COMP6013 Explainable Approaches to Machine Learning
Assignment, S1 2024
@ Computing, Curtin University

Weighting:
This assignment contains 100 points, which weight for 40% of the final mark.

Submission:
You need to submit your prediction to Kaggle (see Section 5.1). You also need to submit everything in **a single ZIP** file to Blackboard. Name the file as `<studentID>_<name>_assignment.zip`. The due date is **05 May 2024 11:59 PM**.

Academic Integrity:
This is an **individual** assignment so that any form of collaboration is not permitted. This is an **open-book** assignment so that you are allowed to use external materials, but make sure you properly **cite the references**. It is your responsibility to understand Curtin's Academic Misconduct Rules, for example, post assessment questions online and ask for answers is considered as contract cheating and not permitted.

1 Introduction

All over the world, large amounts of oil and gas products consisting of hazardous materials are transported every day in different ways. These materials often need be transported through populated areas, thereby increasing the potential risk to surrounding structures and people. In particular, the Liquefied Petroleum Gas (LPG) (i.e. the mixture of liquefied propane, butane, and other hydrocarbons) transportation by road is extremely intense in industrialized countries. Therefore, public concerns of the risk posed by these transportation activities have been growing [1].

Boiling Liquid Expanding Vapour Explosions (BLEVEs) are extreme explosions driven by nonlinear physical processes associated with explosively expanded vapour and flashed liquid. Blast loading generated from BLEVEs may severely harm structures and people. Prediction of such strong explosions is not currently feasible using simple tools [2].

Therefore, we are going to attempt using data-driven machine learning approaches to address this problem by predicting pressure generated by blast waves.

2 Problem Description

In this assignment, you will perform predictive analysis for the peak pressure generated by BLEVEs. The BLEVE happens in a rectangular tank located in the 3D environment and a rigid wall is placed with some distance to the source of BLEVE acting as building structure. The wall obstacles the blast wave and produces reflections and deflections of the energy (Figure 2), which makes it an even more complicated nonlinear problem.

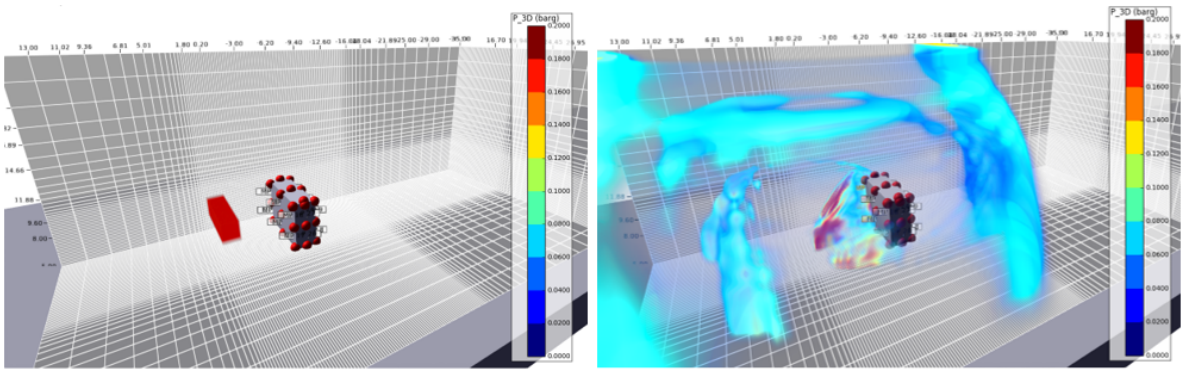

Figure 1: BLEVE blast wave propagation in an obstacle environment.

The goal is to predict the peak pressure around the obstacle. 27 sensors (monitoring points) are placed around each side wall of the obstacle, including 9 sensors front, 9 sensors back, and 9 sensors side, as shown in Figure 2. The problem is simplified by sampling training data from these sensor points and the testing will be on these points as well.

The 3D environment is characterised by many physical measurements related to the BLEVE, including temperature, pressure, gas and liquid ratio, tank size, obstacle size and so on. The complete list of features is as follows:

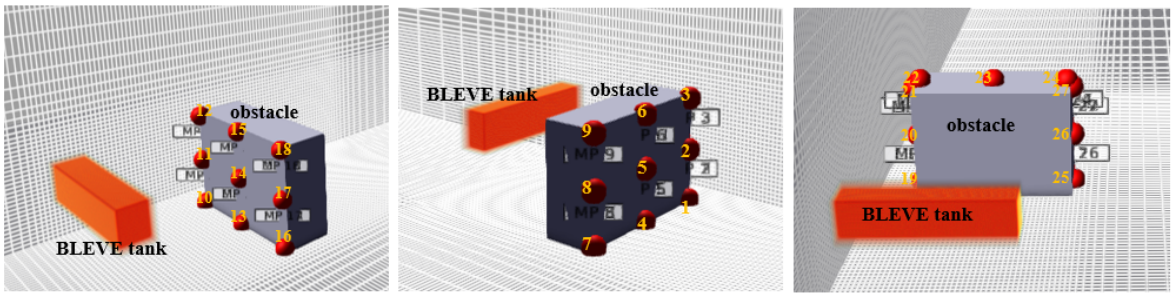

Figure 2: Sensor points on the obstacle. Left to right: font wall, back wall, and side wall.

- `Tank Failure Pressure`: the pressure within the tank when BLEVE happens (in bar)
- `Liquid Ratio`: the ratio of liquid in the tank (liquid and vapour coexist)
- `Tank Width`: the width of tank (in meter)
- `Tank Length`: the length of tank (in meter)
- `Tank Height`: the height of tank (in meter)
- `Vapour Height`: the height of vapour in tank (in meter)
- `Vapour Temperature`: the temperature of vapour (in K)
- `Liquid Temperature`: the temperature of liquid (in K)
- `Obstacle Distance to BLEVE`: the distance of obstacle to BLEVE (in meter)
- `Obstacle Width`: the width of obstacle (in meter)
- `Obstacle Height`: the height of obstacle (in meter)
- `Obstacle Thickness`: the thickness of obstacle (in meter)
- `Obstacle Angle`: the angle between the line connecting obstacle centers and BLEVE centers and the horizontal line
- `Status`: the status of liquid, either subcooled or superheated
- `Substance Critical Pressure`: the pressure required to liquefy a vapour of the substance at its critical temperature (in bar)
- `Substance Boiling Temperature`: the temperature above which liquid of the substance turns into vapour at atmosphere pressure (in K)
- `Substance Critical Temperature`: the temperature above which vapour of the substance cannot be liquefied, no matter how much pressure is applied (in K)

- `Sensor ID`: the ID of the sensor ranging from 1 to 27
- `Sensor Position Side`: the side of the wall where the sensor locates
- `Sensor Position x`: the x coordinate of the sensor
- `Sensor Position y`: the y coordinate of the sensor
- `Sensor Position z`: the z coordinate of the sensor
- `Target Pressure`: the target peak pressure to be predicted (in bar)

You are provided with `train.csv` and `test.csv` which contain the training set and testing set of the data. The ground truth of training set is given but not for the testing set. You are going to train a machine learning model with the training set and evaluate your model's performance on the testing set.

3 The Tasks

In this assignment, you will have three main tasks, including data preprocessing, model development, and model interpretation.

3.1 Data Preprocessing

Data preprocessing is vital for machine learning. You can consider using the following data preprocessing techniques:

- **Feature Selection**: some features may has little or no correlation with the target, and they can be probably removed. More in general, a "sparse" model can be trained with the most important features
- **Feature Engineering**: you do not have to restrict yourself in the set of features provided. You can create new features on your own! E.g., the ratio $\frac{\texttt{Tank Width}}{\texttt{Tank Length}}$ can be considered as another feature to your model
- **Data Type Conversion**: depends on the model you use, you may need to convert features to the suitable data type. E.g., you may want to convert categorical features to numeric with certain encoding methods
- **Feature Scaling**: you may use normalization, standardization, or whitening to scale your data
- **Data Augmentation**: you may increase data instances using augmentation techniques
- **Others**: any other data preprocessing techniques

3.2 Model Development

In this task, you will be doing model selection, i.e., finding a machine learning model that is best suited for the given regression problem, in terms of the prediction performance. You will examine **at least three different machine learning models** and compare their performance on the provided BLEVE data using **at least two different regression metrics**. You will adopt the model with the highest accuracy or lowest error.

- **Model selection**: you need to examine at least three different machine learning models, such as linear models, support vector regression, random forest, xgboost, neural networks, or any other models you think it is suitable.
- **Hyparameter tuning**: note that for any machine learning model there are several hyperparameters that are critical for the model's performance. You will need to tune these hyperparameters carefully to get good results.
- **Evaluation metrics**: the compulsory evaluation metrics for this assignment are MAPE (Mean Absolute Percentage Error) and R^2, which are both available in Scikit-learn library. You can use additional metrics, such as RMSE, MAE, etc.

Once you are happy with the model you trained, you can apply it on the test set to get predictions. A Kaggle competition is available for you to evaluate the performance (see Section 5.1).

3.3 Model Interpretation

We have trained a machine learning model and now we want to interpret our model to understand its behaviour. In particular, we are interested to know the following global interpretation:

- **Feature effect plot**: a summary plot showing the contribution of features to the prediction
- **Partial dependency** between target and features: if you're using dependency plot, you do not have to show plot for all features but **up to five "significant" plots** - those that has the largest effect or abnormal ones that contain much information)
- **Feature importance plot**: you need to present a complete feature importance plot for all features you used, including those you created if there is any
- **Feature interaction plot**: you need to identify **2 pairs of features** that are most correlated with each other in terms of the prediction task

We are also interested to interpret individual predictions for several critical data instances. You need to use local interpretation techniques, such as LIME or SHAP, to explain the following predictions in the **training set**.

- **The lowest prediction**: identify the data instance that has the lowest prediction of pressure, and explain why
- **The highest prediction**: identify the data instance that has the highest prediction of pressure, and explain why
- **The largest error**: identify the data instance that has the highest prediction error (in terms of absolute percentage error $e = |\frac{y-\hat{y}}{y}|$), and explain why

Note that for each of the interpretation method you used or any plots you obtained, you need to demonstrate your findings, i.e., *"what knowledge or insight do we get from the plot?"*

4 Python Environment

You will use `Python` for this assignment and you can use any library you like. You can conduct experiments with your local python environment but the final submission has to be a **Jupyter Notebook that can be run on Google Colab**. Note that the notebook you submitted should contain markdown cells that contains your interpretation, code cell comments, and/or anything else useful for understanding your work. That is, if a code cell is producing the feature effect plot, then a markdown cell should appear next to explain the plot. The plot should be included when you save the notebook. Colab does save cell outputs by default. If you are not sure, double-check the notebook setting and make sure the "Omit code cell output when saving this notebook" is disabled.

5 Submission

5.1 Kaggle submission

A private Kaggle competition is created for this assignment. You can make use of Kaggle to monitor how your model is performing on the test set. Specifically, the test set is split into two halves, and one is used on the public leaderboard while the other half is used on the private leaderboard.

The public leaderboard is used for you to monitor the model performance on the test set. Before the deadline of the assignment, you can submit your prediction to Kaggle and you will see your model's score in MAPE on the public leaderboard. You will also see others' scores. Note that this is by no means "a competition" but mainly for you to get an idea about how you are doing. On Kaggle, you can use any nickname to be anonymous but if you need to provide the nickname in your submission, ideally at the top of your notebook. You can make five submissions a day.

At the assignment due, Kaggle will use your best prediction (based on your public leaderboard scores) to obtain the final score on the other half of the test set, and the result will be shown on the private leaderboard. The mark for "model performance" is given based on this score (see Section 6 for details).

Note that you don't have to submit your prediction to Kaggle everyday. But **you must submit your final prediction to Kaggle before the deadline of the assignment**. Otherwise you will not have a score on the private leaderboard and lose the "performance" marks (20%). Therefore, **I strongly recommend you submit your prediction regularly to monitor your score**. You may have a big surprise to see the big difference between your training performance and testing performance, if you submit at the last day.

The Kaggle link: `https://www.kaggle.com/t/17cac451c5214c08ba0363b47567884d`. Please contact the unit coordinator if you cannot participate in this competition.

5.2 Blackboard submission

Beside the Kaggle submission, you will also need to make a final submission to BlackBoard. You are required to submit a single zip file that contains all documents, including:

- The source code main.ipynb with your code, comment, cell output, and Kaggle name at the top
- The csv file that contains your prediction for the test set **prediction.csv** (as in the format of sample_solution.csv on Kaggle)
- The signed **declaration form**
- (Optional) The **README** file which contains all other information that is not suitable to put into markdown cells of your jupyter notebook

6 Marking

This assignment has 100 marks in total and it is distributed as follows:

- **Satisfactory submission [10 marks]**: which concerns if you have followed all submission requirements properly, such as,
  - if all required files are included
  - if all files are named and organised properly
- **Data preprocessing [20 marks]**: which concerns the quality of data preprocessing performed, e.g.,
  - if features are not scaled without an explanation
  - if feature are in the inappropriate types without a good reason
  - if advanced preprocessing which brought in performance gain, such as feature engineering
- **Model development [20 marks]**: which concerns if a serious model selection process has been taken, such as,

  - if a wide variety of models are considered
  - if hyper-parameter tuning is well conducted, e.g., hold-out validation, cross validation
  - if the evaluation is fair

- **Model Interpretation [30 marks]**: which concerns if a comprehensive and insightful model interpretation is demonstrated, such as,
  - if all required plots are given
  - if additional interpretation is given
  - More importantly, if knowledgeable insights about the data and model is drawn

- **Prediction [20 marks]**: which concerns the performance of the final model, including
  - the performance (R^2 and MAPE) on the testing set
  - the difference between training performance and testing performance

Note that some factors are not directly marked but they will have significant impact on the marks for all sections above, e.g., **the readability of your notebook** (both code and text). If you fail to make yourself clear and I did not understand it, you lose the marks for that part. Always try your best to maintain high-quality code and report!

This is the end of the assignment specification. Have fun!

References

[1] Jingde Li and Hong Hao. Numerical simulation of medium to large scale bleve and the prediction of bleve's blast wave in obstructed environment. *Process Safety and Environmental Protection*, 145:94–109, 2021.

[2] Jingde Li, Qilin Li, Hong Hao, and Ling Li. Prediction of bleve blast loading using cfd and artificial neural network. *Process Safety and Environmental Protection*, 149:711–723, 2021.