COMP80131 Assignment #1

Project: Machine Learning for Nuclear Graphite

# Task and Background

Background

The effects of cracking within nuclear graphite has traditionally been modelled using complex finite element analysis (FEA) methods such as the cracked brick neighbourhood array (CBNA) approach [[[1]](#endnote-1)]. These methods allow the calculation of important metrics, such as loads and stresses within the graphite structure as a function of input metrics, such as crack position/orientation or temperature. Major drawback of these models is their complexity; hence, they are difficult to produce and require engineering expertise. Additionally, these models are also computationally expensive and so require significant time and computer resources to complete.

Method Proposed

Machine learning [[[2]](#endnote-2)], a method of computationally training a computer algorithm to follow patterns in data, could be used to approximate the results produced by complex FEA models. The intention is to reduce the complexity and computational intensity of FEA methods yet provide comparable functionality and accuracy. A machine learning model will be produced that will be trained on existing FEA analyses and evaluated against their results. The model will be of the linear regression type to produce a continuous output value (such as stresses or displacements in the graphite).

# Input Data and Evaluation Metrics

Training Data

Training data, upon which the machine learning model will be trained, will be taken from existing FEA model results which are available from published papers. The input metrics, known as features in machine learning, will include positions/types/orientations of cracked graphite bricks, core temperature/age, graphite irradiation/weight loss, loads/pressures and material properties. The values that the model will be trained to produce, known as labels, will be peak stresses or crack displacements.

Evaluation and Testing

The features and label data taken from existing models will be split into two groups, selected from the global population at random. These are:

1. The training instances (80%): these will be used to train the model. The model will be provided the features and labels from the data. Linear Regression will be used to generate numerical weights for each feature which then fit the trend of the data.
2. The testing instances (20%): the features from this group will be entered into the model produced by 1.), which in turn will produce output labels. These will be compared against the original values and so an accuracy value generated.

The randomised group splits will be produced using a pseudorandom number generator: the seed used to generate each split can be recorded and reported, allowing the results to be repeated/reproduced. The original dataset will also be reported, including features and labels.

Methods and Processes

The training and analysis will be performed using statistical regression. The specific tools used will be linear regression and support vector regression. The SciKit Learn tools which are available in Python programming platform will be used.

The process of splitting the data into training and testing groups as described in the previous section will be repeated numerous times, each time taking a different random sample. The training and testing process will be performed on each one, providing an accuracy score.

# Analysis and Results

A distribution of the results generated, and statistical judgments can then be made. For example, if the mean of the machine learning accuracy is within 10% of the FEA result.

Another important metric is computational intensity and

1. Development and Application of a 3D Array Finite Element Model of Interacting AGR Graphite Core

   Components - Huaguo Teng – August 2015 [↑](#endnote-ref-1)
2. What is Machine Learning? <https://emerj.com/ai-glossary-terms/what-is-machine-learning/> accessed last 13/12/18 [↑](#endnote-ref-2)