Nuclear Engineering and Technology 55 (2023) 4607–4616

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

journal homepage: www.elsevier.com/locate/net

Original Article

Limiting conditions prediction using machine learning for loss of condenser vacuum event

Dong-Hun Shin [a,1], Moon-Ghu Park [b,*], Hae-Yong Jeong [b], Jae-Yong Lee [b], Jung-Uk Sohn [c], Do-Yeon Kim [d]

[a] Department of Nuclear and Quantum Engineering, Korea Advanced Institute of Science and Technology, 291 Daehak-ro, Yuseong-gu, Daejeon, 305-701, Republic of Korea
[b] Department of Quantum and Nuclear Engineering, Sejong University, 209 Neungdong-ro, Gwangjin-gu, Seoul, 05006, Republic of Korea
[c] ZettaCognition, 310 Yulgok-Gwan, 209 Neungdong-ro, Gwangjin-gu, Seoul, 05006, Republic of Korea
[d] Department of Mechanical Engineering, Pusan National University, 2, Busandaehak-ro 63beon-gil, Geumjeong-gu, Busan, 46241, Republic of Korea

ARTICLE INFO

Keywords:
Machine learning
Safety analysis
MARS-KS
XGBoost
LOCV

ABSTRACT

We implement machine learning regression models to predict peak pressures of primary and secondary systems, a major safety concern in Loss Of Condenser Vacuum (LOCV) accident. We selected the Multi-dimensional Analysis of Reactor Safety-KINS standard (MARS-KS) code to analyze the LOCV accident, and the reference plant is the Korean Optimized Power Reactor 1000MWe (OPR1000). eXtreme Gradient Boosting (XGBoost) is selected as a machine learning tool. The MARS-KS code is used to generate LOCV accident data and the data is applied to train the machine learning model. Hyperparameter optimization is performed using a simulated annealing. The randomly generated combination of initial conditions within the operating range is put into the input of the XGBoost model to predict the peak pressure. These initial conditions that cause peak pressure with MARS-KS generate the results. After such a process, the error between the predicted value and the code output is calculated. Uncertainty about the machine learning model is also calculated to verify the model accuracy. The machine learning model presented in this paper successfully identifies a combination of initial conditions that produce a more conservative peak pressure than the values calculated with existing methodologies.

1. Introduction

The construction and operation of nuclear power plants require demonstrating that the design has sufficient safety margins under various operating conditions and unlikely but possible accident conditions. The safety margin of a nuclear power plant is defined as the difference or the ratio between the value of the actual power plant and the limit value at which the related system or parts faulted when the set-point exceeds a specific value. Therefore, if a power plant operates while maintaining a sufficient safety margin, it is guaranteed to be safe despite its operating condition.

Fig. 1 is a general concept of safety margin suggested by the International Atomic Energy Agency (IAEA) [1]. Generally, the safety margin is set to the Departure from Nucleate Boiling Ratio (DNBR) or the nuclear fuel and cladding temperatures that confirm the barrier integrity against radioactive material leakage. Furthermore, it may be set to the pressure or stress required to maintain the integrity of the reactor coolant system, the temperature and pressure of the containment system, and the amount of radiation affecting the environment. Since we can evaluate the limitation on numerous factors threatening safety in several ways, the safety margin could be understood as the difference between the legally defined acceptance criteria and the actual state value of the nuclear power plant.

Power plant designers should perform safety analysis and measure safety margin using safety analysis codes to ensure that nuclear power plants can operate with sufficient safety margin when an accident occurs. Several safety analysis methodologies can be selectively used to confirm this depending on the type of the computer code, assumption about system availability, assumption of initial and boundary conditions. Table 1 summarizes various safety analysis methodologies specified in the IAEA safety standards. This study uses a combined method using the best-estimate code and the application of conservative initial

* Corresponding author.
E-mail address: mgpark@sejong.ac.kr (M.-G. Park).
[1] First author.

https://doi.org/10.1016/j.net.2023.08.038
Received 30 June 2022; Received in revised form 20 July 2023; Accepted 25 August 2023
Available online 26 August 2023

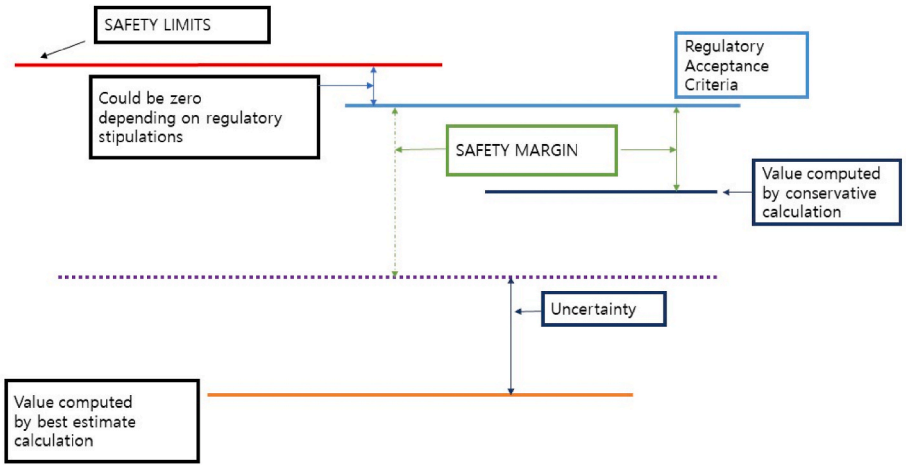

Fig. 1. Concept of safety margins [1].

Table 1
Classification of safety analysis methodology [2].

Option	Computer code	Availability of systems	Initial and boundary conditions
Conservative	Conservative	Conservative	Conservative
Combined	Best estimate	Conservative	Conservative
Best estimate	Best estimate	Conservative	Realistic + Uncertainty
Risk informed	Best estimate	Probabilistic	Realistic + Uncertainty

conditions and boundary conditions. This method is widely used to analyze design basis accidents and anticipated operational occurrences for licensing calculations [1,2].

In such a case, the main concern is conservative accident results and a combination of the initial conditions. To ideally verify the safety margin for an accident using the combined method, various input conditions within the Limiting Conditions for Operation (LCO) range must be configured. After that, the analysis is performed for each input condition to derive conservative results and prove that it is under the legal acceptance criteria. However, this method requires a lot of time and effort because it requires analysis of numerous cases. Therefore, we performed safety analysis by composing initial conditions with a combination of maximum, minimum, and nominal LCO ranges in the Korea Non-LOCA Analysis Package (KNAP) methodology currently applicable to safety analysis in Korea. The validity of this method, based on numerous sensitivity analyses accomplished by experts, has been recognized enough to be used for current licensing [3]. Yet, because this method performs safety analysis only with a limited combination of initial conditions, there may be initial conditions that cause conservative results that have not been discovered. Accordingly, it is necessary to confirm the existence of an initial condition combination that derives conservative results beyond the combination suggested in the existing methodology.

Machine learning [4,5], one of the data mining methodologies, is a research area that mixes statistics, artificial intelligence, and computer science. It performs well if an accurate understanding of the problems to be solved and refined data are involved [6]. eXtreme Gradient Boosting (XGBoost), chosen among the machine learning techniques, is an ensemble algorithm based on Gradient Boosting Decision Tree (GBDT)

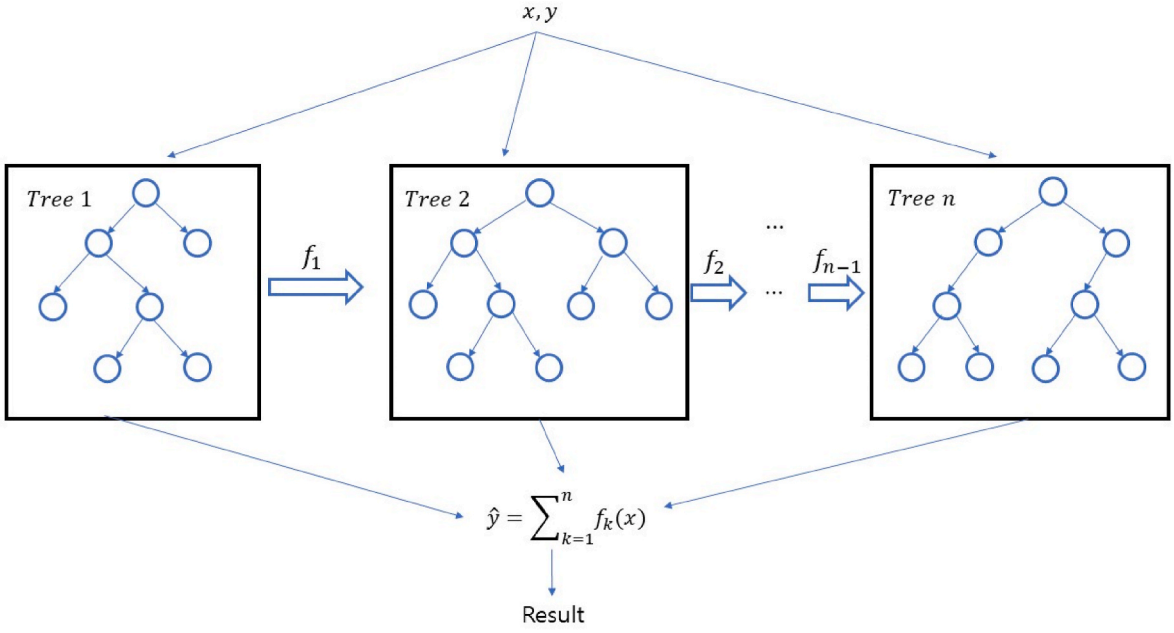

Fig. 2. Basic concept of XGBoost [8].

Table 2
Initial condition range for LOCV.

Initial conditions	LCO	Analysis range in the present method
Reactor coolant flow rate, lbm/s	32,065–39,153	32,869–39,128
Pressurizer pressure, MPa (psia)	13.79–16.03 (2000–2325)	15.18–15.68 (2202–2275)
Pressurizer level, %	21.9–60	37.7–53.4
Steam generator level, % WR	35–98.2	45.3–89.1

[7]. Fig. 2 shows the principle of XGBoost. In Fig. 2, x and y denote independent and target variables, respectively, and f_k denotes the result of the kth tree. XGBoost is an interactive decision tree algorithm. All trees take the results from the previous tree and modify the weights to reduce the residual [8]. XGBoost is one of the most frequently used algorithms, showing good training and high prediction performance [9]. This technology can solve the existing time-consuming problem, such as parallel processing, and has good scalability as an open-source technology. In addition, it supports regularization to prevent data overfitting and can use the desired objective function [10]. Because of these advantages, previous studies using XGBoost have also been conducted in the nuclear field [11–13]. However, no previous study on the XGBoost model predicts conservative accident analysis values for safety margin verification.

We implement an XGBoost-based regression model that predicts conservative results for nuclear power plant accident analyses through various combinations of initial conditions. Additionally, the model-predicted value undergoes validation through a safety analysis code to ensure its accuracy. Furthermore, we also conducted the model uncertainty analysis.

2. Dataset

We analyzed the Loss Of Condenser Vacuum (LOCV) accident dataset with the MARS-KS code [14] for OPR1000. MARS-KS is the safety analysis code used for data generation that is actively used for regulation in Korea. OPR1000 is the reference power plant. LOCV is caused by the failure of the circulating water system that supplies coolant or by the failure of the main condenser evacuation system. Heat removal from the primary system to the secondary system is rapidly reduced by causing a rapid interruption of the steam flow to the turbine and feed water flow to the steam generator. Therefore, the peak pressures in the primary and secondary systems are significant safety concerns [15,16].

When analyzing LOCV, selecting the appropriate initial condition variables and sampling range of the accident is necessary. The variables and ranges of the initial conditions necessary for analyzing LOCV are selected with KNAP. These variables are selected because they influence LOCV most from the results of many sensitivity analyses, and we used these variables as they are. KNAP methodology generates initial conditions by combining maximum, minimum, and nominal values within the LCO range. On the other hand, this study generates accident initial condition data based on the indirect sampling method presented in Ref. [17].

Control variables that dominantly affect the input variables constituting the initial conditions are selected for indirect sampling. After the selection, we adjusted the control variable within the range that does not affect power plant behavior abnormally to generate a combination of various initial conditions within the operating range. This method can only generate data in a narrower range than the existing LCO range. For this reason, it is difficult to apply this method to the current licensing calculation, and it is necessary to study to produce data up to the LCO range by selecting additional control variables and performing sensitivity analysis in the future. Because this methodology is suitable for generating a significant amount of data by automatically creating a steady state for various initial conditions, it has the advantage of easily obtaining the numerous data required for machine learning. In other words, creating one input file valid for a steady state and changing the control variables only to generate another data set requires less effort than the existing methodology that creates various input files. We generated 60,000 data using the method suggested in Ref. [17]. Table 2

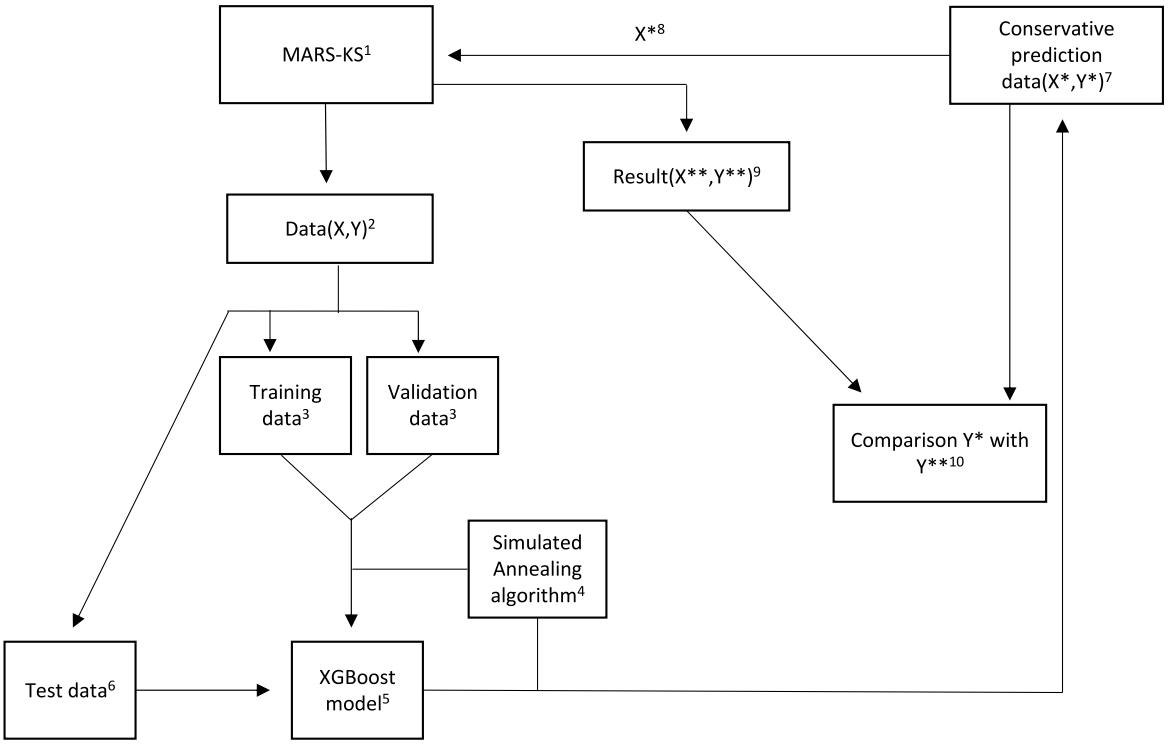

Fig. 3. Schematic of entire research process.

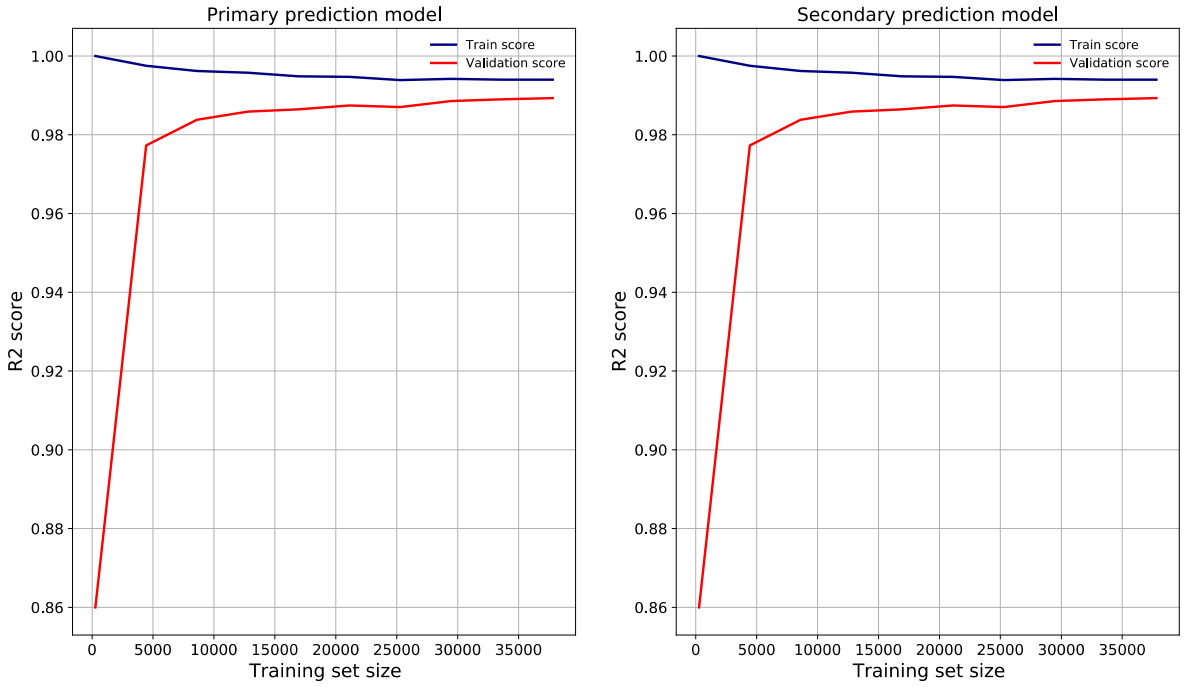

Fig. 4. Learning curve of each model.

compares the LCO and analysis ranges used in this paper. The initial conditions we select are the same as those selected by KNAP, and it has been shown in Ref. [17] that these variables are independent. The specification of the reference plant implemented to generate these data is as used in Ref. [17].

3. Methodology

Fig. 3 presents a schematic that illustrates the overall contents of the research. The symbol X represents the initial condition, while Y denotes the peak pressure. We use asterisks to distinguish between data with different identities. For instance, although the symbols are indistinguishable, Y and Y* have different identities. The superscript number in the box indicates the order of progress. We divide the data to learn the model, which is then trained, and the hyperparameters are calibrated to obtain optimal results. We created two models to predict the primary and secondary peak pressures. We use the XGBoost model to generate conservative peak pressure prediction values. In section 4, we calculate the peak pressures using the safety analysis code and compare the calculated value with the value predicted by the model using the initial conditions of the conservative value. Additionally, although it is not shown in the schematic, we perform an uncertainty analysis of the results. A detailed explanation of this process follows.

In general, the more training data, the better it is to improve the model's prediction accuracy [18]. Sufficient data composed of various combinations should be prepared to understand the inherent correlation in the data. It is effective in solving problems such as overfitting [19]. XGBoost is an algorithm that uses decision trees in an interactive manner, where the output of each tree is used to adjust the weights in the direction of minimizing the residual error. This behavior easily leads to overfitting, where the model tries to fit the data too closely. If the model is overfitting, the accuracy of the model drops as new data comes in because it is trained too accurately on the training data. Therefore, when using this model, it is important to prepare many training data or tune the hyperparameters appropriately to avoid overfitting.

Because it often takes significant time and effort to produce data, producing sufficient data to solve a given problem and using it only for training is expected. The adequacy of the data can be determined by observing the learning curve, regardless of whether or not enough data has been obtained [20]. The learning curve is a graph showing how the performance of the model for training data and validation data changes according to the amount of training data. Therefore, before calibrating hyperparameters, we divide 60,000 data into training, validation, and test data in a 6:2:2 ratio to generate the learning curves for the primary and secondary regression models. The x-axis of the learning curve shown in Fig. 4 is the size of the training dataset, and the y-axis is the coefficient of determination (R2 score). The evaluation score is mainly used as a coefficient of determination because it is possible to see the suitability of the regression model for data intuitively. In both primary and secondary regression models, as the amount of training data increases, the training score decreases and the validation score increases; both scores show convergence to a value near 0.99. Although there is no absolute standard for this value, generally in engineering, it is considered meaningful if the coefficient of determination is more than 0.7 [21].

Both scores converge to a high value of around 1 compared to 0.7 because we produce data with the safety analysis code and the high correlation among features inherent in the data. In addition, XGBoost is suitable for this data because it is vulnerable to noise due to the nature of the model that uses boosting algorithm to learn while reducing residuals. Still, there is little noise in safety analysis data [22]. Per Fig. 4, when the data size is over 30,000, the train and validation scores are saturated due to the bias-variance tradeoff issue of the model. Therefore, 42,000 training data, which is 60% of the 60,000 data we generated, is sufficient for training and optimization of the model. The remaining 40% of the data is divided evenly for validation and testing.

We further optimize hyperparameters of the XGBoost model in the superscript step 4 at Fig. 3 as follows. Hyperparameters are options that the user must manually set to determine the parameters of the model for improving performance. Optimizing hyperparameters can prevent the overfitting of the model in the learning process, maximizing the performance of the model. Therefore, in implementing the model, hyperparameter optimization is a process that must be proceeded [23]. There are various methods for hyperparameter optimization, and Simulated Annealing (SA) [24] is one of them. SA is a method that simulates the quenching phenomenon, a physical process until the metal is sufficiently heated and then crystallized in a complete lattice state. This method

Algorithm 1 Algorithm

```
procedure MyProcedure
    x ← <Initial random solution>
    i ← 0
    while <Global stop condition not satisfied> do
        t ← CoolingSchedule(i)
        i ← i + 1
        while <Local stop condition not satisfied> do
            x* ← PickRandomNeighbor
            F ← XGBoostModel
            ΔE = F(x*) − F(x)
            if ΔE < 0 then
                x ← x*
            else
                r ← <Random uniform number between 0 and 1>
                if r < e^{−ΔE/(k*t)} then
                    x ← x*
```

Fig. 5. SA pseudo code applied to the XGBoost.

Table 3
Search range for hyperparameter and optimization result.

Hyperparameters	Search range	Optimization results	
		Primary model	Secondary model
n_estimators	1000–3000	2000	2000
learning_rate	0.05–0.3	0.069	0.055
colsample_bytree	0.6–1.0	0.8333	0.9925
colsample_bylevel	0.6–1.0	0.9213	0.9210
colsample_bynode	0.6–1.0	0.9949	0.8264
max_depth	2–12	6	11
subsample	0.5–1.0	0.8777	0.9429
gamma	0 .0–15.0	0.1712	1.2756
reg_alpha	0.0–100.0	9.0	14.0
reg_lambda	0.0–100.0	1.5	7.5

allows escaping the local minima by allowing the algorithm to choose what is not the best in its state when it is likely to converge to the local minimum rather than to find the global minimum. Therefore, we use SA not to make cost function converge to the local minimum but to find the global minimum during optimization. Equation (1) describes the cost function. y and $\hat{y}$ represent the validation data and predicted data, respectively. The cost function is the sum of the mean absolute error (MAE) and variance of the error (Var) so that it is robust to outliers. The bias and variance of the model are evenly lowered. Fig. 5 is a pseudo-code that shows the process of optimizing the hyperparameters of the XGBoost model using SA.

$$Cost = MAE\ (\varepsilon) + Var\ (\varepsilon), \tag{1}$$

Where $\varepsilon = |y_i - \hat{y}_i|$.

Table 3 summarizes the range of hyperparameters and optimized results by Simulated Annealing for each model predicting peak pressure on the primary and secondary models. We calibrated a total of 10 hyperparameters, whereby we select n_estimators and max_depth as integers within a specified range, and the remaining hyperparameters are selected as real values. A detailed description of the hyperparameters is given in Ref. [25].

In the optimization results, the other hyperparameter values exhibit similarity. In contrast, the gamma and max_depth parameters, which influence the convergence of the machine learning model, are noticeably more prominent in the secondary model. Additionally, the learning rate employed in the machine learning model is substantially lower (0.069 or 0.055) than the default value of 0.3, which can lead to the problem of underfitting. SA's optimization algorithm intricately determines these parameter values, and the inherent opacity of machine learning, often referred to as a 'black box,' makes precise explanation challenging. However, as mentioned earlier, the XGBoost model we utilized tends to be susceptible to overfitting, suggesting that the model may have used a low learning rate.

Fig. 6 shows the loss according to the optimized XGBoost model's training times. The x-axis is the n_estimators, the number of rounds for boosting, and the y-axis is the loss defined as a root mean square error (RMSE). Boosting is repeated 2000 times for each model, and the figure shows only up to 120 times with a significant change in the loss. As the step progresses, the loss gradually decreases. The red line represents the final loss value after learning, with a value of 0.5209 for the primary regression model and 0.9196 for the secondary regression model. Before hyperparameter calibration, we have shown sufficient training data through the learning curve shown in Fig. 4, and we use a large number of iterations during training. The models learn to reduce the loss well, as

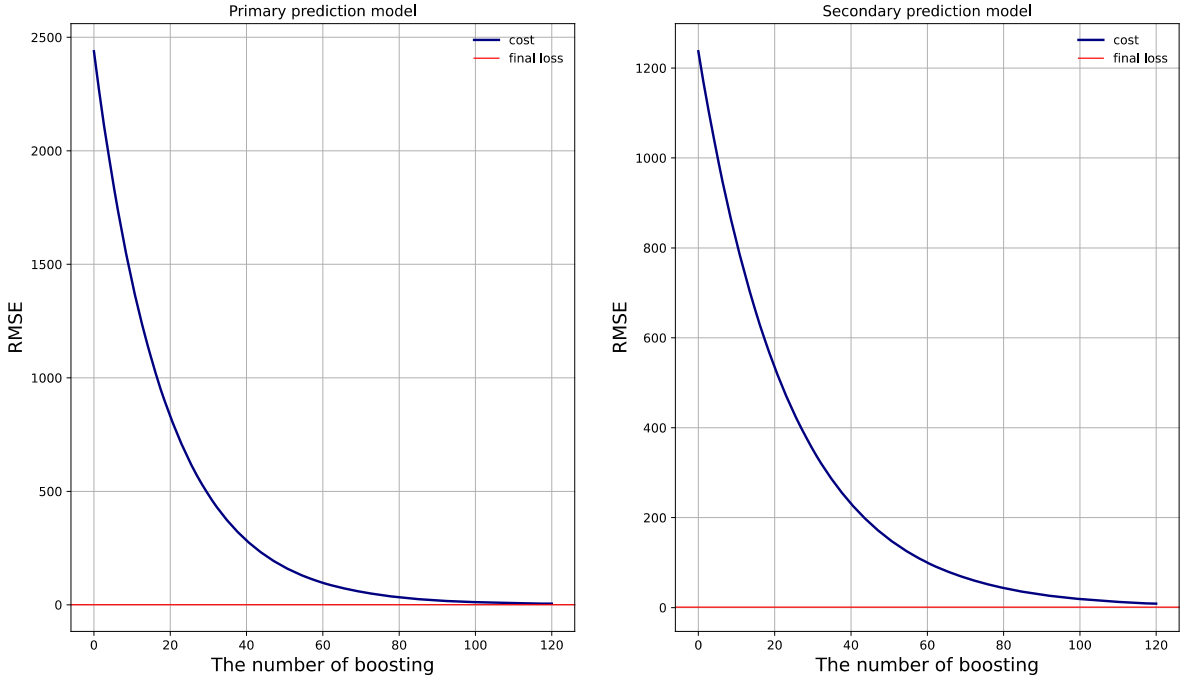

Fig. 6. Loss of the model according to the number of boosting.

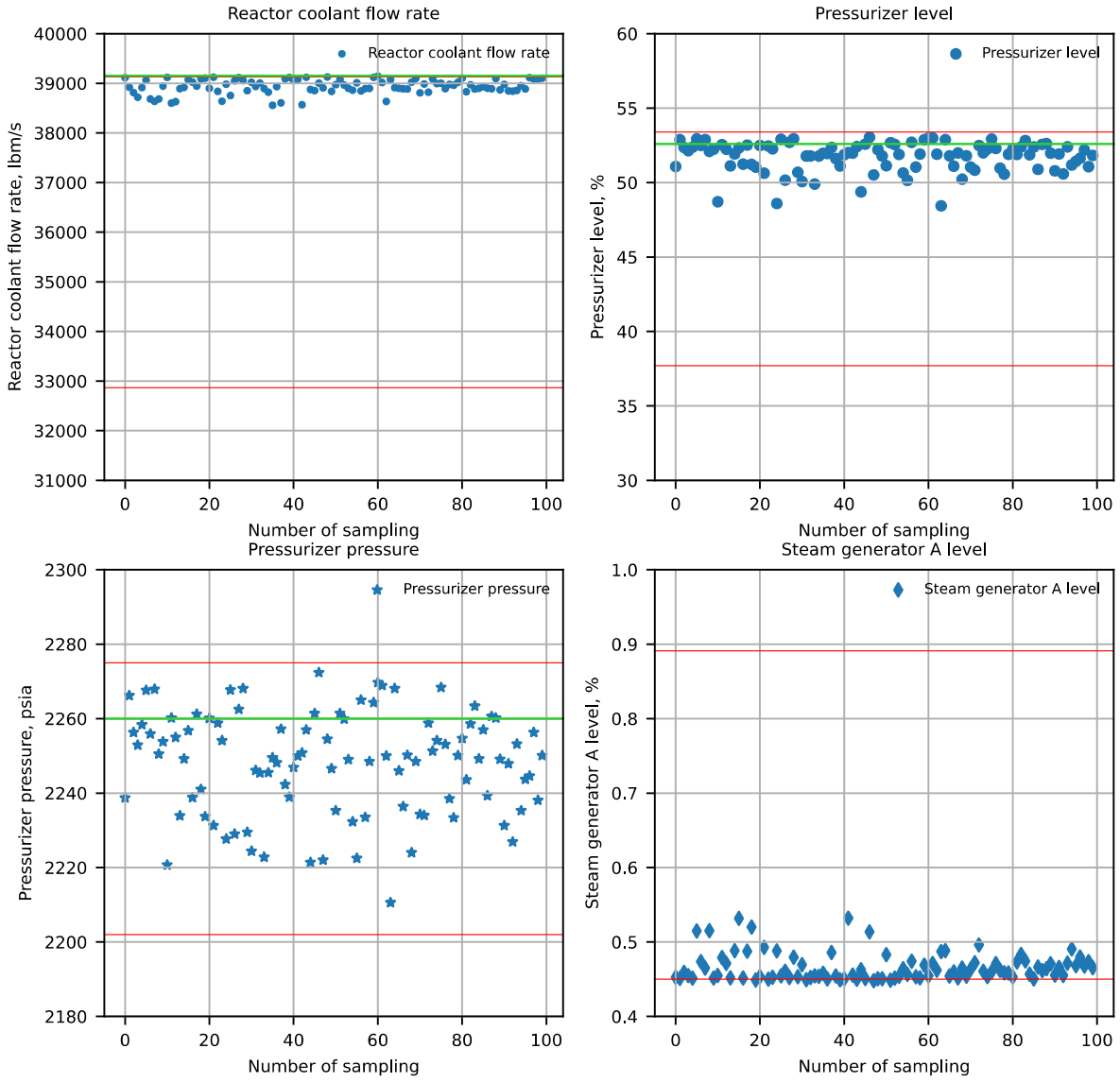

Fig. 7. Conservative initial condition distribution predicted by the primary model.

shown in Fig. 6. These figures illustrate that underfitting does not occur due to the low learning rate we are concerned about.

As mentioned above, the peak pressures of the primary and secondary systems are the major safety concern in LOCV. The power plant designers must prove that the peak pressure that may occur in the LOCV is within limit specified in the final safety analysis report. Therefore, it is important to find a conservative peak pressure through a combination of various initial conditions, and this value is obtained using the XGBoost regression model implemented above. First, a random input variable combination is created within the LCO range presented in Table 2. We sample each input variable using uniform, normal, and log-uniform distributions [17]. Then, the generated input variables are put into the regression model to obtain predicted values. The purpose of finding the conservative peak pressure is met through an iterative process. Since this process is a combination optimization problem, SA is used. Through many iterative processes, we select 100 initial conditions that cause conservative peak pressure for each prediction model, the primary and the secondary.

4. Results and discussion

Figs. 7 and 8 are 100 initial condition cases predicted by the primary and the secondary of XGBoost model, respectively. We compare these results with the conservative initial conditions presented by the KNAP methodology. The red line represents the range of data used for model learning, and the green line is the initial condition value suggested by KNAP. The data distribution of the reactor coolant flow rate and pressurizer level in Fig. 7 is concentrated on the red line at the top and contains green lines. Pressurizer pressure data is relatively widely distributed within the range of the red line. Compared to the green line, peak pressure may occur at various pressurizer pressures than the value suggested by KNAP. There is no green line in the steam generator water level data because the steam generator level presented by KNAP is 35%. The minimum value of the LCO range and the minimum range of the data we used for model learning is 45.3%. Nevertheless, the fact that the data distribution is concentrated on the red line at the bottom shows that the model predicts the low steam generator level in the range of learned data as a conservative initial condition value, indicating that it is consistent with the trend suggested by KNAP. Fig. 8 shows a similar pattern except for pressurizer pressure. In this case, the data is distributed around the red line at the bottom in the case of the pressurizer pressure. This phenomenon can be interpreted as follows: when the initial pressure is low, the reactor trip occurs later due to high pressure. This increases the amount of heat transferred from the primary to the

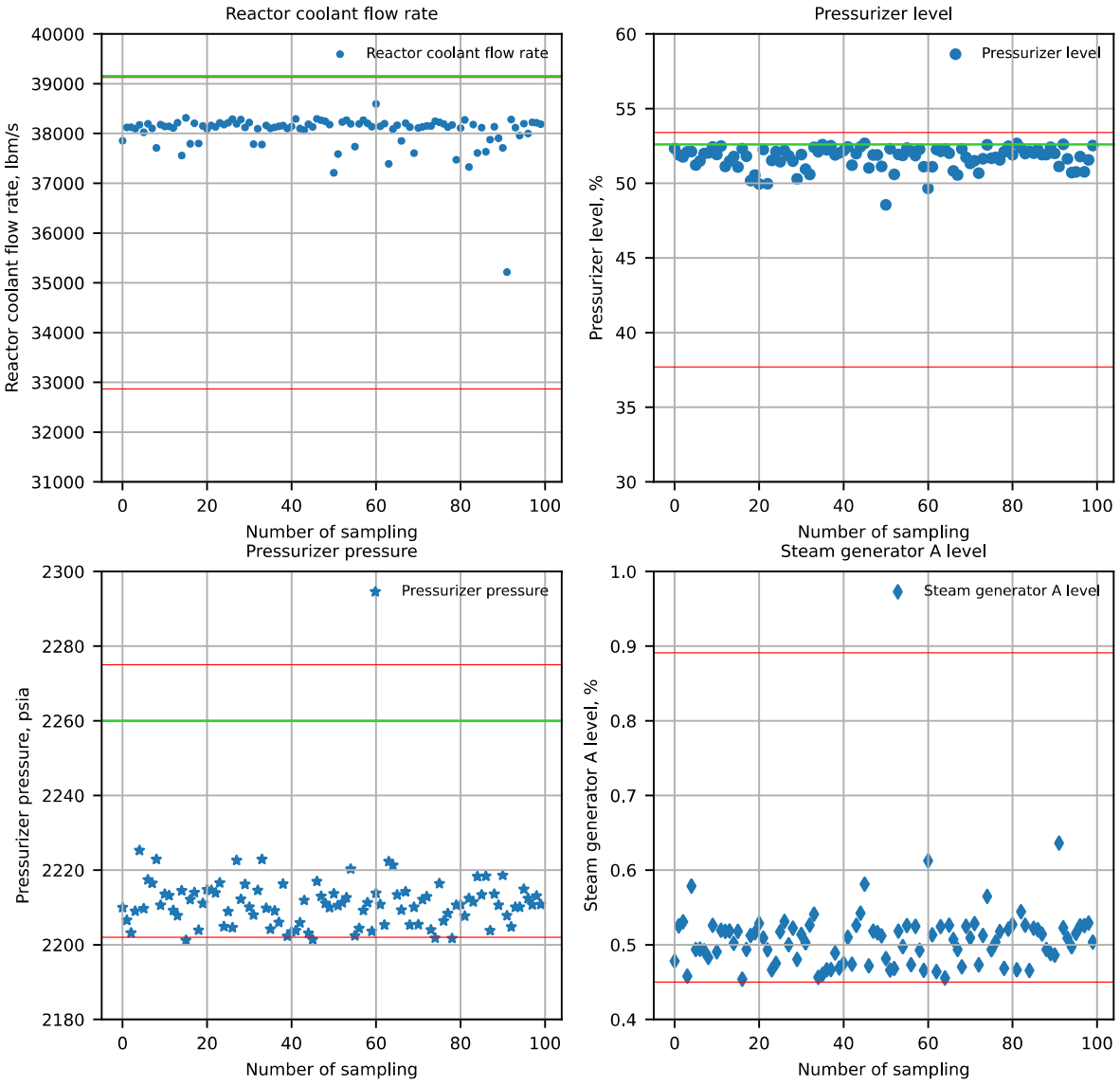

Fig. 8. Conservative initial condition distribution predicted by the secondary model.

Table 4
Comparison of conservative input conditions for LOCV accidents.

Initial input parameters	KNAP	MARS-KS	
		Primary peak pressure	Secondary peak pressure
Reactor coolant flow, lbm/s	39,153	39,103	38,111
Pressurizer pressure, psia	2260	2233	2205
Pressurizer level, %	52.6	51	51.3
Steam generator level, %	35	44	50

secondary, resulting in an increase in the peak pressure on the secondary system. However, if the main steam safety valves open before the high pressurizer pressure trip, high initial pressurizer pressure could make the maximum reactor coolant system over-pressurization. Because of this complicated effect, the KNAP methodology takes a steady state pressurizer near the nominal value as an initial condition, which makes differences from our initial conditions. Table 4 compares the conservative initial condition values presented by KNAP with the initial condition results presented in this study.

We predict 100 conservative peak pressures and the initial conditions causing them with the XGBoost model and compare them to the results of KNAP. However, since this value is simply the predicted value of the machine learning model, it is necessary to verify it through comparison with the value calculated in the safety analysis code. Fig. 9 compares the primary peak pressure predicted by the primary XGBoost model with the result calculated by MARS-KS code. The root mean square error between the predicted and code values is 0.7231 psi, and the largest difference between the two values is 1.6 psi, which appeared in the 33rd case. The highest peak pressure value is 2652.5 psia in the 20th case, and the error is 0.8476 psi. Considering that the y-axis scale is near 2650 psia, the error size can indicate that the primary regression model is successfully predicted. Fig. 10 compares the secondary peak pressure predicted by the secondary XGBoost model with the result calculated by MARS-KS code. The root mean square error is 4.5612 psi, showing the maximum difference of 8.67 psi between the two values in the 56th case. The highest peak pressure value is 1358.7 psia in the 29th case, and the error, in this case, is 8.61 psi. The secondary regression model has a higher error than the primary regression model. This shows that the data used in learning is more suitable for describing the primary system than the secondary system. To improve the accuracy of the secondary model in the future, data related to the secondary system,

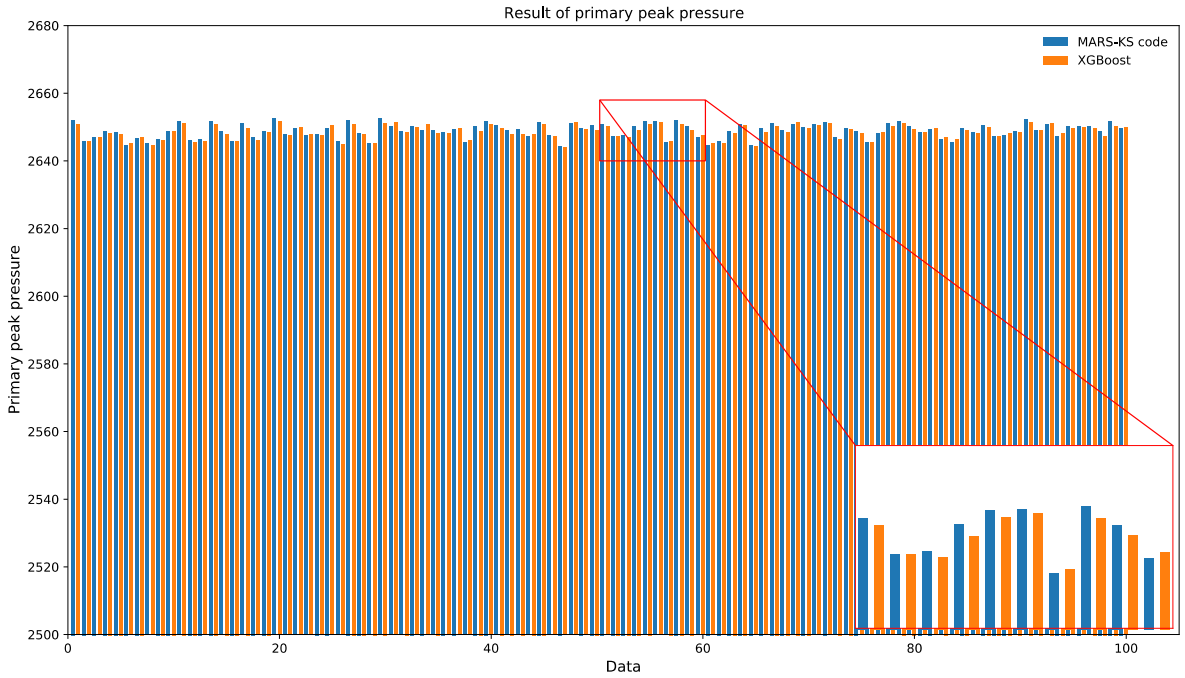

Fig. 9. Results of primary peak pressure.

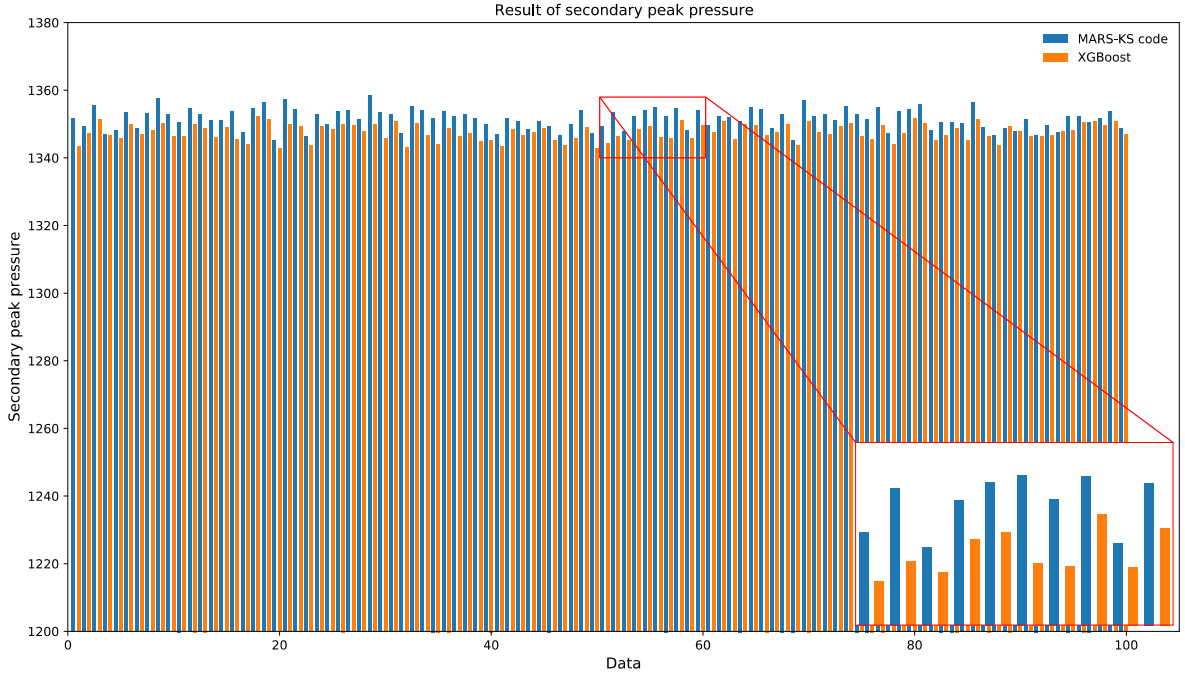

Fig. 10. Results of secondary peak pressure.

such as feed flow rate, could be added.

We used Simulated Annealing for hyperparameter calibration. In addition to SA, various methods could be used for hyperparameter calibration, each with varying results. Therefore, we use Particle Swarm Optimization (PSO) technique and SA to calibrate the hyperparameters of XGBoost. PSO is an optimization algorithm created by imitating swarm objects and finding the optimal solution through self-learning [26]. We use a batch of particles called swarm, and these particles traverse the search space to find the global minimum. Because the algorithm is simple and easy to implement, it is widely used for function optimization and signal processing. Table 5 compares the performance of XGBoost model optimized by SA and PSO techniques. The primary and secondary model's root mean squared errors show slightly better in SA than PSO, but the differences are very small. The highest peak

Table 5
Comparison of performance of SA and PSO optimization techniques.

System	Primary		Secondary	
Optimization technique	SA	PSO	SA	PSO
Root mean squared error (psi)	0.7231	0.7261	4.5612	5.4696
Highest peak pressure (psia)	2652.5	2652.4	1358.7	1357.8

Table 6
Peak pressures in LOCV accident determined by different methodologies.

Code	Methodology	Peak pressure in primary system (psia)	Peak pressure in secondary system (psia)
MARS-KS	Combined approach + Machine learning[a]	2652.5	1358.7
MARS-KS	Combined approach[b]	2647	1343
RETRAN	KNAP	2625	1322
CESEC-III	ABB-CE	2667	1343
Acceptance criteria	–	2750	1397

[a] Methodology presented in this study.
[b] Methodology presented in [16].

pressure predicted by the model optimized with each technology shows a similar level of value. Therefore, there is no significant change of result according to the hyperparameter optimization method.

Table 6 compares the primary and secondary peak pressures found through the method presented in this paper with the values derived from existing methodologies. The existing methodologies are the KNAP methodology based on Reactor TRANsient (RETRAN) code [27] and ABB Combustion Engineering (ABB-CE) methodology based on the CESEC-III code [3,28]. Although each methodology has a slight difference in assumptions and the uncertainty implied by the code, it can be a proper indicator of the result presented in our study compared to methods used in the licensing process. And it is recognized that LOCV can be adequately simulated. The highest primary peak pressure we obtained through our study is 2652.5 psia, located between the values calculated by the RETRAN code and the CESEC-III code. The secondary peak pressure presented in the study is 1358.7 psia, higher than the values suggested by the two codes. We also added the calculation results performed in Ref. [17] by applying the combined method with MARS-KS code. Since the detailed assumptions and codes are the same between reference [17] methodology and the suggested one in this paper, the calculation results can be directly compared. The primary and secondary peak pressures presented by us are higher than the previously calculated results of 2647 psia and 1343 psia, respectively. Because the values found by the machine learning model is closer to the acceptance criteria than the existing values, the safety margin may appear to have narrowed. However, this has been the case due to the application of the machine learning methodology, but rather the discovery of previously undiscovered peak pressure. This methodology can avoid the cliff-edge effect and efficiently validate the safety margin by performing sensitivity analyses with more combinations of initial conditions. These values show that it is possible to successfully find the more conservative peak pressure and the combination of initial conditions that induce it using machine learning techniques.

We quantify the uncertainty of the machine learning model using tolerance intervals [29]. As the number of data used to obtain the tolerance intervals increases, the tolerance factor decreases, and this phenomenon reduces the uncertainty by narrowing the range of the tolerance intervals. Ideally, to obtain the uncertainty of the model, we produce many additional data by increasing the total amount of data. However, producing a large amount of additional data requires many resources, such as time, and obtaining additional data could be limited in some cases. In such cases, we can use Bartlett's Test to obtain uncertainty by using more data than the original data. Bartlett's Test is a technique to validate whether different samples have the same variance or not. It is known that the performance is good when a sample follows a normal distribution and even non-normal data can be used when the number of samples is large [30]. Passing Bartlett's Test means the relationship between the datasets is homogeneous, and thus data pooling is possible. First, we create several sub-datasets by dividing the original data differently into train and test datasets. After performing a Bartlett's Test on each train and test dataset, the data is pooled if the test is passed. Otherwise, a sub-dataset is created again to perform additional Bartlett's Test.

The data pooling process is performed 100 times, and the original data size is 60,000, so the total data size after the data pooling process is 6 million. This dataset is used to obtain a 95%/95% upper tolerance limit [31]. The values predicted by the model are the peak pressures on the primary and secondary systems, so the upper tolerance limit could be considered for additional conservatism in the nuclear energy field. In calculating the tolerance limit, we use a one-sided tolerance limit because the minus error direction is not the conservative way. The distribution used in this case is noncentral t-distribution, mainly used in the one-sided tolerance limit calculation [32]. Fig. 11 shows the uncertainties according to the amount of data for the primary and secondary systems. The x-axis implies the number of times the data is pooled. The y-axis is the upper tolerance limit. As we expect, the upper tolerance limit tends to decrease as the number of data increases in the figure. However, as data increases, the decline rate of uncertainty decreases, and uncertainty is saturated to a specific value. For the primary system, the 1st value is +1.1315, and the 100th value is +1.1156, just a 1.4% decrease. For the secondary system, the 1st value is +1.5604, and the 100th value is +1.5247, a 2.3% decrease.

This shows that although the amount of data affects the uncertainty, the effect is insignificant. Therefore, if the amount of data is guaranteed to some extent, uncertainty can be reasonably obtained within a specific range. Finally, we can say that the uncertainties of our machine learning models are +1.1315 and + 1.5604, respectively when interpreted from a

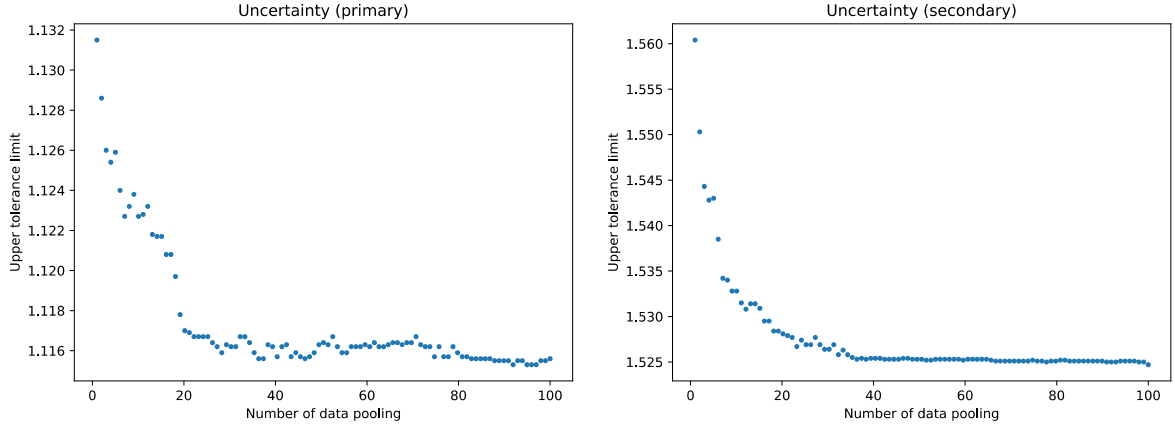

Fig. 11. Uncertainty of machine learning model.

conservative perspective.

5. Conclusion

In LOCV, the peak pressures of the primary and secondary systems are the main concern. Whether the plant can maintain a safe state during an accident is analyzed through safety analysis. In such cases, only the limited range is analyzed because the initial conditions are generated by a combination of minimum, maximum, and nominal within the LCO range. To overcome this limitation, in this study, we create a machine-learning model that predicts the peak pressure and the initial conditions that induce it. We obtain data using MARS-KS code, train the machine learning model with this data, and perform hyperparameter tuning for optimization. Several simulations are performed with the machine learning model, and the peak pressures on the primary and secondary systems and the initial conditions that induce them are successfully found. As a result of comparison with the calculated values by several safety analysis codes performed previously for the LOCV, the machine learning model predicts a more conservative peak pressure within a small error range. In addition, the uncertainty inherent in the machine learning model is also calculated and quantified. Also, the uncertainty change as the amount of data increases is generated.

The model can perform a safety analysis by combining the initial conditions within the LCO range. This can find new combinations of initial conditions that generate peak pressures that existing methodologies have not discovered. This method can be used to evaluate the safety margin of a pressurized water reactor to confirm the validity of the conservative evaluation methodology presented by the plant designer as a regulative perspective. In addition to the LOCV accident, this methodology can be applied to various design basis accidents.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the KETEP funded by the Korea government Ministry of Trade, Industry and Energy (20206510100040 and 2021040101002D).

References

[1] I.A.E. Agency, Safety Margins of Operating Reactors-Analysis of Uncertainties and Implications for Decision Making, International Atomic Energy Agency, 2003.
[2] A. International Atomic Energy, Deterministic Safety Analysis for Nuclear Power Plants : Specific Safety Guide, IAEA, Vienna, 2019.
[3] KHNP, Korea Non-LOCA Analysis Package, KHNP, TR-KHNP-0009, 2007 (in Korean).
[4] M.G. Genton, Classes of kernels for machine learning: a statistics perspective, J. Mach. Learn. Res. 2 (2) (2002) 299–312.
[5] H. Mannila, Data mining: machine learning, statistics, and databases, in: Eighth International Conference on Scientific and Statistical Database Systems, Proceedings, 1996, pp. 2–9.
[6] V. Gudivada, et al., Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations, Int. J. Adv. Software 10 (1) (2017) 1–20.
[7] X. Shi, et al., An accident prediction approach based on XGBoost, in: 2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), IEEE, 2017.
[8] Y. Wang, et al., A hybrid ensemble method for pulsar candidate classification, Astrophys. Space Sci. 364 (8) (2019).
[9] T. Chen, C. Guestrin, XGboost: a scalable tree boosting system, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016.
[10] S.S. Dhaliwal, et al., Effective intrusion detection system using XGBoost, Information 9 (7) (2018) 149.
[11] H.T. Bang, et al., Application of machine learning methods to predict a thermal conductivity model for compacted bentonite, Ann. Nucl. Energy 142 (2020), 107395.
[12] J. Cai, et al., An assembly-level neutronic calculation method based on LightGBM algorithm, Ann. Nucl. Energy 150 (2021), 107871.
[13] C. Xu, et al., A study of predicting irradiation-induced transition temperature shift for RPV steels with XGBoost modeling, Nucl. Eng. Technol. 53 (8) (2021) 2610–2615.
[14] B.D. Chung, et al., MARS CODE MANAUAL VOLUME IV-Developmental Assessment Report, Korea Atomic Energy Research Institute, 2010.
[15] NUREG-0800, Standard review plan for the review of safety analysis reports for nuclear power plants, Rev. (2007).
[16] (Chapter 15), June, APR1400 Standard Safety Analysis Report, 2002.
[17] D.H. Shin, et al., Application of a Combined Safety Approach for the Evaluation of Safety Margin during a Loss of Condenser Vacuum Event, Nuclear Engineering and Technology, 2021.
[18] R. Cuocolo, et al., Current applications of big data and machine learning in cardiology, J. Geriatr. Cardiol.: JGC 16 (8) (2019) 601.
[19] X. Ying, An overview of overfitting and its solutions, in: Journal of Physics: Conference Series, IOP Publishing, 2019.
[20] A.N. Richter, T.M. Khoshgoftaar, Learning curve estimation with large imbalanced datasets, in: 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE, 2019.
[21] C. D, J. R, Statistics without Maths for Psychology, Prentice Hall, 2011, p. 175, 5th edition.
[22] A. Gómez-Ríos, et al., A study on the noise label influence in boosting algorithms: AdaBoos, GBM and XGBoost, in: International Conference on Hybrid Artificial Intelligence Systems, Springer, 2017.
[23] S. Sun, et al., A survey of optimization methods from a machine learning perspective, IEEE Trans. Cybern. 50 (8) (2019) 3668–3681.
[24] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science 220 (4598) (1983) 671–680.
[25] dmlc XGBoost stable", https://xgboost.readthedocs.io/en/stable/parameter.html, accessed 28 June 2022.
[26] R. Poli, et al., Particle swarm optimization, Swarm Intell. 1 (1) (2007) 33–57.
[27] M. Paulsen, et al., RETRAN-3D–A Program for Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems, 1996. NP-7450 1(4).
[28] Combustion Engineering Inc, CESEC Digital Simulation of a Combustion Engineering Nuclear Steam Supply System, 1981. LD-82-001.
[29] C. Liao, H. Iyer, A tolerance interval for the normal distribution with several variance components, Stat. Sin. (2004) 217–229.
[30] H. Arsham, M. Lovric, Bartlett's test, Int. encycl. Stat. Sci. 1 (2011) 87–88.
[31] M.G. Vangel, One-sided nonparametric tolerance limits, Commun. Stat. Simulat. Comput. 23 (4) (1994) 1137–1154.
[32] D.S. Young, Tolerance: an R package for estimating tolerance intervals, J. Stat. Software 36 (2010) 1–39.