

## Assignment No. 5

EECS 658

Introduction to Machine Learning

Due: 11:59 PM, Thursday, October 31, 2024

Submit deliverables in a single zip file to Canvas

Files in other formats (e.g., .tar) will not be graded

Name of the zip file: FirstnameLastname\_Assignment5 (with your first and last name)

Name of the Assignment folder within the zip file: FirstnameLastname\_Assignment5

### Deliverables:

1. Copy of Rubric5.docx with your name and ID filled out (do not submit a PDF)
2. Python source code.
3. Screen print showing the successful execution of your Python code. (Copy and paste the output from the Python console screen to a Word document and PDF it).

### Assignment:

- I have created an imbalanced iris dataset called “imbalanced iris.csv” to use for this assignment which is located in the Assignment 5 folder.
- Use 2-fold cross-validation with the Neural Network machine learning model for each part.
- This assignment has three parts (listed below by number). Before each part’s printout, printout the part number.

#### Part 1: Imbalanced Data Set

- Print out and label the Confusion Matrix and Accuracy score using the imbalanced iris data set.
- Print out and label the calculated Class Balanced Accuracy as described in the Imbalanced Datasets lecture.
- Print out and label the calculated Balanced Accuracy as described in the Imbalanced Datasets lecture.
- Print out and label the balanced accuracy score calculated by the scikit-learn function `balanced_accuracy_score` as described in the Imbalanced Datasets lecture.

#### Part 2: Oversampling

- Use the imbalanced-learn toolbox (Imbalanced Datasets lecture) for each of the following:
  - Balance the imbalanced iris dataset with random oversampling and print-out and label the Confusion Matrix and Accuracy score. (We don’t need to use one of the balanced accuracy scores from Part 1 because the set is balanced now).
  - Balance the imbalanced iris dataset with SMOTE oversampling and print-out and label the Confusion Matrix and Accuracy score.
  - Balance the imbalanced iris dataset with ADASYN oversampling and print-out and label the Confusion Matrix and Accuracy score. Use the `sampling_strategy='minority'` parameter, or you will get an error message. (Note: the resulting dataset may still not be balanced, which is okay).

#### Part 3: Undersampling

- Use the imbalanced-learn toolbox (Imbalanced Datasets lecture) for each of the following:
  - Balance the imbalanced iris dataset with random undersampling and print-out and label the Confusion Matrix and Accuracy score.
  - Balance the imbalanced iris dataset with Cluster undersampling and print-out and label the Confusion Matrix and Accuracy score.
  - Balance the imbalanced iris dataset with Tomek links undersampling and print-out and label the Confusion Matrix and Accuracy score. (Note: the resulting dataset may still not be balanced, which is okay).

Rubric for Program Comments		
Exceeds Expectations (90-100%)	Meets Expectations (80-89%)	Unsatisfactory (0-79%)
Software is adequately commented with prologue comments, comments summarizing major blocks of code, and comments on every line.	Prologue comments are present but missing some items or some major blocks of code are not commented or there are inadequate comments on each line.	Prologue comments are missing all together or there are no comments on major blocks of code or there are very few comments on each line.

#### Adequate Prologue Comments:

- Name of program contained in the file (e.g., EECS 658 Assignment 1)
- Brief description of the program, e.g.,
  - Check versions of Python & create ML “Hello World!” program
- Inputs (e.g., none, for a function, it would be the parameters passed to it)
- Output, e.g.,
  - Prints out the versions of Python, scipy, numpy, pandas, and sklearn
  - Prints out “Hello World!”
  - Prints out the overall accuracy of the classifier.
  - Prints out the confusion matrix.
  - Prints out the P, R, and F1 score for each of the 3 varieties of iris.
- All collaborators
- Other sources for the code ChatGPT, stackOverflow, etc.
- Author’s full name
- Creation date: The date you first create the file, i.e., the date you write this comment

#### Adequate comments summarizing major blocks of code and comments on every line:

- Provide comments that explain what each line of code is doing.
- You may comment each line of code (e.g., using `//`) and/or provide a multi-line comment (e.g., using `/*` and `*/`) that explains what a group of lines does.
- Multi-line comments should be detailed enough that it is clear what each line of code is doing.

- Each block of code must indicate whether you authored the code, you obtained it from one of the sources listed in the prolog, or one of your collaborators authored the code, or if it was a combination of all of these.

Collaboration and other sources for code:

- When you collaborate with other students or use other sources for the code (e.g., ChatGPT, stackOverflow):
  - Your comments must be significantly different from your collaborators.
  - More scrutiny will be applied to grading your comments in particular explaining the code “in your own words”, not the source’s comments (e.g., ChatGPT’s comments).
- Failure to identify collaborators or other sources of code will not only result in a 0 on the assignment but will be considered an act of Academic Misconduct.
- Students who violate conduct policies will be subject to severe penalties, up through and including dismissal from the School of Engineering.