prints the accuracy of the model.
   - You should include a "README.md" file that provides an overview of your code. You should provide instructions for running your code locally.
   - Your code should be adequately documented. Specifically, you should include references for *all* relevant machine learning activities. There should be clear indicators for your coarse grid search and fine grid search. Each file should begin with documentation that includes your name as an author and provide a brief description of the file.

5. Report Requirements

- You must include a two-page report on your project. The report must include your name, the name of the course, and the semester. For both types of models, your report should include the following sections:
  - **I. Introduction.** Describe an overview of the dataset and the problem you are solving.
  - **II. Pre-Processing.** Any steps taken to "clean" the dataset before being used for machine learning, e.g. removing rows with missing data / unusual characters.
  - **III. Grid Search.** Articulate the explored hyperparameter configurations that your program evaluated and what was observed. Specifically report how k-Fold cross-validation was employed.
  - **IV. Best-Performing Models.** Describe the configuration of your best performing model and their accuracy. You should include both Precision-Recall plots for your best-performing models.
- You should conclude your written report with a section named "**V. Conclusion**" in which you briefly summarize what you learned from your process. Specifically state which type of model that you believe to be more effective.
- The report must be submitted as a PDF file.

IV. Grading and Submission

The assignment is worth 100 points. Create a ZIP file that includes all of your files. Upload the ZIP file to the Project 3 submission folder on Canvas by the due date. For questions regarding the late policy for assignment submission, please consult the syllabus.

The following grading scheme will be used for all students:

Requirement	**Point Value**
README exists and follows specification	10 points
Code is documented following specification	10 points
Model Requirements	10 points
Evaluation Requirements	20 points
Program Requirements	25 points
Report Requirements	25 points
Total	100

Note I: Failure to meet the Language and Library Requirements will result in an automatic grade of zero.

Note II: You should submit only the required files: 4 Python files, 1 Readme file, and your PDF report.