Nuclear Engineering and Technology 53 (2021) 3563–3579

Contents lists available at ScienceDirect

Nuclear Engineering and Technology

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

Original Article

Machine learning of LWR spent nuclear fuel assembly decay heat measurements

Bamidele Ebiwonjumi, Alexey Cherezov, Siarhei Dzianisau, Deokjung Lee*

Department of Nuclear Engineering, Ulsan National Institute of Science and Technology, 50 UNIST-gil, Eonyang-eup, Ulju-gun, Ulsan, 44919, Republic of Korea

ARTICLE INFO

Article history:
Received 29 September 2020
Received in revised form
10 May 2021
Accepted 28 May 2021
Available online 8 June 2021

Keywords:
Decay heat
Spent nuclear fuel
Machine learning
Light water reactor
Synthetic data
Uncertainty analysis

ABSTRACT

Measured decay heat data of light water reactor (LWR) spent nuclear fuel (SNF) assemblies are adopted to train machine learning (ML) models. The measured data is available for fuel assemblies irradiated in commercial reactors operated in the United States and Sweden. The data comes from calorimetric measurements of discharged pressurized water reactor (PWR) and boiling water reactor (BWR) fuel assemblies. 91 and 171 measurements of PWR and BWR assembly decay heat data are used, respectively. Due to the small size of the measurement dataset, we propose: (*i*) to use the method of multiple runs (*ii*) to generate and use synthetic data, as large dataset which has similar statistical characteristics as the original dataset. Three ML models are developed based on Gaussian process (GP), support vector machines (SVM) and neural networks (NN), with four inputs including the fuel assembly averaged enrichment, assembly averaged burnup, initial heavy metal mass, and cooling time after discharge. The outcomes of this work are (*i*) development of ML models which predict LWR fuel assembly decay heat from the four inputs (*ii*) generation and application of synthetic data which improves the performance of the ML models (*iii*) uncertainty analysis of the ML models and their predictions.
© 2021 Korean Nuclear Society, Published by Elsevier Korea LLC. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

During nuclear reactor operation, fission products and actinides continue to build up from nuclear fission, radioactive decay, and neutron capture. The particles emitted during the radioactive decay processes *i.e.*, alpha, beta, gamma particles, and other radiative particles, lose their kinetic energy when they interact with atoms of materials in the irradiated fuel, and the energy lost is converted into heat. The decay processes continue long after the spent fuel is discharged from the reactor. The energy released from the decay of radionuclides in spent nuclear fuel (SNF) needs to be characterized for safety applications such as storage, transportation, and final disposal, during the back-end of the fuel cycle. Specifically, the decay heat needs to be characterized to ensure that temperature requirements are fulfilled during dry storage/transportation in casks and disposal in geological repositories. Moreover, decay heat needs to be determined so that spent fuel pools at reactor sites can be cooled sufficiently. SNF pools around the world are getting closer to saturation and it is difficult to measure decay heat of all discharged fuels from nuclear reactors. Besides, the calorimetric methods of measuring decay heat, which is the most accurate amongst other methods requires several hours to measure the decay heat of one assembly. Thus, the determination of decay heat depends largely on predictions of computer codes. Computation time is large when decay heat for large number of assemblies are required to be calculated such as in a spent fuel pool application. A much faster alternative to obtain the decay heat estimates is desirable. Machine learning (ML) models can be developed to serve this purpose. ML is the process of training computer algorithms to learn relationships present between the inputs and outputs of a given dataset without explicitly programming them to do so. As a subset of artificial intelligence, ML has become very attractive in many areas of data analysis, pattern recognition [1], speech recognition, language processing, handwriting/face, and character recognition [2], self-driving cars, stock market analysis [3] and fraud detection [4]. ML models could also be referred to as surrogate models, meta models, reduced order models (ROM), emulators or response surfaces. These are approximations with very high computational efficiency and are commonly used in place of computationally expensive code calculations when large number of

* Corresponding author.
E-mail addresses: ebiwonjumi@unist.ac.kr (B. Ebiwonjumi), alcher@unist.ac.kr (A. Cherezov), siarheid@unist.ac.kr (S. Dzianisau), deokjung@unist.ac.kr (D. Lee).

https://doi.org/10.1016/j.net.2021.05.037
1738-5733/© 2021 Korean Nuclear Society, Published by Elsevier Korea LLC. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

repeated runs are needed such as during sensitivity analysis, uncertainty quantification, and optimization studies. The most common use of the computationally expensive code is to generate data of solutions at different input conditions for the problem of interest. Then the data is used to develop the learning model in a non-intrusive manner, requiring no modification to the code.

Several research have appeared in literature in which ML models are trained using experimental data and the models developed are shown to work in practice. Various applications can be found in nuclear fusion and plasma physics [5], high energy physics [6], biological sciences, energy sources application, experimental physics, and chemical reaction processes [7–11]. The advantages of ML of experimental data are yet to be exploited in reactor physics (RP) modelling and simulations (M&S) where there are numerous databases of experimental data which can be used to train ML models. ML of nuclear data was reported in [12] to predict the bias in nuclear criticality safety analysis using as input the sensitivity profiles. In [13], data from research reactor was used in learning models to predict core behavior during different core power levels and loss of flow accidents. ML has been applied in reactor core analysis to predict reactor core neutronics characteristics, optimize the fuel loading pattern of the reactor core, and perform sensitivity and uncertainty analysis. The ML models that have been applied include neural networks (NN), support vector machines (SVM), Gaussian process regression (GPR) and group method of data handling (GMDH) [14–20]. A detailed review of artificial intelligence method applied to in-core fuel management can be found in [21].

Most ML applications in RP is focused on training data generated with M&S tools that are computationally expensive to run. A different approach is employed in the current research. The data used in this learning process comes from publicly available measurement. ML tools are not yet widespread in SNF analysis and to the best of the author's knowledge, this study is the first to use ML to predict the decay heat of irradiated/discharged light water reactor (LWR) SNF assemblies. ML works on fuel loading pattern optimization and in-core fuel management referenced above are limited to fresh or partly spent fuel which are still to be further irradiated in the core. The purpose of this work is to develop a model that learns measured decay heat data of discharged LWR SNF assemblies, and then uses the knowledge learned to efficiently and accurately evaluate the decay heat of other LWR SNF assemblies discharged after irradiation from the reactor. The rest of this paper is organized as follows. First, the LWR SNF assembly decay heat measurement dataset used to train the learning models is described. Second, an exploratory analysis of the measurement dataset and the ML algorithms are presented. We propose to learn the relationship between measured decay heat of a fuel assembly and the assembly material/irradiation history information using models such as GPR, SVM and NN. Third, because regression techniques on small size nuclear engineering RP datasets have not been considered, the method of multiple training is proposed to obtain a model with an optimal training/testing set and good generalization capability. In addition, the use of synthetic data having the same statistical properties as the original dataset is proposed, and generated from the original dataset, to address problems encountered in the ML of small datasets. The stochastic nature of training data selection and hyperparameter optimization during training of ML algorithms causes statistical fluctuations that are more pronounced in small datasets and can cause large variance in the trained models. Fourth, the uncertainty analysis of the models developed, and their estimates are conducted to quantify the errors due to the training data selection, size of the training dataset and perturbations of the input features. Fifth, the conclusions and future perspectives are outlined.

2. Measured data of light water reactor spent fuel assembly decay heat

Three different sets of decay heat measurements are combined for use in this study. The three data sets contain benchmark LWR SNF decay heat data measured from the calorimetric analysis of fuel assemblies irradiated in 16 different LWRs. The three measurements were respectively carried out by:

- Hanford Engineering Development Laboratory (HEDL) in the United States. HEDL measured the decay heat the spent fuel assemblies from the Turkey Point reactor.
- General Electric Morris Operation (GE-Morris) in the United States. GE-Morris measured the decay heat of spent fuel assemblies from the San Onofre 1, Point Beach 2, Cooper, Dresden 2 and Monticello reactors.
- Swedish Nuclear Fuel Waste Management Company, Svensk Kärnbränslehantering AB (SKB). The decay heat measurement of spent fuel assemblies from reactors: Ringhals 2, Ringhals 3, Barsebäck 1, Barsebäck 2, Forsmark 1, Forsmark 1, Forsmark 2, Forsmark 2, Forsmark 3, Oskarshamn 2, Oskarshamn 2, Oskarshamn 3, Oskarshamn 3, and Ringhals 1, were performed by SKB at the Swedish interim storage facility, CLAB, between 2003 and 2010.

The three sets (*i.e.*, HEDL, GE-Morris and SKB measurements) of decay heat benchmarks are summarized and detailed in the references [22–26]. The characteristics of the measured fuel assemblies are summarized in Table 1. This table contains the measurement facility, reactor name, the assembly design, the fuel enrichment, number of fuel assemblies measured, number of measurements performed, fuel assembly discharge burnup, and the reactor type. HEDL and GE-Morris measurements were performed in 1980 and 1985, respectively, for assemblies which were irradiated in reactors operated in the United States. SKB measurements were performed for assemblies which were irradiated in reactors operated in Sweden. Information on the assembly designs and irradiation history of the fuel assemblies which were measured can be found in the references. The assembly designs comprise pressurized water reactor (PWR) and boiling water reactor (BWR) assemblies. The measured fuel assemblies have average enrichments from 1.0 to 4.0 wt% ^{235}U, average burnups from 5 to 51 GWd/t and cooling time after discharge from 2 to 27 years. A total of 91 measurements of 52 PWR assemblies and 171 measurements of 95 BWR assemblies are used in this study. The datasets are thus divided into two according to reactor types: PWR and BWR dataset. Fig. 1 presents the geometry of an example of PWR 17 × 17 and BWR 8 × 8 lattice designs.

SKB conducted series of measurements to provide data for the validation of computational tools applied in the analysis of SNF wet/dry storage facilities and repository in Sweden. The calorimeter design used at CLAB is similar to the calorimeter used at GE-Morris. The calorimeter is placed inside a pool of water and contains the assembly to be measured. Sensors are placed in the water inside the calorimeter, on the internal and external surfaces of the calorimeter and in the water outside the calorimeter to measure the increase in temperature. The increase in temperature of water in the calorimeter is compared to a calibration curve obtained from an electric heater. Gamma detectors are used outside the calorimeter to determine the gamma ray emission and to account for the energy loss due to gamma radiation. The calorimeter used at HEDL is a boil-off type calorimeter in which the decay heat of a fuel assembly is measured by differential steam condensate collection rates. Although these measurement data were acquired to support the validation of computational tools used in SNF analysis, it is reasonable to employ these data for more applications such as

Table 1
LWR fuel assembly decay heat measurement dataset.

Measurement Facility	Reactor	Fuel Design	Enrichment[a](wt.%^{235}U)	Assemblies Measured	Max. Burnup[a](GWd/t)	No. of Measurements	Reactor type
HEDL	Turkey Point	15 × 15	2.557	4	28.59	6	PWR
	San Onofre 1	14 × 14	3.865–4.005	8	32.36	8	PWR
	Point Beach 2	14 × 14	3.397	6	39.38	6	
GE-Morris	Dresden 2	7 × 7	2.128	1	5.28	2	BWR
	Cooper	7 × 7	1.090–2.500	54	28.05	81	
	Monticello	7 × 7	2.250	6	20.19	13	
CLAB	Ringhals 2	15 × 15	3.095–3.252	18	50.96	33	PWR
	Ringhals 3	17 × 17	2.100–3.404	16	47.31	38	
	Barsebäck 1	8 × 8	2.922–2.953	2	41.13	4	BWR
	Barsebäck 2	8 × 8	3.154	1	40.01	1	
	Forsmark 1	8 × 8	2.090–2.970	2	34.19	3	
	Forsmark 1	9 × 9	2.938	3	37.89	3	
	Forsmark 2	8 × 8	2.095	1	19.94	1	
	Forsmark 2	SVEA-64	2.850–2.920	3	32.84	3	
	Forsmark 3	SVEA-100	2.770	2	31.28	3	
	Oskarshamn 2	8 × 8	2.201–2.875	7	34.89	8	
	Oskarshamn 2	SVEA-64	2.902	1	46.65	1	
	Oskarshamn 3	8 × 8	2.577	1	25.16	1	
	Oskarshamn 3	SVEA-100	2.711	2	40.36	2	
	Ringhals 1	8 × 8	2.640–2.911	9	44.86	45	

[a] Assembly average value.

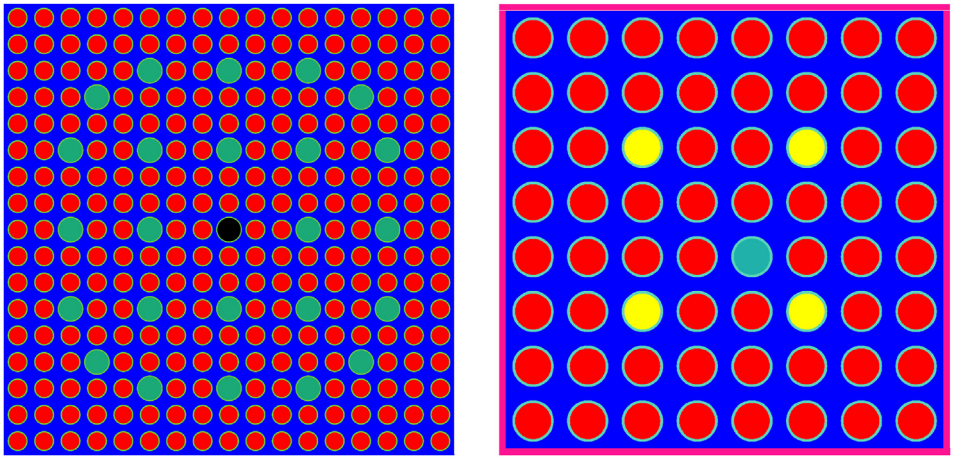

Fig. 1. Schematic of typical fuel assemblies: PWR 17 × 17 (left), BWR 8 × 8 (right). Legend: red (UO_2), blue (moderator), green (guide tube), black (instrument tube), yellow (UO_2-Gd_2O_3). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

machine learning and inverse uncertainty quantification. It becomes more reasonable considering that measurement data are costly to obtain. It would thus be desirable to have a tool which can learn the relationships between the measurements and the associated fuel assemblies so that the learned knowledge can be used to make predictions concerning other assemblies.

The decay heat measurement for one assembly takes several hours for high accuracy. Although another method is available for measuring decay heat of spent fuel such as gamma ray spectroscopy [27,28], decay heat measurements generally are usually scarce to come by. For the measurement data used in this research, the uncertainties are noted in the references. The decay heat uncertainty reported for the CLAB PWR measurement is ±9.2 W (3.7%) at 250 W and ±18.2 W (2.1%) at 900 W. For the CLAB BWR measurements, the uncertainties are given as ±4.2 W (8.4%) at 50 W and ±6.2 W (1.8%) at 350 W [23]. The quoted CLAB uncertainties are at the 95% confidence level and these are reported to be random errors. From these values, the uncertainty in each CLAB PWR or BWR measurement can be obtained by linear interpolation. For the HEDL measurements, the accuracy was estimated to be ± 5% for decay heat greater than about 1000 W, and ±10% at 100 W. The HEDL measurement uncertainties are not reported to be either systematic or random. At the GE-Morris measurement facility, the systematic error associated with the measurements is about ±2% for a thermal output of about 700 W, and ±4% in the 200 W range. In addition, the random error reported from repeated measurements of the same assembly is about ±4.3% [22]. It is also reported in [23] that the GE-Morris calorimeter gave an uncertainty of ±7.5% (at two standard deviations) at 200 W. Another report [29] indicates that the uncertainty of the US PWR fuel assembly decay heat measurement (*i.e.*, HEDL and GE-Morris) is quoted to be 2%.

3. Decay heat measurement data exploration

An exploratory data analysis of the decay heat measurement is performed. The correlations between the measured decay heat and some assembly parameters are examined for the PWR dataset and shown in Table 2. We can notice a small correlation between the measured decay heat and the assembly discharge burnup, ^{235}U enrichment, and initial heavy metal mass. However, the measured

Table 2
Correlation matrix of input and output features (PWR dataset).

	Decay heat (W)	Burnup (GWd/tU)	Cooling time (years)	Enrichment (wt.%^{235}U)	Uranium mass (kg)
Decay heat (W)	1				
Burnup (GWd/tU)	0.203	1			
Cooling time (years)	*−0.749*[a]	0.108	1		
Enrichment (wt.%^{235}U)	0.159	*0.603*	*−0.280*	1	
Uranium mass (kg)	−0.188	0.007	*0.572*	*−0.704*	1

[a] Values in italics correspond to significant correlation coefficient at the 0.01 significance level.

decay heat is strongly and negatively correlated with the cooling time after discharge. Correlations between the input parameters are further observed, most notably between the assembly discharge burnup and ^{235}U enrichment, and between enrichment and uranium mass. These correlations re-iterate the physics of the problem at hand. Moreover, the correlations also show that there is multicollinearity between the input features (burnup, cooling time, enrichment, and uranium mass). For example, it can be observed that the measured decay heat has a similar relationship with the burnup and enrichment. An input feature space including both burnup and enrichment may not improve the accuracy of decay heat estimation since they have similar correlation coefficients of 0.20 and 0.16, respectively, with the decay heat. The conclusion of multicollinearity is reached by calculating the significant Pearson correlation coefficients for the input features at the 0.01 significance level. We found that for some of the input features, the Pearson correlation coefficients are significant at the 0.01 significance level. These are shown in italics in Table 2. Additional test to examine the presence of multicollinearity was performed using the variance inflation factor (VIF) [30]. The VIF measures the degree of multicollinearity in an ordinary least square regression. It quantifies the amount of increase in the variance of a regression coefficient estimate due to collinearity. The VIF is analyzed in each iteration by considering only the input features shown per column in Table 3. A VIF <10 indicates that multicollinearity is low. Table 3 confirms earlier finding that burnup and enrichment have similar correlation coefficient with decay heat, and this is seen in the high VIF (see 1st iteration column of Table 3). When we exclude other input features, we found that retaining burnup and cooling time as input features yields VIF <10. It appears only these two inputs are relevant to predicting the decay heat. However, for reasons explained in the next section, all the four inputs will be included in the model. For the BWR dataset, the same VIF trends are noted and similar correlations (between measured decay heat and cooling time; between burnup and enrichment; between enrichment and uranium mass) are observed as in the PWR dataset.

3.1. Selection of input features

To screen the input parameters, it was decided to only select parameters that are available for all the assemblies measured. In addition, any input information not reported for some assemblies is excluded for all the remaining assemblies. Moreover, some input parameters reported in some of the benchmark documentations are not the actual values for the concerned assemblies. This may introduce uncertainties which are quantified later in Section 4.2. It is important to note that considering all the assembly design and irradiation history information, the dimension of the input space could be up to 17. This would be challenging to some machine learning models due to the curse of dimensionality. Furthermore, it will become a disadvantage to the machine learning models which are surrogates whose advantages are to reduce the dimension of the feature space. From the physics of the problem, many of the assembly design and irradiation history information are correlated and redundancies could exist in the input space. Moreover, previous sensitivity studies have highlighted some input parameters which have negligible effect on light water reactor (LWR) decay heat calculation results [31,32]. This physics based ML is important since the ML algorithms could be black box models. The input features considered are summarized in Table 4 alongside their ranges. The input features in Table 4 are finally selected after it was established that these information can be obtained for all the measured assemblies. Another reason these features are chosen is because they could be obtained for any other assembly aside those in the decay heat data set. This is an advantage since the models trained in this work could then be applied to predict integral decay heat of assemblies discharged from any LWR, provided the input data lie in the ranges specified in Table 4.

From the previous section, it was established that the decay heat has similar correlations with the burnup and enrichment. And that enrichment and uranium mass are responsible for high multicollinearity in the input space. Nevertheless, we decided to include the enrichment and uranium mass among the input features. The reason for this inclusion is because in many cases two different fuel assemblies discharged from a reactor may have similar burnup and in such cases the enrichment and/or the uranium mass might be useful to differentiate such assemblies. The PWR and BWR datasets are explored separately to detect extreme values which are

Table 4
Dataset input and output features.

Input features	Ranges	
	PWR dataset	BWR dataset
Decay time (days)	859–9734	857–9750
Discharge burnup (GWd/tU)	19.699–50.962	5.28–46.648
^{235}U enrichment (wt.%)	2.09–4.005	1.09–3.15
Heavy metal mass (kg)	361.72–463.898	126.68–195.48
Output feature		
Decay heat (W)	209.79–1550	19.5–395.40

Table 3
Variance inflation factor of input features (PWR dataset).

	1st iteration	2nd iteration	3rd iteration	4th iteration	5th iteration
Burnup	49.74	24.26	6.40	48.75	–
Cooling time	9.91	8.68	6.40	6.43	9.39
Enrichment	56.68	-[b]	–	41.41	27.64
Uranium mass	44.96	32.85	–	–	44.07

[b] Missing values means the feature is excluded in the VIF calculation.

considered outliers, for removal from the data set. The extreme values are only observed in the BWR dataset. Not included in this report, the visualization of the BWR dataset indicates that at burnup less than 10 GWd/tU, ^{235}U enrichment less than 1.5% and heavy metal mass less than 160 kg, the data points are far away from the general pattern in the data and there is scarcity of data around those points. Because of this, these data points are considered outliers and are removed from the BWR dataset. Specifically, 6 data points are removed from the BWR dataset. These correspond to assemblies with 5GWd/tU burnup, 1.0 wt% ^{235}U enrichment, and heavy metal mass between 120 and 160 kg.

3.2. Generation of synthetic data

The size of the PWR dataset is 91. After removing outliers, the BWR dataset has a size of 165. Each of these datasets are supposed to be split into training, validation, and testing set. The dataset is a small one and could bias the model and cause the trained model to perform poorly in out-of-sample cases. Small dataset is one of the major challenges for ML and many approaches have been developed to tackle the issue such as creating synthetic data [33,34], data augmentation [35], under-sampling the majority class and over-sampling the minority class [36]. Some other studies employ strategies that resample or bootstrap [37] which use the original dataset to create synthetic data that is a subset of the original one. The synthetic data can then be used to perform model training, testing and cross-validation. Bootstrapping has been particularly useful in evaluating the bias, variance and confidence intervals of ML models and their estimates. Another application of the bootstrap method in ML is found in the method of bagging (bootstrap aggregating) of ML models, *i.e.*, a given ML model is trained on different datasets bootstrapped from the original training data and the outputs of all the models on testing set are averaged in regression problems or voted on in classification task. Later in Section 4.2, we used the method of bootstrap to generate different training data to perform uncertainty analysis of the ML models and their predictions.

However, in this section, a novel approach in synthetic data generation is introduced. It is inspired by the Monte Carlo method used in uncertainty quantification (UQ) and sensitivity analysis (SA). It involves the following steps: (*i*) fit the training set of the PWR or BWR dataset to a model to obtain an accurate input to output mapping function (*ii*) assign uncertainties (standard deviation) to each input feature, the continuous values in each input feature are considered nominal values (mean/average) (*iii*) assume each input feature follows a normal distribution, then sample from this distribution using the mean and standard deviation of step *ii* to obtain perturbed values of the inputs for each dataset sample (*iv*) use model from step *i* to predict the outputs associated with the perturbed inputs generated in step *iii* (*v*) combine the set of perturbed inputs and corresponding outputs from steps *iii* and *iv*, respectively, to obtain the synthetic data generated.

Concerning the synthetic data generation, uncertainties assigned in step *ii* to the input features can be obtained from literature [38,39]. Another well-known probability distribution of choice can be employed in step *iii*. The selected probability distribution is to enable us to generate random samples of the input features. It is possible to determine the actual probability distribution of the input features which will then be used in the synthetic data generation. This is however beyond the scope of this work. An important feature of the synthetic data is that it has the same statistical characteristics as the original data. If this is not the case, then it is possible that the probability distribution selected is not sufficient to model the input features and should be changed. The fitted model in step *i* should have high accuracy in mapping the input features to the output response in our dataset. The proposed synthetic data generation method is different from the method of perturbing the original samples or adding noise to the input data [40–43]. Synthetic data should retain the statistical properties (mean, variance, and correlations) of the original data set. Synthetic data is used significantly in data-driven applications when the confidentiality of original datasets must be protected [44] or when the available dataset is small. We generated PWR and BWR synthetic data that mimics the properties of the original dataset which contains 91 and 165 measurements, respectively. The original data and synthetic data are compared in terms their probability distribution functions (PDFs) and correlation matrices. The correlation matrices of the original and synthetic data are the same and the PDFs for the PWR dataset are shown in Fig. 2. Moreover, Table 5 shows the summary of the statistics of both the original data and synthetic PWR dataset. As can be seen in Fig. 2 and Table 5, the statistical properties are consistent between the original and synthetic data. This implies that the distribution assumed, and uncertainties assigned to the input features are reasonable and justified. Fig. 2 and Table 5 excludes the uranium mass input feature because we considered it to be constant and was not perturbed in each input sample.

3.3. Training and testing methodology

The performance of a ML model depends on a number of factors, one of which is how the dataset is randomly divided into training and testing set. Due to the small size of the dataset used in this work, we decided to employ other strategies used in the ML of small datasets such as the method of multiple runs whose description is presented as follows. All the models developed on the original datasets are trained 1000 times to obtain a model that generalizes well to unseen data. This retraining is done to obtain properly sampled training and testing set. The training set should cover all the patterns in the data as much as possible. Each training begins with different divisions of the data into 90% training and 10% testing sets randomly. The models with the smallest mean absolute error (MAE) are the ones reported as the final models. The optimum hyperparameters of each model were found using the training set. Then by looking at the performances of the models on the testing set (which is not used for hyperparameter finetuning), the prediction capabilities are assessed. The dataset divisions does not include a validation set due to the small size of the measurement data. For the method of multiple runs, the dataset is divided 1000 times randomly into a training and testing set, while noting the random numbers used during each division, for reproducibility of results. The method is effective in that the final model gives smaller prediction errors compared to when the models are trained once. Due to the random nature of ML algorithms, the models produce different result every time they are run. Although the method of multiple runs requires more time, we gained more accuracy in the process. And the multiple training does not have to be done again since the final model can be easily reproduced and used for predictions as needed. The method of multiple runs has been applied to overcome the problem of ML of small datasets and shown to work in medical applications [7,52,53]. After training with the original dataset to establish a baseline performance, we then used synthetic data in another set of training. Please note that the synthetic data are generated using only the training set of the original dataset. As a result, the performances of the models developed with the synthetic data are assessed on the 10% testing set of the original dataset. However, since the synthetic data is relatively larger, the training with synthetic data was conducted one time per model. At least the models developed with synthetic data should have similar performance in terms of error, as the models based on the original

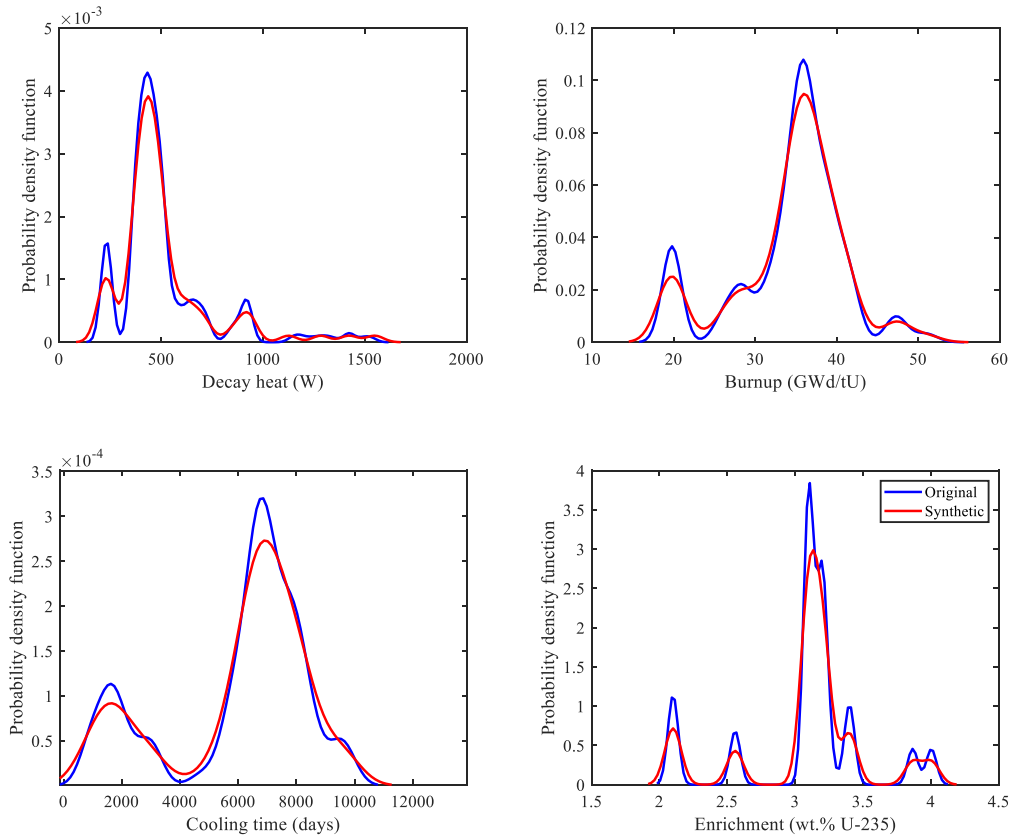

Fig. 2. Probability density function of original and synthetic data (PWR dataset).

dataset, if the synthetic data reflects the complex relationships between the features of the original dataset. Nevertheless, with a larger dataset, although synthetic, the developed models should perform better. The use of synthetic data in ML have been shown to work in myriads of statistical learning applications. Because the published works are numerous to be listed here, we refer the interested readers to a few of the references [7,10,52,54]. It should be mentioned that the ML models trained in this work are applicable to predict decay heat of assemblies whose features lie in the ranges specified in Table 4. This corresponds to decay time between 2 and 27 years, which is important for SNF in storage and transportation applications. For input data far from the ranges specified in Table 4, the accuracy of the models cannot be guaranteed because that will be equivalent to using the ML models for extrapolation. In such cases, additional data should be generated with a computer code and this is planned to be considered as part of future works.

Table 5
Original and synthetic dataset statistical summary (PWR dataset).

Dataset	Features	Mean	Std. Dev.	Minimum	Maximum
original	Heat (W)	510.82	242.19	209.79	1550.00
	Burnup (GWd/tU)	34.38	6.77	19.70	50.96
	Cooling (days)	6036.87	2434.28	859	9734
	Enrichment (wt.% ^{235}U)	3.09	0.45	2.10	4.01
synthetic	Heat (W)	511.26	237.99	210.15	1541.02
	Burnup (GWd/tU)	34.39	6.75	19.23	52.20
	Cooling (days)	6032.79	2422.84	829.16	9988.71
	Enrichment (wt.% ^{235}U)	3.09	0.45	2.10	4.01

3.4. Machine learning methods

To conclude on the use of ML algorithms which can model complex and non-linear input-output relationships, we examined the individual relationships between a fuel assembly decay heat and its burnup, enrichment, cooling time and uranium mass. The results are presented in Fig. 3. These results are calculated with the reactor analysis code STREAM [45]. The decay heat versus burnup, decay heat versus cooling time, and decay heat versus enrichment results are calculated for a typical 17 × 17 PWR fuel assembly. The decay heat versus uranium mass results are obtained for different PWR fuel assembly designs: 14 × 14, 15 × 15, 16 × 16, 17 × 17, and 18 × 18, with varying uranium mass content. For each input-output relationship, *e.g.*, decay heat versus burnup, the other input features (cooling time, enrichment, and uranium mass) are constant. Fig. 3 shows that the decay heat versus burnup and decay heat versus enrichment relationship is quadratic or parabolic function. It further shows that the decay heat versus cooling time relationship is an exponentially decreasing function. This relationship is well-known. In addition, from Fig. 3, the decay heat versus uranium mass have a linear relationship. The main takeaway from Fig. 3 is that we need non-linear models to obtain the best solution to predict LWR SNF assembly decay heat.

3.4.1. Gaussian Process

Gaussian process regression (GPR) modeling is a ML technique that is based on statistics, probability, and interpolates between functions rather than points. GP is a stochastic model which considers the output of a model as a Gaussian process (GP). The GP of a model output Y is given according to the equation:

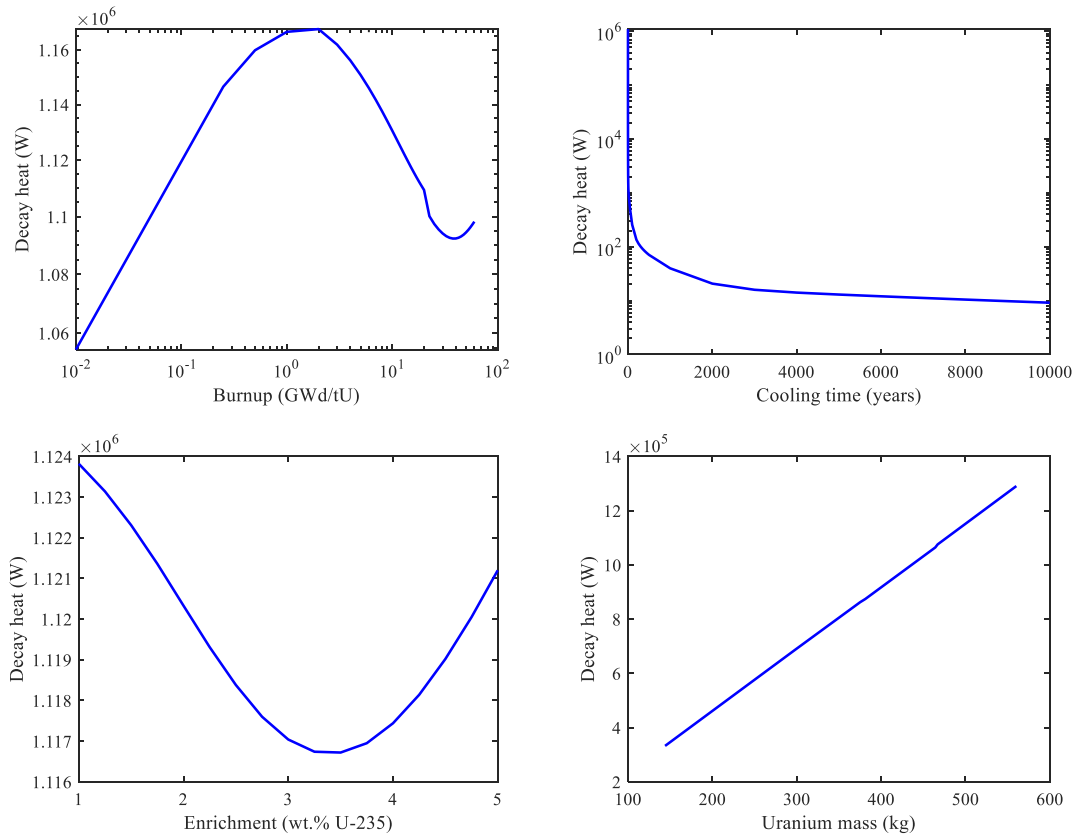

Fig. 3. Relationship between fuel assembly decay heat and burnup, cooling time, enrichment, and uranium mass. The decay heat relationship with these assembly features is a non-linear function which requires a complex model to capture.

$$Y(\xi) = \sum_{n=0}^{q} a_n \psi_n(\xi) + \sigma^2 Z(\xi, w) \tag{1}$$

where ξ are the model input variables. The first term on the right hand side of Eq. (1) is known as the trend (or mean value) which consist of coefficients term (a_n) and arbitrary functions (ψ_n) of order q. Different options exist for the choice of the trend such as a constant value a_0, linear $a_0 + \sum_{n=1}^{q} a_n \psi_n$, quadratic $a_0 + \sum_{n=1}^{q} a_n \psi_n + \sum_{n=1}^{q} \sum_{m=1}^{q} a_{nm} \psi_{nm}$ or polynomial functions. The second term is made up of the variance of the GP σ^2 and a stationary GP $Z(\xi, w)$ with zero mean and unit variance. The probability space associated with GP is denoted by w and is defined based on a family of correlation functions $R = R(\xi, \xi'; \theta)$ having hyperparameters θ. The correlation function is also referred to as the kernel or covariance of the GP, and different families of the GP correlation functions are available to be selected from, including the following:

$$\text{Linear},\ R(\xi, \xi'; \theta) = \max\left(0,\ 1 - \frac{|\xi - \xi'|}{\theta}\right) \tag{2}$$

$$\text{Exponential},\ R(\xi, \xi'; \theta) = \exp\left(-\frac{|\xi - \xi'|}{\theta}\right) \tag{3}$$

$$\text{Gaussian},\ R(\xi, \xi'; \theta) = \max\left(-\frac{1}{2}\left(\frac{|\xi - \xi'|}{\theta}\right)^2\right) \tag{4}$$

$$\text{Matérn} - 3/2,\ R(\xi, \xi'; \theta) = \left(1 + \sqrt{3}\ \frac{|\xi - \xi'|}{\theta}\right) exp\left[-\sqrt{3}\ \frac{|\xi - \xi'|}{\theta}\right] \tag{5}$$

$$\text{Matérn} - 5/2,\ R(\xi, \xi'; \theta) = \left(1 + \sqrt{5}\ \frac{|\xi - \xi'|}{\theta} + \frac{5}{3}\left(\frac{|\xi - \xi'|}{\theta}\right)^2\right) exp\left[-\sqrt{5}\ \frac{|\xi - \xi'|}{\theta}\right] \tag{6}$$

The hyperparameters θ of the correlation function are also known as characteristic length scale or scale parameter. They are

obtained by finding the solution of an optimization problem using the dataset provided for training the GPR. Along with the other GP parameters a_n and σ^2, the hyperparameters are determined by maximum-likelihood estimation of the actual response or minimizing the cross-validation error. The GP assumes that the vector of output predicted by the GPR and the actual output from a given dataset, have a joint Gaussian distribution which considers the correlations between the inputs. The statistics of the GP, *i.e.*, the mean and variance are then determined conditional upon the actual inputs and outputs [46]. These statistics are required to make new predictions at arbitrary points. One of the advantages of the GP is that the uncertainty and the confidence interval of its point estimates can be obtained.

3.4.2. Support vector machines

For a given dataset with input and output features, SVM for regression tasks seeks to find a mathematical relationship of the form:

$$Y(\xi) = w^T \phi(\xi) + b \tag{7}$$

where w is a vector containing weight coefficients, ϕ is a mapping function and b is an offset parameter. w and b are determined by minimizing a loss function. For non-linear problems, the input data is projected into a high dimensional feature space by an inner product $\phi^T(\xi_i)\phi(\xi_j)$ or using a kernel function $k(\xi_i, \xi_j)$. This projection transforms the problem into a linear form in the high dimensional feature space:

$$Y(\xi) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) \phi^T(\xi_i)\phi(\xi) + b \tag{8}$$

$$Y(\xi) = \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) k(\xi_i, \xi) + b \tag{9}$$

where α_i and α_i^* are coefficients (also known as support vectors) to be determined and when obtained, Eq. (9) can be used to make predictions at arbitrary points. The coefficients are evaluated by maximizing the Lagrangian function:

$$L(\alpha_i, \alpha_i^*) = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} (\alpha_i - \alpha_i^*)\left(\alpha_j - \alpha_j^*\right) k(\xi_i, \xi_j) - \sum_{i=1}^{n} (\alpha_i + \alpha_i^*)\varepsilon + \sum_{i=1}^{n} (\alpha_i - \alpha_i^*) y_i \tag{10}$$

with the constraints:

$$\sum_{i=1}^{n} (\alpha_i - \alpha_i^*) = 0 \; and \; 0 \le \alpha_i, \alpha_i^* \le C, \quad i = \{1, \ldots, n\} \tag{11}$$

where ε is the loss function, C is a regularization parameter and ξ_i, y_i is an input/output set. This is an optimization problem and can be solved by quadratic programming. This solution also produces the value of the b term in Eq. (7). Various kernel functions are employed for SVM such as:

$$\text{Linear}, \; k(\xi, \xi') = \xi^T \xi' \tag{12}$$

$$\text{Polynomial}, \; k(\xi, \xi') = \left(\xi^T \xi' + d\right)^p \tag{13}$$

$$\text{Sigmoid}, \; k(\xi, \xi') = \tanh\left(\frac{\xi^T \xi'}{a} + b\right) \tag{14}$$

where p is the polynomial degree, d, a and b are kernel parameters. In addition, the correlation functions listed in Section 3.4.1 for GPR (*i.e.*, Eqs. (2) – (6)) can be used as kernel functions for SVM and these would have hyperparameter θ. The hyperparameters of the SVM are C, ε and θ, and their optimum values are determined by minimizing the error between the SVM predictions and actual response of the given dataset [47].

3.4.3. Neural network

Neural networks (NN) are arrangements made up of connected neurons as in the human brain. NN are trained by adjusting this connections (which are referred to as weights). In learning the relationships present in a data, NN adjusts itself to map a given input to its corresponding output. In doing this, the NN compares its own output to the output present in the dataset. The difference between these outputs then determines if the weights used needs to be adjusted. Until these two outputs agree, the adjustment of the NN continues. All inputs to the NN are assigned to one neuron and many neurons can be employed in one layer of a NN. The inputs are weighted individually, summed up and a bias is added. A bias term is added to each neuron in a layer of the NN to introduce non-linearity into the process through an activation or a transfer function. The sum of the bias and weighted inputs is the input to the transfer function. Different types of activation functions exist including sigmoid, linear, and radial basis function. The adjustment of the NN involves adjusting the weights and biases of the network. A single layer of the NN is made up of neurons and transfer functions and many layers can be connected in one NN. These are also known as hidden layers, to differentiate them from the input and output layers which contains the NN inputs and outputs, respectively. In a multilayer NN, the outputs of one hidden layer serve as inputs to the next layer. The difference between the NN output and actual output in the dataset is also used to determine the NN optimal performance. This performance is quantified in terms of an error or loss function such as the mean square error. The gradient or Jacobian of the error function with respect to the weights are calculated, after which the weights and biases are adjusted. This process of performing calculations backwards through the NN is known as backpropagation, in contrast to forward propagation which is the process of performing computations forward from the input layer, through the hidden layer(s) to the output layer [48]. The schematic of the NN used in this work is shown in Fig. 4 below where $\boldsymbol{W}$ represent the weight vector, $\boldsymbol{b}$ is the bias vector, $f(\cdot)$ is the activation function, $\boldsymbol{a}$ is the output vector from each hidden layer, $\boldsymbol{X}$ is the input vector and $\boldsymbol{Y}$ is the output. The hyperbolic tangent sigmoid and linear activation functions which can be used in the hidden and output layers, respectively, are defined in Eqs. (15) and (16), where $h = WX + b$, and X represents the vectors $[X_1\ X_2\ X_3\ X_4]$, a_1, a_2, a_3, subsequently, as we progress from the input layer, through the hidden layers, to the output layer. In Eq. (16), $a = a_3$, *i.e.*, the output from the third hidden layer. The weights and bias of the final original PWR and BWR NN model are provided in the data file supplemental to this paper.

$$f(h) = \frac{2}{1 + e^{-2h}} - 1 \tag{15}$$

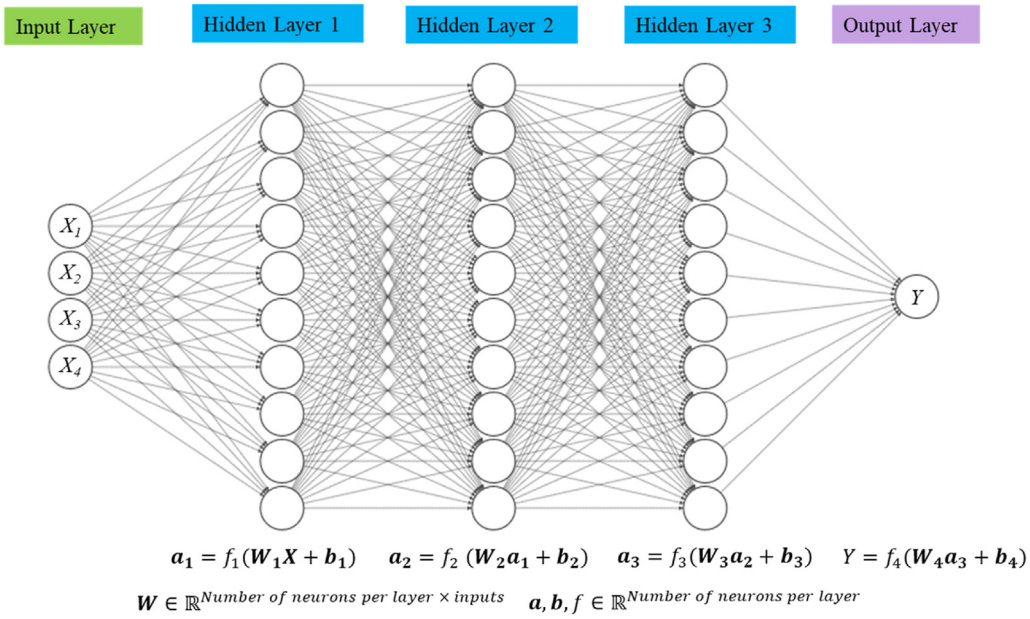

Fig. 4. Schematic representation of three layer feedforward neural network used in this work.

$$Y = f(Wa + b) = Wa + b \tag{16}$$

3.5. Model evaluation

The performances of the ML models are obtained by comparing the decay heat predictions of the trained models and the measured decay heat. The performances of the developed models are evaluated in terms of their MAE and root mean square error (RMSE) calculated respectively as:

$$MAE = \frac{1}{N}\sum_{j=1}^{N}\left|y_{model} - y_{ref}\right| \tag{17}$$

$$RMSE = \sqrt{\frac{\sum_{j=1}^{N}\left(y_{model} - y_{ref}\right)^2}{N}} \tag{18}$$

where y_{model} is the ML model predicted decay heat, y_{ref} is the measured decay heat and N is the number of samples in the training or testing set. The RMSE and MAE are given in the same units as the decay heat which is Watts (W). The RMSE is the square root of the average squared error between the predicted decay heat and the measured decay heat. The MAE is the average absolute error between the model outputs and the measured decay heat. The closer the RMSE and MAE to zero, the better the performance of the developed model. For the GPR and SVM models, the implementations in the MATLAB UQLab tool [49] are used. For the GPR training on the original PWR and BWR datasets, we used the constant trend and exponential correlation function. The GPR trend contains the basis function. For a constant trend, the basis function has order $q = 0$ in Eq. (1) and $\psi_0(\xi) = 1$. The exponential kernel in Eq. (3) is used in training the SVM on the original PWR and BWR datasets. For the SVM, the mapping function in Eq. (7) is usually replaced by the kernel function which is the inner product of the mapping function. In training the GPR on the synthetic PWR dataset, we used the quadratic trend and exponential correlation function. The leave-one-out cross-validation (LOOCV) technique is employed during the GPR and SVM training. LOOCV is the special case of k-fold cross-validation (CV) for which k = size of training set. A testing set (10% of the original dataset) not used in training is used to assess the performance of the trained models. Before training the GPR and SVM models, the dataset is standardized by scaling to give each feature zero mean and unit variance. In this work, the NN is implemented in MATLAB using the feedforward network with backpropagation. For the NN training, the dataset is scaled into [-1 1] using the *mapminmax* function. For the original and synthetic PWR datasets, we used three hidden layers with 10 neurons per hidden layer plus one output layer (see Fig. 4). Four hidden layers with 10 neurons per hidden layer are used for the BWR original and synthetic datasets. The physics of a BWR core and fuel assembly is different and more complex than that of a PWR core and fuel assembly. Hyperbolic tangent sigmoid activation functions are used in the hidden layers and linear activation function is used in the output layer (see Eqs. (15) and (16)), respectively). The NNs based on the original datasets are trained using the Bayesian regularization algorithm using 1000 training epochs. For the synthetic data, the NNs are trained with the Levenberg-Marquardt algorithm, coupled with early stopping. This means that the error during training is monitored and when this error begins to increase, the training stops. Then the weights and biases when the error is smallest are returned. In this case the number of training epochs is determined by this early stopping. The original and synthetic dataset NN models use training functions (*i.e.*, Bayesian regularization and Levenberg-Marquardt algorithms) which do not use learning rate. To reduce computation time of the NN because we trained the NN 1000 times, the computations are run using the

Table 6
PWR Model performance (original dataset) (W).

Model	Training		Testing		
	RMSE	MAE	RMSE	MAE	Maximum absolute error
GPR	<0.01	<0.01	2.56	2.27	3.82
SVM	0.02	0.02	1.85	1.39	3.84
NN	2.43	1.52	1.26	1.10	2.32

Parallel Computing Toolbox during training and testing. In training all the models, all other hyperparameters are left at their default values after performing series of finetuning.

4. Results and discussions

4.1. Model performance

In this section, we discuss the performance of the developed models based on the PWR and BWR original and synthetic data sets. The model's training and testing performances on the original PWR dataset are shown in Table 6. On the training set, the GPR and SVM show high prediction and generalization capacity with RMSE and MAE less than 0.1 W. However, the NN has an RMSE and MAE of 2.43 W and 1.52 W, respectively, which is also a good prediction and generalization capability. The GPR and SVM models achieved high prediction accuracy and showed better performance in training, compared to the NN. This is likely because they are relatively simpler models and have relatively fewer hyperparameters to be tuned. In addition, they are well suited for problem with small samples, compared to NN that can handle large datasets. For the testing set, the GPR and SVM have MAE values of 2.3 W and 1.4 W, respectively. The NN has an MAE of 1.1 W. The NN performs better than the GPR and SVM on the testing set, generalizing better in out-of-sample cases. These results are very encouraging, considering the size of the dataset, as we were able to learn the input and output relationship in the data. Moreover, there is not much difference in the training and testing RMSE and MAE of the NN. This is the reason why the NN performed better on the testing set than the GPR and SVM models. Another reason is because the NN is better able to learn complex and non-linear relationships between inputs and outputs of a dataset. The predicted decay heat is compared against the measured values in Fig. 5 which includes a linear fit to show that the model predictions match the measured values. The 95% confidence interval (CI) calculated from the standard deviation predicted by the GPR model is added to the GPR predictions of the testing set in Fig. 6. All the 95% confidence interval shown in this paper are the ±2 standard deviations from the mean. The 95% CI for the training set are not shown because they are too small (on the order of 10^{-3}) to be displayed on the plots.

The performances of models developed with the original BWR dataset are shown in Table 7 and the predicted decay heat is compared against the measured values in Fig. 7. The GPR have high predictive capability on the training set. The NN has the largest training error with MAE of ~9 W and the SVM shows the largest error on the test set at 7 W MAE. The GPR has the best performance among the three models on the training and testing set.

Concerning the synthetic dataset, the results are shown in Tables 8 and 9 for the synthesized PWR and BWR dataset, respectively. Due to longer training times of the GPR and the calculations could not be parallelized, only the NN model employ larger sizes of

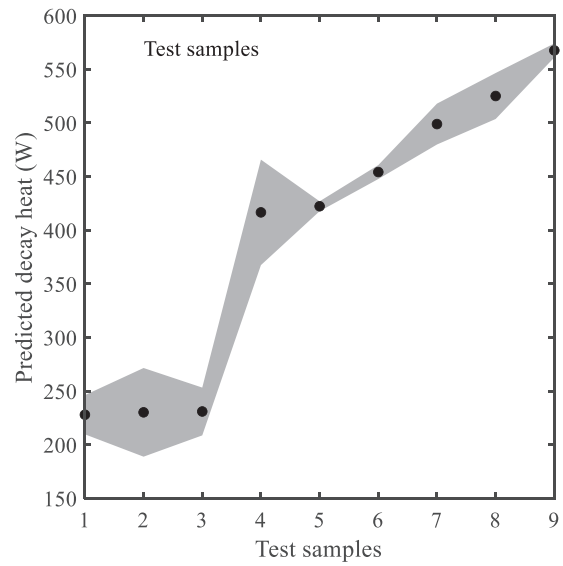

Fig. 6. GPR predictions and uncertainties at 95% confidence interval (original PWR dataset).

Table 7
BWR Model performance (original dataset) (W).

Model	Training		Testing		
	RMSE	MAE	RMSE	MAE	Maximum absolute error
GP	0.04	<0.01	5.36	4.28	13.53
SVM	6.59	4.22	9.92	7.07	24.54
NN	13.46	9.30	6.60	5.56	15.77

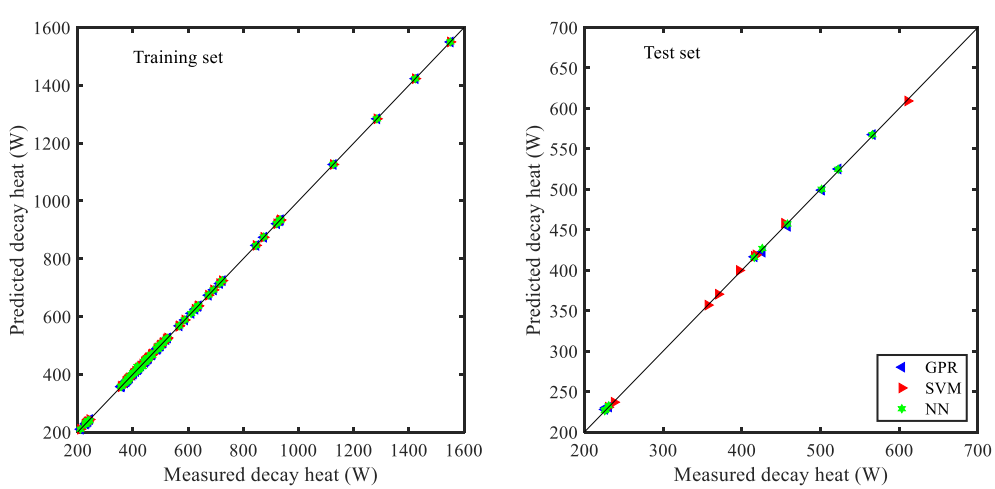

Fig. 5. Comparison of measured and predicted decay heat (original PWR dataset).

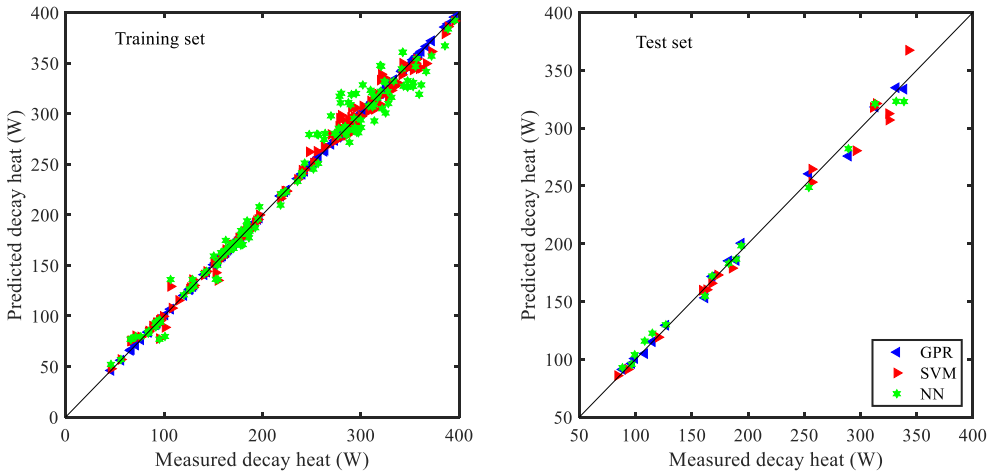

Fig. 7. Comparison of measured and predicted decay heat (original BWR dataset).

Table 8
PWR Model performance (synthetic dataset) (W).

Model	Training		Testing		
	RMSE	MAE	RMSE	MAE	Maximum absolute error
GPR (820 samples)	<0.01	<0.01	1.70	1.47	2.77
SVM (287 samples)	0.01	0.01	2.39	1.92	4.54
NN (41,000 samples)	1.33	0.89	1.10	1.05	1.62

Table 9
BWR Model performance (synthetic dataset) (W).

Model	Training		Testing		
	RMSE	MAE	RMSE	MAE	Maximum absolute error
NN (44,700 samples)	1.50	0.88	6.79	5.05	12.94

the synthetic data. The synthetic data size for each model is shown in Tables 8 and 9. For the PWR dataset and models (GP and NN) developed with synthetic data, the SVM was used as the fitted model to generate synthetic data (see Section 3.2). Comparing Tables 6 and 8, the RMSE/MAE of the GPR and NN models on the synthetic dataset are similar to those of the original dataset, considering the testing set. This outcome is similar to the finding in the reference [54] where the authors realized similar level of accuracy between the original small dataset and the synthetic large dataset models. Although the GPR shows some improvement in terms of a decrease in the error. Please recall that both the original and synthetic models use the same test set (10% of the original dataset) to assess model performance. For the BWR models with original and synthesized data, the NN has similar performance as can be seen comparing Tables 7 and 9. For the BWR NN model with synthetic data in Table 9, the GPR model was used to generate the synthetic data to train the NN. The maximum absolute error on the PWR and BWR test sets decreases from the original to the synthetic dataset models. This means that the use of synthetic data improves the performances of the models.

The fact that the synthetic data model have better performance on the testing set could be difficult to explain, especially since the synthetic dataset and its ML models are approximations of the original dataset. This could be due to statistical fluctuations. The limitation of this study concerns the size of the dataset. The size of testing set is also small, 9 samples. The primary contribution of this study is the methodology and a demonstration of its feasibility and applicability. The SVM model trained with synthetic data was applied to the test set to verify its accuracy. This result is included in Table 8. Compared to the original dataset SVM model in Table 6, the synthetic SVM model in Table 8 shows slightly higher error on the test set. This finding is similar to the outcome reported in [52] where the real dataset model outperforms the synthetic dataset model. However, with the performances shown in Table 8, the accuracy of the SVM model used to generate the synthetic data is guaranteed. Furthermore, this guarantees the accuracy of the output from the models built with the synthetic data. Not only did we confirm the similarity between the statistical characteristic of the synthetic data and original data (see Sec. 3.2), the accuracy of the SVM model used to generate the synthetic data is also verified. Further verification of the SVM model based on synthetic data is pursued by comparing its performance to another SVM model built with synthetic data obtained by bootstrap method. For the bootstrap synthetic data, 1000 set of samples (each set having the same size as the training set) are drawn from the original training set, by sampling with substitution. The bootstrap replications are then used to train an SVM model, which is used to predict the output of the training and testing set. The outputs are then averaged and compared to original dataset output. The result is shown in Table 10, where it can be seen that the synthetic data model from the SVM outperforms those of the bootstrap. Table 11 presents the computation time required to train the original and synthetic dataset models once. The computation time to generate synthetic data is on the order of seconds. The accuracy and computing time are both important in engineering applications. In some applications, accuracy is more important than the computing time and vice versa in other applications. Our recently published works on inverse uncertainty analysis by Bayesian method [50] and forward uncertainty analysis [32] demonstrate some applications of ML models which cannot be dealt with by time consuming computer code systems.

4.2. Uncertainty quantification

The discussion of uncertainties in the models developed and their point estimates is presented in this section. The uncertainties are presented only for original PWR dataset as a demonstration of the proof of concept. In RP, assembly design information are proprietary. When such details are needed during M&S, typical values

Table 10
PWR SVM Model performance (synthetic dataset) (W).

Model	Training		Testing		
	RMSE	MAE	RMSE	MAE	Maximum absolute error
Synthetic data from bootstrap	29.01	11.34	7.73	4.33	21.94
Synthetic data from SVM	0.01	0.01	2.39	1.92	4.54

Table 11
Computation time in training (seconds).

Model	PWR		BWR	
	Original	Synthetic	Original	Synthetic
GPR	0.90	60.80	1.03	–
SVM	3.83	27.46	13.44	–
NN	18.04	15.74	33.31	51.38

Table 12
PWR fuel assembly input parameters and their uncertainties at one sigma.

Parameter	Uncertainty (%)	Distribution	Ref.
Burnup (GWd/tU)	1.6	Normal	[38]
Decay time (days)	1.5		
Enrichment (wt.% ^{235}U)	0.05		[39]

might be used which may not be the actual value for some problem specifications. Moreover, when operation data such as irradiation history of an assembly are declared by the reactor operators, they are not without uncertainties due to how the information is obtained either by code calculations or measurements. In other words, uncertainties can be introduced due to perturbations of the input features considered. The solution is to give an error estimate of the output to increase confidence in the results. In addition, uncertainties will exist in the point estimates from ML models due to the training data selection and size of the training set. This section quantifies the uncertainty in the ML models and their decay heat estimates, due to these two sources. The uncertainty analysis (UA) is performed on the original PWR datasets.

In the first UA, the uncertainties in the selected input features (see Table 4) are considered. Due to uncertainties, there are studies in safeguard and non-proliferation applications [51] which aim to verify some nuclear fuel assembly model parameters even when such information are declared by the reactor operators. The input data and associated uncertainties are assumed to follow a normal distribution. The uncertainties at one sigma value in the input data are obtained from literature and presented in Table 12 for the PWR assemblies. Due to lack of uranium mass uncertainty information in literature, the uranium mass was considered unperturbed, and its nominal value was left unchanged in each sample. The input parameters are assumed to be independent so that correlations between them are neglected. The first UA is the conventional Monte Carlo method and it is summarized in the following steps: (*i*) assign uncertainties to all the input features as shown in Table 12 (*ii*) for each of the 91 PWR dataset records, generate 10^5 samples of perturbed input data following a normal distribution by Latin Hypercube Sampling (LHS) (*iii*) evaluate the decay heat using the perturbed data as inputs in the trained ML models (GPR, SVM and NN) (*iv*) perform the needed statistics on the decay heat evaluations. These steps are repeated for all the fuel assembly measurement data in the PWR dataset. The nominal input values are in the dataset and the input uncertainties are listed in Table 12. All the original PWR dataset records are used to train the models in performing the first UA *i.e.*, UA due to perturbed inputs features. The models developed in this work serve as cheap surrogates for fast evaluation of the decay heat, despite the large number of evaluations required in the UA. The generation of 10^5 samples of each perturbed input record and calculation of the decay heat by the model takes about 2 min for the entire dataset. This is another application of ML models which cannot be dealt with by time consuming computer code systems. For the PWR dataset, a total of 9.1×10^6 perturbed inputs are generated in this first UA for which the perturbed decay heat output should be calculated. For a computer code this will be computationally prohibitive. However, with the ML models developed in this work, the large number of perturbed outputs can be calculated in very few minutes. This demonstrates the effectiveness of the ML models suggested in this work in applications where time consuming computer code systems cannot be applied. The GPR and SVM models accurately map the input features to the decay heat response because their training errors are very small. The results of the UA due to uncertainties in the input data are shown in Figs. 8 and 9. The uncertainties reported in Fig. 8 corresponds to the relative standard deviation (*i.e.*, ratio of decay heat standard deviation to the mean) and these are presented against varying fuel assembly burnup, decay time and enrichment, to identify physics based trend. Because of the presence of high multicollinearity in the input space (see Section 3), it is difficult to identify any trend in Fig. 8. Fig. 9 shows the decay heat PDFs of 10^5 runs of two different records from the dataset. The absolute uncertainties are also shown in Fig. 9. First we observe, as expected, in Fig. 9 that the uncertainties due to perturbed input features are similar in the three models. Second observation is that the decay heat uncertainties due to perturbed input features are not large considering that the CI bands are small. Despite the small uncertainties in Fig. 9, it is important that such error estimates can be quantified to increase confidence in ML predictions. An important aspect of the models developed in this work is that the predictions can have intervals where the true value lies at a specific confidence level.

The second UA is due to the selection of training data and the size of the input data used in training. This UA is carried out by bootstrap method. The steps involved in the second UA can be stated as follows: (*i*) split the PWR dataset into 90% training and 10% testing set (*ii*) draw 100 sets of bootstrap samples from the training set, by resampling with substitution to create bootstrap replications, each replication having the same size as the training set (*iii*) use each bootstrap sample to train the models (*iv*) use the trained models to predict the decay heat of the original training and test sets of step *i*. For each ML algorithm, the different bootstrap training samples will produce a set of 100 different models which are then used to generate 100 sets of responses *i.e.*, a distribution of training and testing outputs. These are then used to calculate the standard deviation of the ML model point predictions. The mean and ±2 standard deviations from the mean of the GPR, SVM and NN point estimates are shown in Fig. 10 – 12. Generally, for the GPR and SVM in Figs. 10 and 11, respectively, the CI on the training set is small. These models show very good performance on the training set (see Table 6). Another reason is the bootstrap samples are drawn from the training set. However, the CI shows high values at some training points because the 100 different models developed from

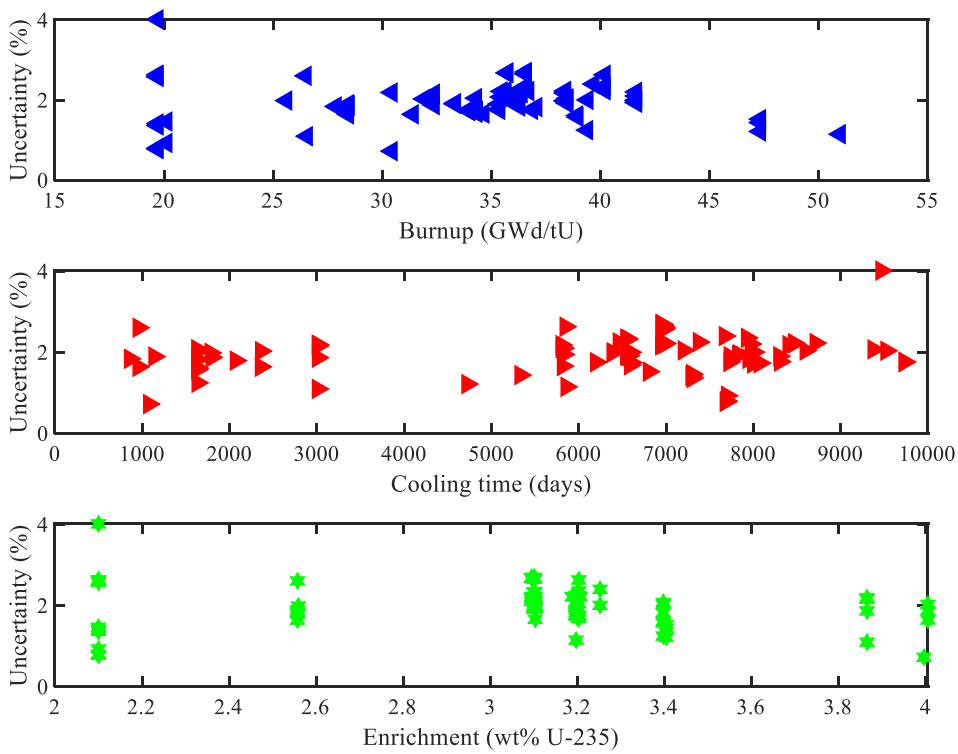

Fig. 8. Decay heat relative uncertainty due to perturbed input features (original PWR dataset).

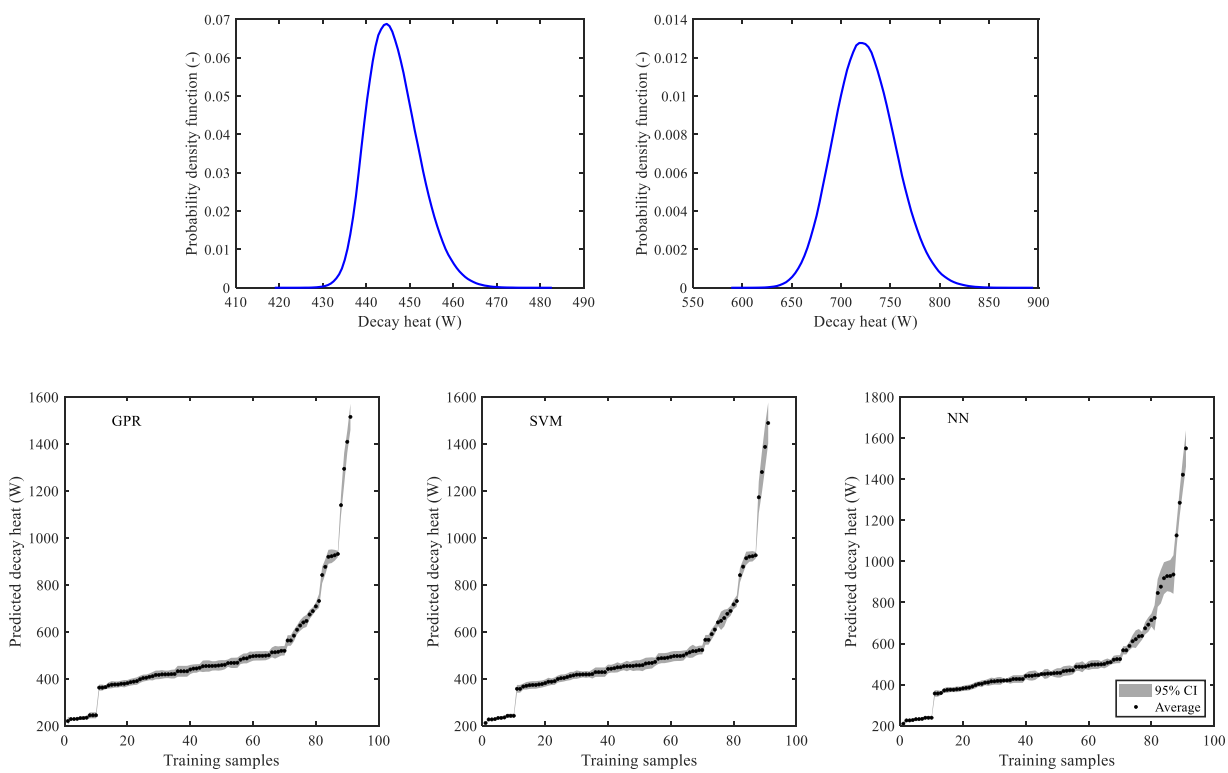

Fig. 9. Decay heat PDFs from 10^5 runs of two different records in the dataset (top). Decay heat absolute uncertainty due to perturbed input features (bottom) (original PWR dataset).

random bootstrap samples have large variance at those training points. This is due to statistical fluctuations associated with the random sampling, causing it to require large number of samples to attain convergence. For the test set, the GPR and SVM CI are larger

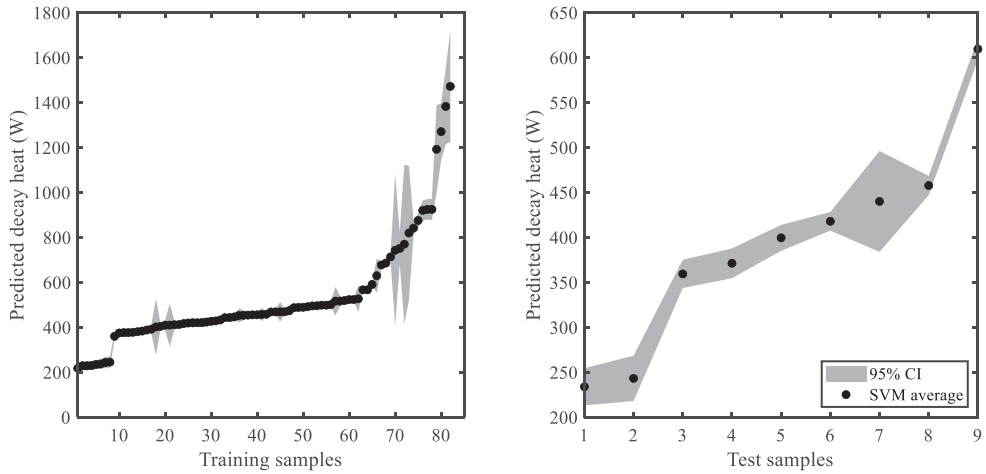

Fig. 10. SVM predictions and uncertainties due to training data selection and size (original PWR dataset).

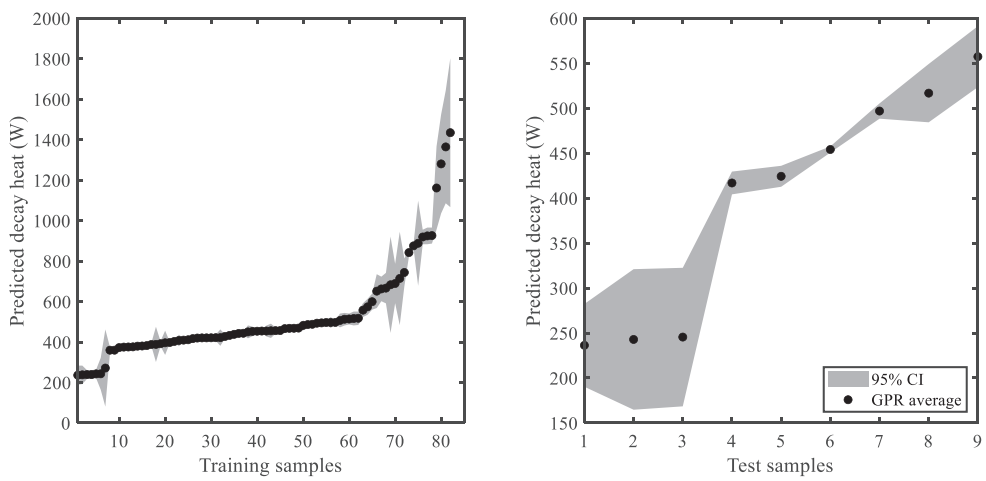

Fig. 11. GPR predictions and uncertainties due to training data selection and size (original PWR dataset).

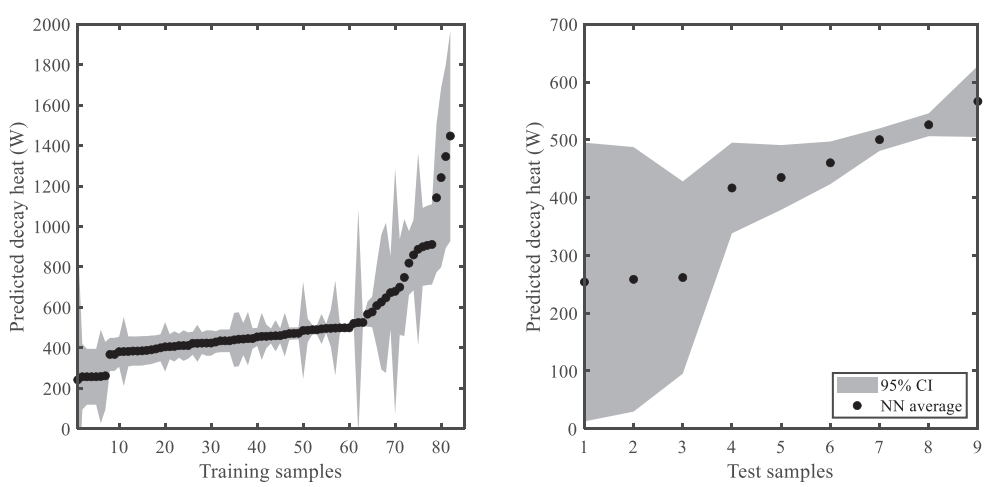

Fig. 12. NN predictions and uncertainties due to training data selection and size (original PWR dataset).

Table 13
Maximum standard deviation of NN predictions (W).

Dataset size	Training set	Test set
91 (original)	349.05	120.55
910 (synthetic)	56.96	86.75
9100 (synthetic)	12.17	10.01
18,200 (synthetic)	10.36	4.92

than those of the training set because these models show large variance on the test set. Fig. 12 shows the NN CI on the training and testing set. A large CI on the test set is observed. The NN may perform poorly on small dataset, and most importantly, is more prone to statistical fluctuations which occur during the random division of data into training and testing sets, random selection of the weights and biases, optimization of hyperparameters during NN training, and the random shuffling of the training set before each epoch. Thus, the effect of these fluctuations and the error due to limited size of the training set will be relatively larger in the NN than for models that are well suited for small datasets and less susceptible to statistical fluctuations such as GPR and SVM.

The error due to the limited size of the dataset can be decreased by acquiring a larger dataset which is possible with synthetic data. This would also eliminate or reduce the effect of the statistical fluctuations. This is summarized in Table 13 showing the maximum standard deviation of NN predictions on the original and synthetic PWR dataset. Besides, as shown in Fig. 13, the CI on the original training and test set when the NN model is built with larger synthetic dataset is reduced, compared to Fig. 12. Table 13 and Fig. 13 are the results obtained from applying the method of bootstrap to the synthetic dataset. The performance of the large synthetic data can be further explained for example, in the NN model as follows: the NN employs a function approximation with many unknowns. With small dataset, the approximating function shows fluctuations due to the paucity of input/output data. These fluctuations are reduced or eliminated when the dataset is large because the approximating function becomes smooth.

That the point estimates of the ML models can be provided with associated uncertainties is important in establishing accuracy and reliability. Furthermore, the quantification of the uncertainties present in the point estimates will be important if the models are used for extrapolation *i.e.*, the input features are outside the range shown in Table 4. Quantifying uncertainties from all known sources and then combining such information with measurement data (if available), measurement uncertainties (if known), the model predictions and bias, is a necessary step in obtaining best estimate plus uncertainty values. Despite the large uncertainties on the small dataset predictions, the differences between the predicted and measured decay heat are not statistically significant. These differences are within the statistical and measurement uncertainties. In addition, the model bias is not included in the CI calculations because the square of the bias is much less than the variance of the predictions.

5. Conclusions and perspectives

A supervised learning approach which enables the prediction of LWR SNF assembly decay heat when given their discharge burnup, decay time after discharge, fuel enrichment and initial heavy metal mass is developed. By training on 256 decay heat data covering LWR assemblies from calorimetric measurements, the ML models make informed estimations of integral decay heat. The models developed can serve as surrogates for measurement. Due to the small size of the measurement dataset, we propose (*i*) multiple training of the models to find the optimal training set with good generalization capability on the test set (*ii*) to generate synthetic data, a method we adopt from other area of statistical learning and apply to a nuclear RP problem. The synthetic data has two main observable advantages: improvement of model performance and reduction of uncertainties caused by training data selection and limited size. The present work also quantifies the prediction errors due to the training data selection/size and uncertainties present in the input features. This demonstrate that with ML models, we can have the intervals where the true value of our predictions lies at a given confidence level.

As part of future works, it is planned to expand the range of inputs considered in this study to higher enrichments, burnups, and cooling times, by generating more training data with a computer code. Having said that, the results presented in this work demonstrate ML as a reliable tool to predict the integral decay heat of discharged LWR SNF assemblies. Furthermore, one of the methodologies introduced in this research (*i.e.*, application of synthetic data) has great potential for RP M&S and other research areas employing computationally expensive codes. Data-driven learning methods are currently not used extensively for most RP M&S applications such as UQ, SA and design optimization due to

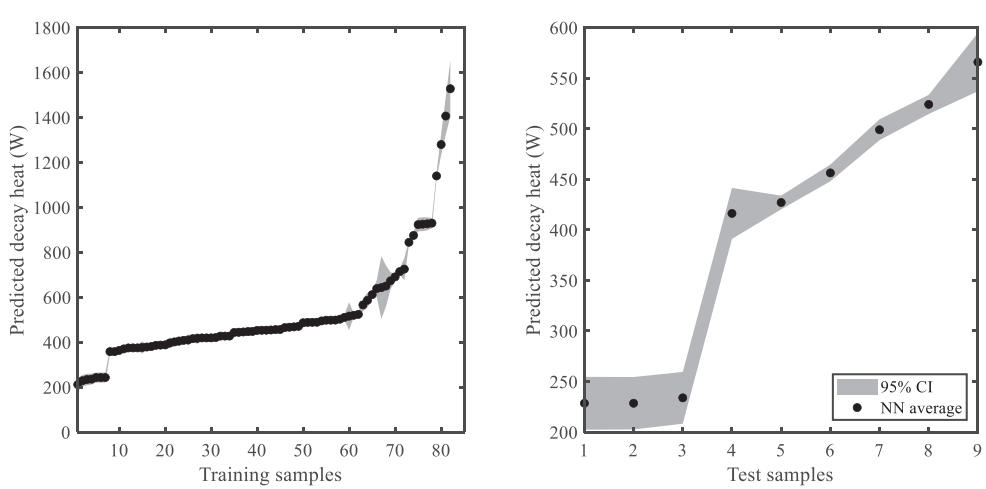

Fig. 13. NN predictions and uncertainties (synthetic PWR dataset model applied to the original dataset).

the huge computational cost of generating data for learning. The computationally expensive tools can be employed to generate small dataset of solutions for training ML models or to be used as snapshots for developing reduced order or surrogate models. One only needs to ensure that the range of input parameters of interest are covered in the small dataset generated. Afterwards, large size synthetic data can be generated to have similar statistical properties from the small dataset. This will not only save computation time; it will also remove the bottleneck caused by small datasets.

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.

Acknowledgements

This work was supported by Korea Institute of Energy Technology Evaluation and Planning(KETEP) grant funded by the Korea government (MOTIE) (20206510100040)

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.net.2021.05.037.

References

[1] I. Bose, R.K. Mahapatra, Business data mining - a machine learning perspective, Inf. Manag. 39 (2001) 211–225.
[2] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (1998) 2278–2324.
[3] W. Huang, Y. Nakamori, S.Y. Wang, Forecasting stock market movement direction with support vector machine, Comput. Oper. Res. 32 (2005) 2513–2522.
[4] J.R. Dorronsoro, F. Ginel, C. Sanchez, C.S. Cruz, Neural fraud detection in credit card operations, IEEE Trans. Neural Network. 8 (1997) 827–834.
[5] A. Hsu, B. Cheng, P.A. Bradley, Analysis of NIF scaling using physics informed machine learning, Phys. Plasmas 27 (2020), 012703.
[6] P. Sadowski, Deep Learning for Experimental Physics, (PhD Thesis), University of California, Irvine, 2016.
[7] T. Shaikhina, D. Lowe, S. Daga, D. Briggs, R. Higgins, N. Khovanova, Machine learning for predictive modelling based on small data in biomedical engineering, IFAC-Papers Online 48 (2015) 469–474.
[8] M. Sabatino, M. Fabiani, M. Božović, S. Garzoli, L. Antonini, M.E. Marcocci, A.T. Palamara, G. De Chiara, R. Ragno, R. Experimental data based machine learning classification models with predictive ability to select in vitro active antiviral and non-toxic essential oils, Molecules 25 (2020) 2452.
[9] Z. Ceylan, B. Sungur, Estimation of coal elemental composition from proximate analysis using machine learning techniques, Energy Sources, Part A Recovery, Util. Environ. Eff. 38 (2020) 1–11.
[10] J. Hoffmann, Y. Bar-Sinai, L.M. Lee, J. Andrejevic, S. Mishra, S.M. Rubinstein, C.H. Rycroft, Machine learning in a data-limited regime: augmenting experiments with synthetic data uncovers order in crumpled sheets, Sci. Adv. 5 (2019), eaau6792.
[11] A. Smith, A. Keane, J.A. Dumesic, G.W. Huber, V.M. Zavala, A machine learning framework for the analysis and prediction of catalytic activity from experimental data, Appl. Catal. B Environ. 263 (2020) 118257.
[12] P. Grechanuk, M.E. Rising, T.S. Palmer, Using machine learning methods to predict bias in nuclear criticality safety, J. of Comput. Theoretical Transport 47 (2018) 552–565.
[13] M.G. Fernandez, A. Tokuhiro, K. Welter, Q. Wu, Nuclear energy system's behavior and decision making using machine learning, Nucl. Eng. Des. 324 (2017) 27–34.
[14] J.J. Ortiz, A. Castillo, J.L. Montes, R. Perusquía, J.L. Hernández, Nuclear fuel lattice optimization using neural networks and a fuzzy logic system, Nucl. Sci. Eng. 162 (2009) 148–157.
[15] A. Pirouzmand, M.K. Dehdashti, Estimation of relative power distribution and power peaking factor in a VVER-1000 reactor core using artificial neural networks, Prog. Nucl. Energy 85 (2015) 17–27.
[16] K. Trontl, D. Pevec, T. Šmuc, Machine Learning of the reactor core loading pattern critical parameters, Sci. Tech. Nuc. Install. 2008 (2008) 695153.
[17] I.H. Bae, M.G. Na, Y.J. Lee, G.C. Park, Calculation of the power peaking factor in a nuclear reactor using support vector regression models, Ann. Nucl. Energy 35 (2008) 2200–2205.
[18] F. Khoshahval, S. Yum, H.C. Shin, J. Choe, P. Zhang, D. Lee, Smart sensing of the axial power and offset in NPPs using GMDH method, Ann. Nucl. Energy 121 (2018) 77–88.
[19] M.G. Park, H.C. Shin, Reactor power shape synthesis using group method of data handling, Ann. Nucl. Energy 72 (2014) 467–470.
[20] M.I. Radaideh, T. Kozlowski, Combining simulations and data with deep learning and uncertainty quantification for advanced energy modeling, Int. J. Energy Res. 43 (2019) 7866–7890.
[21] E. Nissan, An overview of AI methods for in-core fuel management: tools for the automatic design of nuclear reactor core configurations for fuel reload, (re)arranging new and partly spent fuel, Design 3 (2020) 37.
[22] I.C. Gauld, G. Ilas, B.D. Murphy, C.F. Weber, Validation of SCALE 5 Decay Heat Predictions for LWR Spent Nuclear Fuel, US Nuclear Regulatory Commission, Technical Report, 2010. NUREG/CR-6972.
[23] SKB, Measurements of Decay Heat in Spent Nuclear Fuel at Swedish Interim Storage Facilit, in: Y, CLAB. Svensk Kärnbränslehantering AB (SKB), Swedish Nuclear Fuel and Waste Management Co., 2006, pp. R-05–62.
[24] B.D. Murphy, I.C. Gauld, Spent Fuel Decay Heat Measurements Performed at the Swedish Central Interim Storage Facility, US Nuclear Regulatory Commission, Technical Report, 2010. NUREG/CR-6971.
[25] G. Ilas, I.C. Gauld, H. Liljenfeldt, Validation of ORIGEN for LWR used fuel decay heat analysis with SCALE, Nucl. Eng. Des. 273 (2014) 58–67.
[26] G. Ilas, I.C. Gauld, SCALE analysis of CLAB decay heat measurements for LWR spent fuel assemblies, Ann. Nucl. Energy 35 (2008) 37–48.
[27] P. Jansson, A. Håkansson, A. Bäcklin, S. Jacobsson, Gamma ray spectroscopy measurements of decay heat in spent nuclear fuel, Nucl. Sci. Eng. 141 (2002) 129–139.
[28] O. Osifo, S.J. Svärd, A. Håkansson, C. Willman, A. Bäcklin, T. Lundqvist, Verification and determination of the decay heat in spent PWR fuel by means of gamma scanning, Nucl. Sci. Eng. 160 (2008) 129–143.
[29] R.W. Mills, Decay heat calculations a study of their validation and accuracy, in: 2nd International Workshop on Nuclear Data Evaluation for Reactor (WONDER 2009), Cadarache, France, 2009, p. 152. September 29 – October 2.
[30] J.J. Faraway, Practical regression and anova using R, pp. 117, 118. Available at: https://cran.r-project.org/doc/contrib/Faraway-PRA.pdf. (Accessed 27 August 2020).
[31] G. Ilas, H. Liljenfeldt, Decay heat uncertainty for BWR used fuel due to modeling and nuclear data uncertainties, Nucl. Eng. Des. 319 (2017) 176–184.
[32] B. Ebiwonjumi, C. Kong, P. Zhang, C. Alexey, D. Lee, Uncertainty quantification of PWR Spent Fuel due to nuclear data and modeling parameters, Nucl. Eng. Tech. (2021), https://doi.org/10.1016/j.net.2020.07.012 article in press.
[33] B. Nowok, G. Raab, C. Dibben, Synthpop: bespoke creation of synthetic data in R, J. Stat. Software 74 (2016) 1–26.
[34] H. Ping, J. Stoyanovich, B. Howe, DataSynthesizer: privacy-preserving synthetic datasets, in: 29th International Conference on Scientific and Statistical Database Management, June 27-29, 2017. Chicago, IL, USA.
[35] D.A. Van Dyk, X.L. Meng, The art of data augmentation, J. Comput. Graph Stat. 10 (2012) 1–50.
[36] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, J. Artif. Intell. Res. 16 (2002) 321–357.
[37] T.R. Willemain, R.A. Bress, L.S. Halleck, Enhanced simulation inference using bootstraps of historical inputs, IIE Trans. 35 (2003) 851–862.
[38] C. Willman, A. Håkansson, O. Osifo, A. Bäcklin, S.J. Svärd, Nondestructive assay of spent nuclear fuel with gamma-ray spectroscopy, Ann. Nucl. Energy 33 (2006) 427–438.
[39] OECD/NEA, Evaluation Guide for the Evaluated Spent Nuclear Fuel Assay Database (SFCOMPO), Organization for Economic Cooperation and Development, Nuclear Energy Agency, Nuclear Science Committee, Working Party on Nuclear Criticality Safety, NEA/NSC/R, 2016, p. 8 (2015).
[40] C.M. Bishop, Training with noise is equivalent to Tikhonov regularization, Neural Comput. 7 (1995) 108–116.
[41] G. An, The effects of adding noise during backpropagation training on a generalization performance, Neural Comput. 8 (1996) 643–674.
[42] S.S. Lee, Noisy replication in skewed binary classification, Comput. Stat. Data Anal. 34 (2000) 165–191.
[43] J. Yang, X. Yu, Z.Q. Xie, J.P. Zhang, A novel virtual sample generation method based on Gaussian distribution, Knowl. Base Syst. 24 (2011) 740–748.
[44] H. Surendra, H.S. Mohan, A review of synthetic data generation methods for privacy preserving data publishing, Int. J. Sci. Tech. Res. 6 (2017) 95–101.
[45] S. Choi, C. Lee, D. Lee, Resonance treatment using pin-based pointwise energy slowing-down method, J. Comput. Phys. 330 (2017) 137–155.
[46] C. Lataniotis, D. Wicaksono, S. Marelli, B. Sudret, UQLab User Manual – Kriging (Gaussian Process Modeling), Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2019. Report # UQLab-V1.3-105.
[47] M. Moustapha, C. Lataniotis, S. Marelli, B. Sudret, UQLab User Manual – Support Vector Machines for Regression, Chair of Risk, Safety and Uncertainty Quantification, ETH Zurich, Switzerland, 2019. Report # UQLab-V1.3-111.
[48] M.H. Beale, M.T. Hagan, H.B. Denmuth, Deep Learning Toolbox User's Guide, The MathWorks, Inc., 2018.
[49] S. Marelli, B. Surety, UQLab: a framework for uncertainty quantification in MATLAB, in: 2nd International Conference on Vulnerability, Risk Analysis and Management, July 13-16, 2014. Liverpool, United Kingdom.
[50] B. Ebiwonjumi, D. Lee, Bayesian method and polynomial chaos expansion based inverse uncertainty quantification of spent fuel using decay heat measurements, Nucl. Eng. Des. 378 (2021) 11158.

[51] S. Vaccaro, S.J. Tobin, A. Favalli, B. Grogan, P. Jansson, H. Liljenfeldt, V. Mozin, J. Hu, P. Schwalbach, A. Sjöland, H. Trellue, D. Vo, PWR and BWR spent fuel assembly gamma spectra measurements, Nucl. Instrum. Methods Phys. Res. A 833 (2016) 208–225.
[52] T. Shaikhina, N. Khovanova, Handling limited datasets with neural networks in medical applications: a small-data approach, Artif. Intell. Med. 75 (2017) 51–63.
[53] P. Cunningham, J. Carney, S. Jacob, Stability problems with artificial neural networks and the ensemble solution, Artif. Intell. Med. 20 (2000) 217–225.
[54] S. Alfeo, H. Laurie, B. Vivek, J.S. Karen, Overcoming small data limitations in heart disease prediction by using surrogate data, SMU Data Science Review 1 (3) (2018), 12. Available at: https://scholar.smu.edu/datasciencereview/vol1/iss3/12. (Accessed 15 July 2020).

Abbreviations

LWR: Light Water Reactor
SNF: Spent Nuclear Fuel
PWR: Pressurized Water Reactor
BWR: Boiling Water Reactor
ML: Machine Learning
HEDL: Hanford Engineering Development Laboratory
GE-Morris: General Electric Morris Operation
GP: Gaussian Process
GPR: Gaussian Process Regression
SVM: Support Vector Machines
NN: Neural Network
ROM: Reduced Order Models
RP: Reactor Physics
M&S: Modeling and Simulation
GMDH: Group Method of Data Handling
SKB: Svensk Kärnbränslehantering AB
CLAB: Swedish Interim Storage Facility
VIF: Variance Inflation Factor
OECD: Organization for Economic Cooperation and Development
NEA: Nuclear Energy Agency
PDF: Probability Density Function
MAE: Mean Absolute Error
RMSE: Root Mean Square Error
LOOCV: Leave One Out Cross Validation
CV: Cross Validation
CI: Confidence Interval
UA: Uncertainty Analysis
UQ: Uncertainty Quantification
SA: Sensitivity Analysis