University of Lincoln

School of Computer Science

2019 – 2020

Assessment Item 2 of 2 Briefing Document

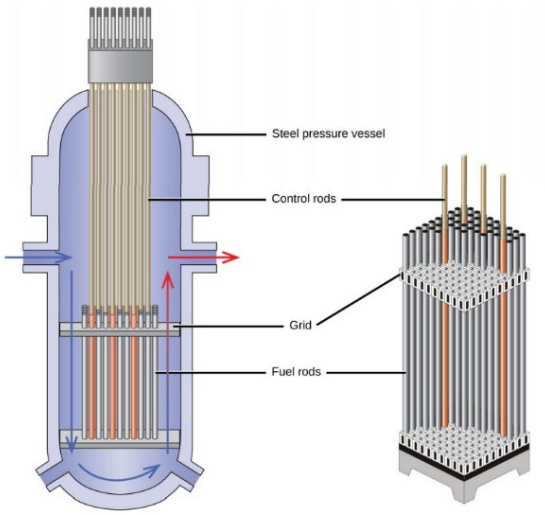
Title: CMP3751M Machine Learning	**Indicative Weighting: 50%**

Learning Outcomes

On successful completion of this component a student will have demonstrated competence in the following areas:

- LO1 Critique and appraise the scope and limits of machine learning methods by identifying their strengths and weaknesses
- LO2: Using a non-trivial dataset, plan, execute and evaluate significant experimental investigations using multiple machine learning strategies

Task Overview: Classification of Pressurised Water Reactor Status



The objective of this assignment is to analyse a dataset concerning pressurised water reactor data, specifically on the properties involved in the fuel assemblies cluster vibrations, alterations of thermal and hydraulic parameters, etc. For over 70 years, the nuclear power industry – in the UK and worldwide – has primarily focused on the technological evolution of reliable nuclear power plants to produce electricity. By monitoring pressurised water reactors (a type of nuclear reactors) while running at nominal conditions, it is possible to collect valuable insight and extract knowledge for early detection of abnormal events. Various types of fluctuations and perturbations can be caused by the turbulent nature of flow in the core, mechanical vibrations within the reactor, the boiling coolant and stochastic character (random noise). The dataset can be downloaded from Blackboard. It is based on data from a research project that investigates how to detect abnormal behaviours and events in pressurised water reactors. The dataset includes two classes (normal/abnormal condition) and a number of features, which will need to be summarised. The class membership of each row is stored in the field 'Status'. Our task is to develop a set of classification models for automatically classifying reactors as normal or abnormal, based on their parameters/features. No prior knowledge of the domain problem is needed or assumed to fulfil the requirements of this assessment whatsoever.

You need to write a report that discusses how you completed the tasks and go into enough depth to demonstrate knowledge and critical understanding of the relevant processes involved. 100% of available marks are through the completion of the written report, with clear and separate marking criteria for each required report section.

Feature information in the dataset include:

- Various vibration measurements in different parts of the reactor
- Various pressures in different parts of the reactor
- Power levels in different parts of the reactor

Status refers to the condition of the nuclear reactor, or in other words, we consider this to be our label/annotation for the sake of all implementations.

Unit of measurement or range of values of each feature are not relevant. However, features can be at different scales and/or measured in different units.

Report Guidance

Your report must conform to the below structure and include the required content as outlined in each section. Information on specific marking criteria for each section is available in the accompanying CRG document. You must supply a written report containing four distinct sections that provide a full and reflective account of the processes undertaken.

Section 1: Data import, summary, preprocessing and visualisation (20%)

As a first step, you need to load the data set from the .csv file into a Python IDE. You should then provide a summary of the dataset (e.g. mean values, standard deviations, min/max values, etc. for each feature) and proceed to data preprocessing. For example, what is the size of the data? How many features are there? Are there any missing values? Are there any categorical variables? Shall we normalise the data before starting training/testing any model?

To visualise the data, you need to generate two plots. The first one shall be a box plot, which will include the two classes ("Status"), i.e. normal/abnormal, in the x-axis and the "Vibration_sensor_1" in the y-axis. The second one shall be a density plot for the feature "Vibration_sensor_2", with the graphs of both classes appearing in the same plot. What information can I obtain from each of these two plots? Can one use any of these two plots to identify outliers? If yes, please elaborate.

Please include your code alongside the plots and explanation.

Hint: You can use available libraries in Python, e.g. pandas.

Section 2: Discussion on selecting an algorithm (30%)

A nuclear power plant is planning to use the dataset that has been provided to you to train a classifier to aid nuclear safety engineers to monitor and detect whether a nuclear reactor is working normally or abnormally. This would be very useful as those parameters are constantly monitored and stored, and therefore can be used to classify different reactors' status. A Machine Learning intern is asked to select the best performing model among many trained models (e.g. many types of classifiers have been trained e.g. KNN, SVM, Neural Networks, etc.). The intern used 70% of the data as training set, another 20% as validation set, and finally a 10% as test set. The intern trained 10 different models

(either by selecting a subset of the available features or by using a different type of classifier) and recorded the accuracy on the test set. Intern's best performing model achieves 90% of accuracy. Intern concludes that this model is the best one to use. Would you agree? Why? Please explain and elaborate.

Section 3: Designing algorithms (30%)

You will now design an artificial neural network (ANN) classifier for classifying pressurized water reactors as normal or abnormal, based on their underlying working conditions. You will use the provided data set to train the model. To design an ANN, use a fully connected neural network architecture with two hidden layers; use the sigmoid function as the non-linear activation function for the hidden layers and logistic function for the output layer; set the number of neurons in the hidden layer to 500. Now randomly choose 90% of the data as training set, and the rest 10% as test set. Train the ANN using the training data, and calculate the accuracy, i.e. the fraction of properly classified cases, using the test set. Please report how you split the data into training and test sets. In addition, please report the steps undertaken to train the ANN in detail and present the accuracy results.

** Bonus: You can try different number of epochs to monitor how accuracy changes as the algorithm keeps learning, which you can plot using the number of epochs in the 'x' axis and the accuracy in 'y' axis.**

Now use the same training and test data to train a random forests classifier. Set a) number of trees = 1000 and b) minimum number of samples required to be at a leaf node = {5 and 50}. Please report the steps for training random forests and show the test set accuracy results.

** Bonus: You can play with tweaking the number of trees and monitor how the performance changes as more trees are added, e.g. 10, 50, 100, 1000, 5000 trees and so on.

Section 4: Model selection (20%)

You have designed ANN and random forests classifiers with almost fixed model parameters. The performance of the model could vary when those model parameters are changed. You would like to understand, which set of parameters are preferable, and also to select the best set of parameters given a range of options. To do so, one method is to employ a cross-validation (CV) process. In this task, you are asked to use a 10-fold CV. As a first step, randomly split the data into 10 folds of nearly equal size, and report the steps undertaken to do this.

For ANN, set the number of neurons in the hidden layer to 50, 500, and 1000. For random forests, set number of trees to 20, 500, and 10000, with the "minimum number of samples required to be at a leaf node" chosen by you (or otherwise default value, if you opted to use the pre-built library and not develop it yourself; either way report what this value is).

Please do the following tasks for both methods:

a) Use the 10-fold CV method to choose the best number of neurons or number of trees for ANN and random forests respectively. b) Report the processes involved when applying CV to each combination/model. c) Report the mean accuracy results for each set of parameters, i.e. for different number of neurons and different number of trees accordingly. Which parameters should we use for each of the two methods, i.e. specifically for ANN and random forests?

Until now, you have selected the best parameters for each method, but we have not decided yet, which the best model is. With the results you have had so far, which one is the best model across all combinations of ANNs and random forests for this data set? Please discuss and justify your choice, reflecting upon your knowledge thus far.

Useful Information

Keep in mind that:

- The report must contain your name, student number and module name
- The report must be in PDF and **no more than 20 pages** in total (including reference list). No cover page required
- The report must be formatted in single line spacing and use an 11pt font
- The report should not include this briefing document
- Please describe and justify each step that is needed to reproduce your results by using code-snippets, screenshots and plots. When using screen shots or plots generated from Python please make sure they are clearly readable
- You interpret the results of your data analysis and model developments
- Explain trends, characteristics or even outliers when you summarise and describe data
- Always refer to accuracy as performance metric when reporting the 'performance' of the algorithm
- Should you decide to use prebuilt python libraries, such as scikit-learn, rather than implementing them yourself, you will need to provide extra justification for the steps undertaken to arrive to the conclusions, and also demonstrate adequate understanding of the algorithms. Analytical approach is required to arrive to credible and justifiable solutions

Submission Instructions

The deadline for submission of this work is included in the School Submission dates on Blackboard. You must make an electronic submission of your work to Blackboard that includes the following mandatory item:

- **A PDF of your written report (following the requirements above), submitted to the Turnitin upload area for assessment 2**

This assessment is an individually assessed component. Your work must be presented according to the School of Computer Science guidelines for the presentation of assessed written work. Please make sure you have a clear understanding of the grading principles for this component as detailed in the accompanying Criterion Reference Grid. Your citations and referencing should be in accordance with University guidelines. If you are unsure about any aspect of this assessment item, please seek the advice of the delivery team.