The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

COMP3009 Machine Learning 2022-23

Assignment 2

Machine Learning for Disease Treatment Response Prediction

Dr Xin Chen

1. Introduction

This assignment assesses your practical skills of applying machine learning methods to a real-world problem. The implementation will be based on Python and third-party Machine Learning libraries. Same as assignment 1, you will **have to** work in the same group, handing in a single report and code by **13th December, 2022 at 3 pm** UK time on Moodle by **member 1** of each group. You can split and distribute the work to individual members, but each individual is expected to understand every aspect of the work.

2. Background

Breast cancer is the most common cancer in the UK for women. Chemotherapy is a commonly used treatment strategy to reduce the size of locally advanced tumour before surgery. However, chemotherapy is a toxic process to human body and it is not aways effective to everyone. Complete tumour resolution at surgery, known as **pathological complete response** (**PCR**), has a high likelihood of achieving cure and longer **relapse-free survival** (**RFS**) time. RFS is the length of time after primary treatment for a cancer ends that the patient survives without any signs or symptoms of that cancer. However, only 25% of patients receiving chemotherapy will achieve a PCR, with the remaining 75% having residual disease and a range of prognosis. Better patient stratification and treatment could be achieved if PCR and RFS could be predicted using information prior to chemotherapy treatment.

3. Aim

You are asked to use advanced machine learning method to predict **PCR (classification)** and **RFS (regression)** using both clinically measured features and features derived from magnetic resonance images (MRI) prior to chemotherapy treatment.

4. Data

Based on the public dataset from The American College of Radiology Imaging Network (I-SPY 2 TRIAL), a dummy (simplified) dataset is generated for this assignment. Each patient in this dataset contains 10 clinical features (Age, ER,

PgG, HER2, TrippleNegative Status, Chemotherapy Grade, Tumour Proliferation, Histology Type, Lymph node Status and Tumour Stage) and 107 imaged-based features. The image-based features were extracted from the tumour region of MRIs using a radiomics feature extraction package (known as Pyradiomics: https://pyradiomics.readthedocs.io/en/latest/). You do not need to understand the meaning of these clinical feature and image-based features to complete this assignment. **"999" in the spreadsheet means a missing data value**. A training dataset (**trainDataset.xls**) is provided and available on Moodle that contains 400 patients. A test dataset that contains *N* patients is reserved (hidden from you) for final performance evaluation. You can assume that the test set and training set are sampled from the same data distribution.

5. Implementation Requirement

You are asked to build a machine learning model for each of the PCR (classification) and RFS (regression) predictions. You need to consider and implement methods for data pre-processing (e.g. how to handle missing data, outlier, normalisation, etc, if needed), feature selection, machine learning modelling, hyperparameter tuning (if applicable) and method evaluation. There is no restriction or requirement of the selection of methods. However, you will likely need to compare a few methods to pick the best one with the best parameter setting.

Your code will be finally tested on a reserved test set by the lab assistants or module convenor after your code is submitted. An example test file is provided (**testDatasetExample.xls**) that only contains 3 examples. It is your responsibility to ensure your code can run on a test file in a similar format but contains more patients. You must name your final test code "FinalTestPCR.py" or "FinalTestPCR.ipynb" for PCR prediction, and "FinalTestRFS.py" or "FinalTestRFS.ipynb" for RFS prediction, so that they can be tested on the test dataset. The code for method development needs to be in a separate file, not in the "FinalTestXXX" file. The output of the "FinalTestXXX" must be a spreadsheet (.cvs or .xls) that contains the predicted outcome for each tested patient (i.e. the first column is the patient ID, the second column is either the predicted PCR or RFS outcome). Classification accuracy will be used to evaluate PCR prediction. Mean Absolute Error will be used to evaluate RFS estimation.

All implementations need to use Python programming language. Any machine learning libraries are allowed (e.g. Scikit-learn, Scipy, Pandas, Tensorflow, Pytorch, etc.). **However, any autoML based package (e.g. accepting the raw data and automatically select the best ML method and optimise the parameter for you) is NOT allowed**.

6. Assessment

Assignment 2 occupies 80% of the coursework mark (i.e. 24% of the whole course mark). The marking will be performed based on the objective performance on the test set, quality of code and quality of technical writing. A

detailed marking criteria is provided in section 8. A single mark and feedback will be given for each group. The final mark for individual student will be calculated based on the contribution table as described in section 7.

7. Deliverables

For the completion of Assignment 2, the following have to be submitted on Moodle. Only one report (.pdf) and one zipped code file need to be submitted per group.

1. The Python code for implementing the two tasks (PCR and RFS prediction). Besides the code for method development, two files "FinalTestPCR" and "FinalTestRFS" must be included for testing the test set.

2. A report in the format of an IEEE conference paper. Technical paper writing will be introduced in one of the lectures. A template of the required format will be provided in word and Latex. Based on the given format, a maximum of 4 pages is allowed, excluding references (references can be on the 5th page).

3. At the end of the paper (excluded from the 4 pages), the following contribution table needs to be completed and agreed by all members, which will be used to calculate individual student's final mark.

Task and Weighting	**Data pre-processing (10%)**	**Feature Selection (20%)**	**ML method development (30%)**	**Method Evaluation (10%)**	**Report Writing (30%)**
Name of member 1	30%	15%	20%	20%	20%
Name of member 2	0%	25%	30%	0%	20%
Name of member 3	30%	20%	20%	10%	20%
Name of member 4	0%	10%	30%	30%	20%
Name of member 5	40%	30%	0%	40%	20%

The percentage of contribution in the above table is an example, which will be different for each group dependent on the true contribution of each member. However, the task names and the their weighting should be unchanged, and the sum of the contributions from all members for each task (i.e. each column) should be 100%. Note that each student can contribute to multiple tasks and each task can involve multiple students.

8. Marking Criteria

Elements	% mark
Performance on test set (objective)	30%
Code quality (e.g. comments, easy to read, robustness, etc)	10%
Description of Method	20%
Explanation and presentation of the results obtained	15%
Discussion of the strengths and weaknesses of the chosen method	15%
Scientific writing and clarity	10%

9. Common Q&As

- **What is the performance of each task we are expecting to achieve?**

  It is a real-world dataset for a challenging clinical task, hence I don't have an estimation of performance. However, a >90% classification accuracy is too good to be true for this task. For the RFS estimation is even more challenging. The performances are expected to be vary across groups.

- **Why don't we use anonymised peer-assessment form to score the contribution of each member?**

  Anonymised peer-assessment form was used in previous years. Occasionally, members stab each other from behind(😊), and it may involve several rounds of interviews to settle the final percentage of contribution. Hence, it is changed to a more transparent and quantitative contribution table this year.

  You may split the tasks and agree on the percentage of contributions at the very beginning, then add/minus the percentage dependent on the final delivery person and quality. Therefore, no surprises when you see your individual mark. Remember that each group is a team rather than competitors. An ideal case for a group of 5 students is that each member contributes to ~20%, but I don't expect it happens for most groups. Please split the tasks dependent on the your group experience from Assignment 1.

Plagiarism check will apply, meaning that high similarities across different groups are not expected. Late submissions in each assignment will result in 5% penalty per day (days rounded up to the next integer).