Department of Electrical & Computer Engineering (ECE), Concordia University

APPLIED MACHINE LEARNING & EVOLUTIONARY ALGORITHMS

COEN 432/6321 - Fall 2024

Assignment 2 (/Project): ML for Cancer Diagnosis (due date: 29 Nov @ 23hr55, via Moodle)

Problem Description. You are given a set of 569 instances, each with information about a patient's cell, that is or is not cancerous. The information includes the patient's ID (which should not be relevant to his/her diagnosis) and a diagnosis (malignant or benign), as well as a set of 30 attributes that are usually used for diagnosis. You are asked to design, implement, and train a machine learning model, and assess its quality in terms of (a) accuracy of diagnosis (of instances not used for training) and (b) computational efficiency and scalability of your solution. Your program shall accept data from a file of instances. It is up to you to correctly use these instances (or a large enough subset of them) to train and test (validate) one machine learning (ML) algorithm of your choice (such as an EA or kNN or a Decision Tree), one that you think is appropriate for the problem at hand, one that is distinguished by:

(1) Instances with features that include rational (numerical) values, and a target variable whose value is M or B;
(2) A *large* search space due to the large number of features;
(3) A *complex* relationship between the values of the features and the value of the target variable.

Description of Ideal Solution. Use N = 40, 140, 240, 340, 440, the ratio N/T must be maintained at 4/1. An instance used for training, in a given train-and-test run, cannot be used for testing. You must retrain the model prior to each test. Further, the (N+T) instances must be chosen at random from the complete set. Though this is not the case in medical practice*, implement accuracy as the percentage of test instances (T) that are correctly predicted by your trained model (i.e., predicted diagnosis = actual diagnosis). For running time, just use actual execution time (of testing, not training), while running your program on the same computer, under identical conditions. [If that's not possible or if you prefer, you may use a counting method, inserting a simple counter at an appropriate point within your program.] *If you're interested, see https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4614595/

Submission Instructions. You must submit a ZIP file that includes **an output file** containing your **brief report** and **a folder with your program**. Name your ZIP file exactly "Assignment2". ONLY SUBMIT ONE ZIP FILE PER INDIVIDUAL/TEAM.

The program must be in Python (preferable), C++, or Java.

Place the names and IDs of all team members on the first line of each file (commented). The program must be able to read any number of instances (for training purposes) from the original instances' file, train ML and test it on other instances then produce an output file with the instances and their predicted accuracy as well as overall prediction accuracy. The report must contain one paragraph of brief analysis + graphs (consider using two methods among Precision, Recall, F1 Score, and ROC-AUC, as evaluation metrics, reference: https://towardsdatascience.com/a-look-at-precision-recall-and-f1-score-36b5fd0dd3ec).

If the submission is in Python, include all your .py files in a folder called "Ass2Python". If the submission is in C++, you must submit your cpp and header files. Include them in a single folder named "Ass2C++". If the submission is in Java, include all your .java files in a folder called "Ass2Java".

Marking Scheme**. If a program does not run then it will not be marked (i.e., you will receive 0).**

	Excellent (100%)	**Good (80%)**	**Satisfactory (60%)**	**Unsatisfactory (<40%)**
Correct Implementation of Machine Learning Algorithm + Validation Accuracy Assessment Method 30+30 = 60 points	Perfectly Correct implementation and evaluation (no errors)	Minor errors that do not invalidate your methods or their results	Significant errors that do compromise the correctness of the either methods or their results	Unsatisfactory methodology and/or invalid assessment of results
In-line Documentation 20 points	Very clear	Sufficient	Unclear	Missing
Software Usability 20 points	Easy	Minor challenges	Requires effort to run	Unusable (by a non-techie)

Graduate Students (**Project** related additional requirements and report). **In *addition* to** the deliverables required for A2, use at least two different ML models, you may use **any** method to optimize the parameters such as an EA or (iterative) grid search, optimize both models to get the best possible performance (validation accuracy) and compare the results of the optimized models. You need to show that, in fact, you have been able to improve validation accuracy using either one of these two approaches. As such, your deliverables must also include a report (**one PDF file**) of 3-4 pages in length (total) and not longer, that includes the following sections:

(A) ***Problem description:*** a 1-2 paragraph clear description of the problem.
(B) ***Methods description:*** high-level flow-chart of the two processes of model building (training) and model evaluation (validation accuracy), with pseudo-code descriptions of the main parts of that flowchart. See https://www.codecademy.com/article/pseudocode-and-flowcharts. Do not copy and paste parts of your code.1-2 pages is sufficient.
(C) ***Results & Conclusions:*** presentation of the results, the original training and validation accuracies, and a figure that exhibits either (a) the temporal progress of best fitness (= validation accuracy) of the population of machine learning models (representing different feature sets) or, (b) in the case of parameter optimization or grid search, a list of the various validation accuracies for the different combinations of parameter values, to show that you made an effort to improve accuracy. Every result should be discussed, meaningfully, but briefly. 1-2 page(s) is sufficient.

Please make sure that your report has a title and the names of the authors with IDs. The report must be readable (sound English) and formatted, following a well-known standard (IEEE, https://www.ieee.org/conferences/publishing/templates.html).

Timeliness. Up to **2 days** of delay in submission leads to no discount of your mark; take any longer than 2 days, and we will not mark down your submission. The same deadline applies to both A2 and Project.

For *clarifications* of the content of the assignment or submission procedure, e-mail the TA (zhiyangdeng.30@gmail.com)