Assignment No. 4 Rubric

EECS 658

Introduction to Machine Learning

Due: 11:59 PM, Tuesday, October 22, 2024

**Student:**

**Student ID:**

# Point Breakdown

|  |  |  |
| --- | --- | --- |
| ***Graded Value*** | ***Points Possible*** | ***Criteria*** |
|  | 1 | Name of the zip file: FirstnameLastname\_Assignment4. Files in other formats (e.g., .tar will not be graded). |
|  | 1 | Name of the Assignment folder within the zip file: FirstnameLastname\_Assignment4 |
|  | 1 | Copy of Rubric 4.docx with your name and ID filled out |
|  | 3 | Python source code for CompareFeatureSelectionMethods |
|  | 3 | Screen print showing the successful execution of CompareFeatureSelectionMethods |
|  | 6 | For each Part, the confusion matrix matches the accuracy metric. |
|  | 6 | The values in the 4 confusion matrices each add up to 150 |
|  | 6 | The final features are listed for each Part. |
|  | 6 | For Part 3, the subset of features, accuracy, Pr[accept], Random Uniform, and Status is printed out for 100 iterations. |
|  | 6 | For Part 3, the Status (i.e., Improved, Accepted, Discarded, or Restart) is correct for each iteration |
|  | 6 | For Part 4, the 5 best sets of features and their accuracy are printed out for 50 generations. |
|  | 6 | For Part 4, the 5 best sets of features for the first generation are not the same as those for the last generation. |
|  | 3 | Answer to 4 is correct. |
|  | 3 | The answer to 5a is correct based on the accuracy metric for each of the Parts. |
|  | 3 | Answer to 5b is correct for 2nd best dimensionality reduction method |
|  | 3 | Answer to 5b is correct for 3rd best dimensionality reduction method |
|  | 3 | Answer to 5c is correct based on the results of Parts 1, 2, 3, and 4. |
|  | 3 | Answer to 5d is correct based on the results of Parts 2 and 3. |
|  | 3 | Answer to 5e is correct based on the results of Parts 2 and 4. |
|  | 3 | Answer to 5f is correct based on the results of Parts 3 and 4. |
|  | 25 | Software is adequately commented. |
|  | **100 pts** |  |

|  |  |  |
| --- | --- | --- |
| **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.

# Grader Comments