



Analytics-at-scale of Sensor Data for Digital Monitoring in Nuclear Plants

January 2022

3rd Annual Report

Cody Walker, Vivek Agarwal, Nancy Lybeck
Idaho National Laboratory

Pradeep Ramuhalli
Oak Ridge National Laboratory

Mike Taylor
Electric Power Research Institute



DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness, of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Analytics-at-scale of Sensor Data for Digital Monitoring of Nuclear Plants

3rd Annual Report

**Cody Walker, Vivek Agarwal, Nancy Lybeck
Idaho National Laboratory
Pradeep Ramuhalli
Oak Ridge National Laboratory
Mike Taylor
Electric Power Research Institute**

February 2021

**Idaho National Laboratory
Idaho Falls, Idaho 83415**

<http://www.inl.gov>

**Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517**

Page intentionally left blank

ABSTRACT

Nuclear power plants (NPPs) collect and store large volumes of heterogeneous data from various components and systems. With recent advances in machine learning (ML) techniques, these data can be leveraged to develop diagnostic and short-term forecasting models to better predict the future condition of equipment. Maintenance operations can then be planned in advance whenever degraded performance is predicted, resulting in fewer unplanned outages and the optimization of maintenance activities. This helps reduce maintenance costs and improve the overall economics of nuclear power.

This report focuses on Fiscal Year 2021 research progress and the development of a short-term forecasting process that leverages a feature selection process to distill large volumes of heterogeneous data and predict specific equipment parameters. A variety of feature selection methods, including Shapley Additive Explanations (SHAP) and variance inflation factor (VIF), were used to select the optimal features as inputs for three ML methods: long short-term memory (LSTM) networks, support vector regression (SVR), and random forest (RF). Each combination of model and input features was used to predict a pump bearing temperature both 1 and 24 hours in advance, based on actual plant system data. The optimal inputs for the LSTM and SVR were selected using the SHAP values, while the optimal input for the RF consisted solely of the response variable itself. Each model produced similar 1-hour-ahead predictions, with root mean square errors (RMSEs) of roughly 0.006. For the 24-hour-ahead predictions, differences could be seen between LSTM, SVR, and RF, as reflected by model performances (in terms of RMSE) of 0.036 ± 0.014 , 0.0026 ± 0 , and 0.063 ± 0.004 , respectively. As big data and continuous online monitoring become more widely available, the proposed feature selection process can be used for many applications beyond the prediction of process parameters within nuclear infrastructure.

This report summarizes the Fiscal Year 2021 research progress encompassing the (1) data cleaning and feature selection necessary for ML applications; (2) development of short-term forecasting models to predict future plant process parameters at both single and multiple time steps ahead; and (3) validation of the feature selection methods and short-term forecasting models, given new data from different systems.

Page intentionally left blank

CONTENTS

ABSTRACT.....	iii
ACRONYMS.....	viii
1. INTRODUCTION.....	1
2. SYSTEM AND DATA DESCRIPTION.....	2
2.1 Short-Term Forecasting.....	3
2.2 Data Preprocessing.....	4
2.2.1 Data Cleaning.....	4
2.2.2 Variance Inflation Factors.....	5
2.2.3 Shapley Additive Explanations.....	6
3. SHORT-TERM FORECASTING MODEL SELECTION.....	7
3.1 Long Short-Term Memory.....	7
3.2 Support Vector Regression.....	7
3.3 Random Forest.....	7
4. MULTI-STEP FORECASTING.....	8
4.1 Direct Multi-step Forecasting.....	8
4.2 Recursive Multi-step Forecasting.....	9
4.3 Direct-Recursive Hybrid Multi-Step Forecasting.....	9
5. RESULTS.....	10
5.1 Metrics Used and Model Parameters.....	10
5.2 Datasets Based on VIF and SHAP Analysis.....	11
5.3 Short-Term Forecasting Model Performance.....	12
5.4 Validation.....	16
5.4.1 BWR Condensate Pumps and Condensate Booster Pumps.....	16
5.4.2 PWR Steam Generator Feedpump.....	16
5.4.3 PWR Turbine.....	17
5.4.4 Validation Model Performance.....	17
6. SUMMARY AND PATH FORWARD.....	20
7. ACKNOWLEDGEMENTS.....	21
8. REFERENCES.....	21

FIGURES

Figure 1. Steps for leveraging digital monitoring to enable cost-effective predictive maintenance for nuclear power plants.....	3
---	---

Figure 2. Outliers being identified and rectified for the average feedwater flow over time. The data range has been normalized so that the maximum value of the variable is 1, and the normalized horizontal axis represents the fraction of plant operation time.....	5
Figure 3. Predictions of pump temperature 1 hour ahead, using the SHAP-determined input to an LSTM model. This includes a closer view of the pump temperature predictions in order to provide a clearer view of the 95% confidence interval.....	13
Figure 4. Predictions of pump temperature 1 day ahead, using the SHAP-determined input to an LSTM model. This includes a closer view of the pump temperature predictions in order to provide a clearer view of the 95% confidence interval.....	14
Figure 5. RF using SHAP inputs and making 1-hour-ahead predictions showed a higher RMSE, due to poor predictions between time steps 1000 and 2000, as well as for the time step at time 6,200.....	15
Figure 6. Comparison of direct vs. direct-recursive forecasting strategies.....	17
Figure 7. Forecasting the main turbine thrust bearing temperature using LSTM and SVR for both one- and 24-step-ahead predictions.....	19
Figure 8. Anonymized steam generator flow and time, showing a significant derate at 2009-11 and an outage at 2021-03.....	20

TABLES

Table 1. LSTM parameters.....	10
Table 2. RF parameters.....	10
Table 3. SVR parameters.....	11
Table 4. Comparison of model performance (RMSE * 1e3).....	11
Table 5. Mean SHAP values for determining feature importance.....	16
Table 6. 1-hour- and 1-day-ahead predictions for both LSTM and SVR.....	18

Page intentionally left blank

ACRONYMS

ANOVA	analysis of variance
BWR	boiling-water reactor
CBP	condensate booster pumps
CP	condensate pumps
FCS	feedwater and condensate system
MI	mutual information
ML	machine learning
NPP	nuclear power plant
PCA	principal component analysis
PWR	pressurized-water reactor
RF	random forest
V&V	validation and verification

Page intentionally left blank

Analytics-at-scale of Sensor Data for Digital Monitoring of Nuclear Plants

1. INTRODUCTION

Nuclear plant sites collect and store large volumes of data collected from various equipment and systems. These datasets typically include plant process parameters, maintenance records, technical logs, online monitoring data, and equipment failure data. The collection of such data affords an opportunity to leverage data-driven machine learning (ML) and artificial intelligence technologies to provide diagnostic and prognostic capabilities within the nuclear power industry. However, these datasets are potentially unstructured and collected at different temporal and spatial resolutions. Handheld (i.e., manual) measurements are collected either at periodic intervals or on an as-needed basis, while other datasets may be streamed and archived via plant computers. The recorded parameters for a specific piece of equipment may also vary from site to site, adding complexity to the data processing methodology. The unstructured nature of the data can be challenging for the development of scalable, reliable models for predicting future equipment parameters (the terms "process parameters" and "parameters" are used interchangeably hereinafter), without proper data cleaning and preprocessing.

For most ML techniques, the collected data are preprocessed and a set of features are selected. It is well known that input features significantly impact the model's prediction performance and training time [1]–[4]. When facing a large number of input features, the dimensionality may be reduced through feature extraction, feature selection, or a combination of both [5], [6].

Feature extraction techniques combine the original measurements in a manner that generates new features (from which a subset can be chosen to reduce the dimensionality), or can be used to extract useful information or features from the data [7]. One example of feature extraction, as found in wind turbine health monitoring, is the absolute difference in blade angle position [6]. Another example of feature extraction is principal component analysis (PCA), in which input features are combined to produce a new set of orthogonal features [8], [9]. Principal components (PCs) are linear combinations of the observed features, with the first PC extracting the maximum amount of information (i.e., variability) from the feature set. Subsequent PCs optimize the remaining information contained within the feature set under the constraint of being orthogonal (i.e., uncorrelated) to the preceding PCs [9]. Because the PCs are the eigenvectors of the associated covariance matrix, the eigenvalues are therefore related to the amount of information contained within each PC. Feature extraction, and thus dimensionality reduction, is then performed by removing those PCs associated with the least amount of information.

Feature selection is the process of choosing the best combination of features from the original input feature space. Feature selection methods are primarily divided into two categories: filters and wrappers [10]. Filters are open-loop methods that measure feature characteristics (e.g., information, dependency, consistency, and distance) while being fast and scalable [5], [10], [11]. Those features calculated as having the best characteristics are then chosen as inputs for the model. Wrappers combine the feature selection process with a learning algorithm so that the selection process is based on model performance. This allows consideration of the dependence between variables [12], [13]. However, wrappers are prone to overfitting and can be computationally expensive [10]. Hybrid and ensemble methods integrate filters and wrappers alike, thereby benefiting from their complementary approaches [10], [14].

Three of the most common feature selection techniques—apart from the ones used in this report—are mutual information (MI) [15], recursive feature elimination [16], and analysis of variance (ANOVA) tests. MI uses entropy as a means of determining the amount of information gained by each input feature. [15] used MI as the basis for a minimally redundant, maximally relevant feature selection method for multi-class support vector machine classification of railcar conditions. Recursive feature elimination recursively trains a model, calculates a cross-validation score, and then removes the least important feature, as determined via the internal feature ranking [16]. Feature importance ranking is common to methods such as random forest (RF). The cross-validation score is used to determine at what point enough

features have been selected to adequately describe the system. To estimate the wear on a gear transmission system, [16] used recursive feature elimination to reduce a set of 5,650 features down to the top 15. ANOVA is a statistical method that uses hypothesis testing to determine whether a given result or feature is significant [17]. [18] used ANOVA to determine whether various wind turbine bearing temperature measurements significantly differed from each other. If the variables do not significantly differ, one of them may be removed from the analysis. Other methods for feature extraction utilize deep learning to skip the feature extraction step altogether. [19] used an end-to-end architecture that fully automated the feature extraction process for diagnosing COVID-19 from x-ray images.

Though many approaches to feature selection are described in the literature, there is still a need to objectively assess short-term forecasting models—especially those using ML—based on data from operating plants. This project closed this gap, along with several others thus far.

In the first year of the research, the project team developed a general methodology for the technoeconomic analysis of wireless sensor modalities for use in monitoring equipment condition [20], [21].

In the second year of the research, the project team focused on developing and evaluating integrated algorithms for diagnostic and prognostic estimation of equipment condition, in collaboration with Exelon Generating Station [22], [23]. The feedwater and condensate system (FCS) was used as the identified balance-of-plant system. Data associated with the system were provided to Idaho National Laboratory and Oak Ridge National Laboratory for analysis. The Electric Power Research Institute provided technical guidance and direction on the project, as well as essential reference documents related to fault conditions observed in the FCS.

In the third year of the research, the project team developed short-term forecasting capabilities by comparing three different ML techniques for predicting a nuclear power plant (NPP)'s FCS parameters, and how various input features affect the ML model's performance. The team then formalized the preprocessing steps required to integrate heterogeneous nuclear plant data. These preprocessing steps include both feature selection and the necessary data cleaning. Finally, validation of the developed approaches and algorithms were completed using new data from different systems [24].

The rest of this report is arranged as follows: Section 2 gives the background for the report, descriptions of each of the selected models, and the data preprocessing needs; Section 3 presents the selection of the short-term forecasting models; Section 4 details the methodology for multi-step forecasting; Section 5 details the model hyperparameters generated for this research, compares the performance of each model (as the input features vary), and covers the validation results; and Section 6 concludes by summarizing the progress and highlighting its significance.

2. SYSTEM AND DATA DESCRIPTION

Data in this analysis were primarily recorded from the condensate pumps (CPs) and condensate booster pumps (CBPs) found within the FCS. The primary purpose of the FCS is to condense steam and collect the drainage in the main condenser before purifying, preheating, and pumping the water back to the reactor vessel [25]. The CPs provide the driving force for pushing the condensate through auxiliary systems such as the steam jet air ejectors condenser, steam packing exhaust condenser, off-gas condenser, and demineralizers—all of which work to condition the condensate. Afterwards, the CBPs are the driving force of the flow as the condensate travels through a string of low-pressure heaters that work to preheat the water to the required temperature. In the boiling-water reactor (BWR) system of interest, the CPs and CBPs are driven by a shared motor.

The available sensor data were recorded for a 5-year period and include variables such as:

- generator gross load (MW)
- average feedwater flow rates (million gallons/second)
- temperatures from the feedwater pumps, CPs, CBPs, and associated motors (°C)

- pressures within the condenser, CPs, CBPs, and turbines (psig)
- current to the CP and CBP drive motors (amps).

The recorded data were primarily collected from the FCS, but some temperatures, pressures, and flow rates came from other components and subsystems such as the reactor or turbine system. Each dataset consists of unlabeled data and is sampled hourly. There was no indication as to whether any portion of the data corresponded with equipment failure. Furthermore, the data did not give sufficient information to determine the cause of each derate. Data preprocessing was necessary, as the data contained outliers, missing values, and several temperature signals that experience a clear seasonal trend: colder in winter, warmer in summer.

2.1 Short-Term Forecasting

From data collection to decision making, the generalized steps for producing short-term forecasting models are as follows:

- Collect, clean, and explore the data.
- Determine the relevant features.
- Train and evaluate the models.
- Visualize the results for more informed decision making.

A more detailed approach that leverages digital monitoring capabilities to create useful diagnostic, prognostic, and short-term forecasting models for existing NPPs can be seen in Figure 1. This research is part of the broader vision detailed in Figure 1. The overall project focuses on addressing digital monitoring challenges that range from deployment of wireless sensor technology in the NPP environment, to feature selection and data analytics that drive online component monitoring, to visualization for decision-making purposes. Each task is vital for ensuring that the right people get the right information at the right time.

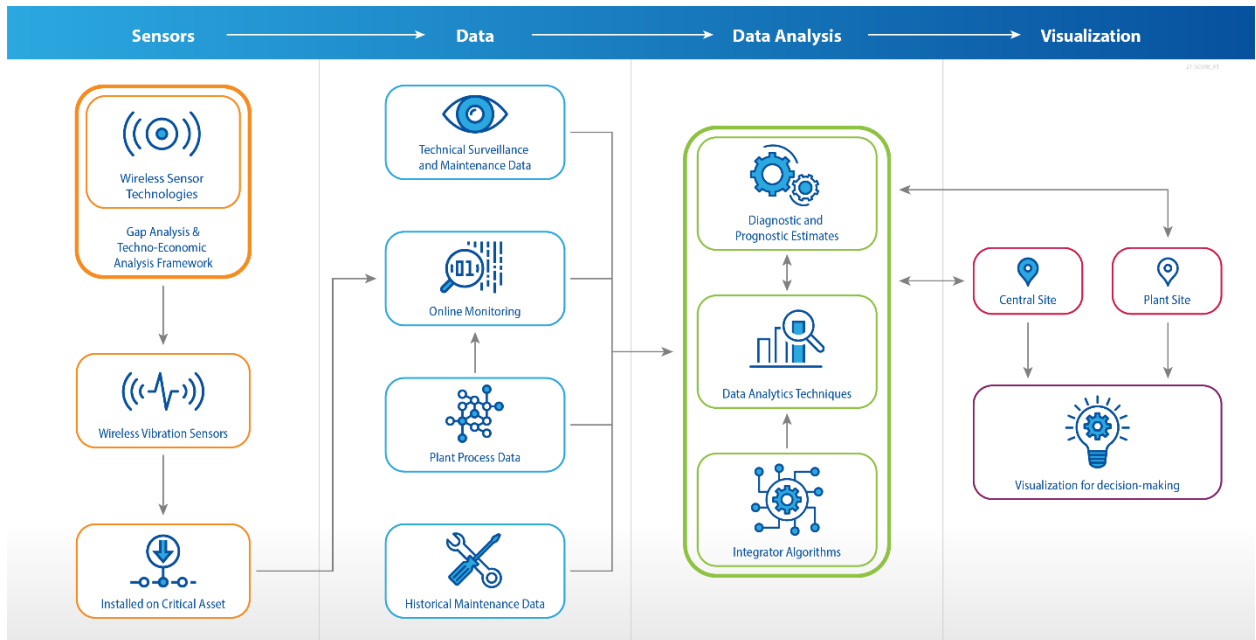


Figure 1. Steps for leveraging digital monitoring to enable cost-effective predictive maintenance for nuclear power plants.

This research focuses on the data processing and data analytics aspects of determining relevant features for short-term forecasting models. Inputting irrelevant features into ML models not only increases training times as a result of the extra input dimensions, but can actively hurt model performance. For coverage of other aspects related to prognostics and forecasting, [7] is an excellent review paper covering critical component analysis and sensor selection, along with prognostic methodologies and tool evaluation.

2.2 Data Preprocessing

The data in this research were taken from a BWR's FCS. Heterogeneous signals across different systems and components (e.g., reactor power, turbine pressures, bearing temperatures, and reactor feedwater pump flows) were provided to supplement these data. The data gathered from the FCS corresponded to a 5-year time frame, which covered periods of steady-state operation, derates, trips, and refueling. Steady-state operation is broadly defined as all instances of the reactor operating at above 90% nominal power; however, fluctuations in power can still be seen in this category. Derates contain all observations made when the reactor is operating at 5–90% of nominal power. Derates are instances of reduced power operations and may be caused by environmental, operational, or reliability issues. Trips, also known as scrams, are emergency shutdowns of the reactor. The refueling time period covers the initial ramp-down of nominal power, the refueling outage, and the subsequent ramp up to steady-state conditions. This research focused on predicting CP temperatures within the broadly defined steady-state conditions.

Large groups of raw data signals are rarely the optimal choice of model inputs. Raw data should be processed, cleaned, and pruned to improve model performance. Raw data can be subject to missing information, outliers, sensor and process noise, different data scales, etc. Many of these issues can be mitigated via data preprocessing (e.g., data cleaning, feature scaling, and feature selection), thereby reducing their effects on model performance. The data preprocessing steps implemented in this research are described in the subsections below.

2.2.1 Data Cleaning

The data were cleaned, processed, and pruned before being analyzed in the short-term forecasting models. The data cleaning focused on addressing any missing data, potential outliers, data selection, and scaling. Missing data were primarily noted when the system or component was offline, in which case the data were left as missing. However, if the component was online and the missing data were due to a sensor or data archival error, the missing values were interpolated using neighboring values. Daylight savings time can also be a minor inconvenience, as it entails the skipping or duplication of time steps, depending on the time of year. Skipped time steps, in this instance, were assumed to share the same value as that of the previous time step.

Many of the datasets within the steady-state portions were heavily skewed in one direction or another. Potential outliers within these datasets were flagged as a result of being four standard deviations away from the mean. For example, the feedwater flow seen in Figure 2 has several potential outliers marked. The axes in this figure, as well as the others in this report, have been anonymized to protect the plant's identity. The potential outliers were replaced using a median filter applied via a sliding window approach. The data were recorded hourly, so a window of 51 points (slightly over 2 days worth of data) was empirically selected as the median filter's width. Temperature data can often contain sensor noise, seasonal variations, or long trains of outliers. A median filter of 700 (roughly 21 days) was used to account for seasonal variations and long trains of outliers. Additional data cleaning for all equipment parameters included standardizing the measurements to zero mean and unit standard deviation to account for the different scales seen within the data. Standardization of data is generally considered a best practice before using most ML models. After data cleaning, selecting the relevant parameters becomes important.

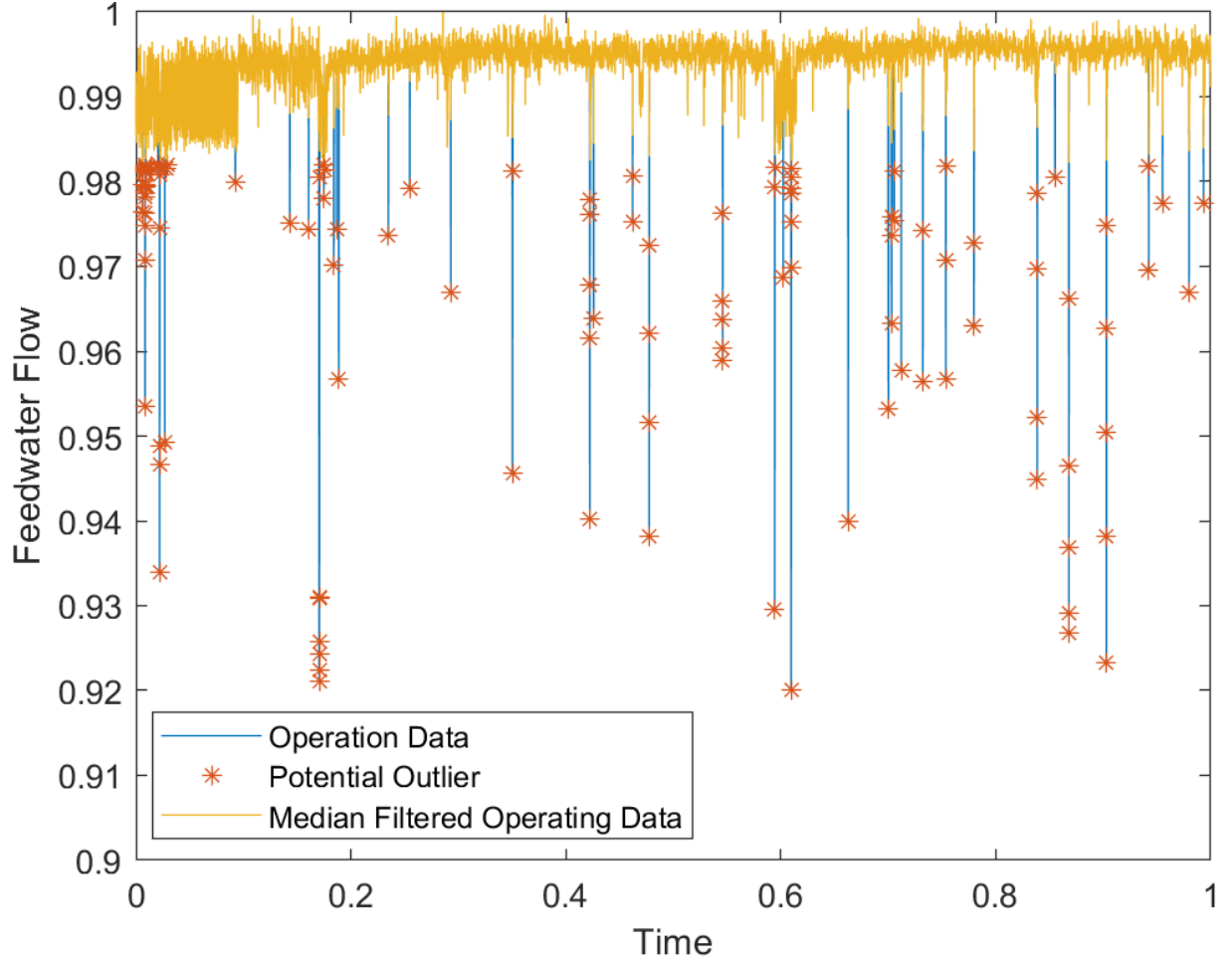


Figure 2. Outliers being identified and rectified for the average feedwater flow over time. The data range has been normalized so that the maximum value of the variable is 1, and the normalized horizontal axis represents the fraction of plant operation time.

The two primary feature selection techniques covered in this report are variance inflation factor (VIF) and Shapley Additive Explanations (SHAP). VIF is a filter-based technique for reducing multicollinearity. SHAP is a wrapper-based technique for determining the contribution, and thus the importance, of each input feature. Both these techniques are detailed below.

2.2.2 Variance Inflation Factors

Multicollinearity occurs when two or more predictor variables are highly correlated [26]. This can lead to unforeseen variability in regression analyses, as the strong relationship between the independent variables distorts the relationship with the dependent variable. Multicollinearity can be rectified in several ways, such as by removing one or more of the highly correlated variables, either through PCA [9], [27] or a regularization technique (e.g., ridge regression) [28]. PCA was initially tested with long short-term memory (LSTM) neural networks, but better results were obtained by simply removing the variables that corresponded to large VIFs.

VIFs measure the amount of multicollinearity between predictor variables, and can carry a value of one or greater. A VIF value of one represents no correlation, whereas values of either five or 10 are commonly used to indicate highly correlated variables [26], [27]. In this research, a VIF of five was used to identify highly correlated variables that are candidates for elimination. VIF can be calculated in two steps. The first step is to withhold one of the predictor variables, then employ an ordinary least square

regression to predict that variable by using the remaining variables, as per Equation 1:

where β_0 is a constant, ϵ is an error term, x_j is the variable to check for multicollinearity, and x_k are the other predictor variables. The second step in calculating the VIF is reflected in Equation 2:

where R^2 is the coefficient of determination from Equation 2. This two-step process can be completed for each predictor variable to determine its VIF. Variables were then removed if their VIF was five or greater, thus eliminating multicollinearity from the input space. The one exception to this rule was the pump temperature being estimated. Because temperatures are relatively slow-moving parameters, one would expect that the best predictor of temperature for time $t+1$ would be the temperature at time t . Therefore, the pump temperature remained in the input space, while other parameters were removed to reduce the multicollinearity in the input space.

VIF has been used to check for and remove multicollinearity in a variety of different situations. [29] used VIF to check for multicollinearity between the input variables for a multivariate logistic regression in order to determine the factors associated with the death outcome in patients suffering from severe reactions to COVID-19. No multicollinearity was seen, as each variable had a VIF of less than two. [30] used a multivariate logistic regression to determine the prognostic nutritional index's impact on postoperative pulmonary complications. They used a VIF threshold of 10 for determining highly correlated variables, then eliminated them from the analysis. [1] used VIF to remove multicollinear variables before estimating the state of health for lithium-ion batteries, using a gradient boosted decision tree. In their analysis, the VIF empirical cutoff was set at 10 for strong multicollinearity, and 100 as the threshold for removal. Each of these examples had differing cutoffs for determining whether the predictor variable space exhibited enough multicollinearity to warrant action. However, [31] showed that rules of thumb for VIF and its tolerance should be put into context with other variables (e.g., sample size and variance of the independent variable) that might affect the variance of the i^{th} regression coefficient in Equation 1. Even though the dataset in this report satisfies these rules of thumb, a threshold of five was selected so as to be conservative.

2.2.3 Shapley Additive Explanations

SHAP values are based on a game-theoretic concept that considers each input feature as a "player" on a "team" of features that work together to influence the model's overall output [32]. A baseline model output is first determined by averaging over all the predictions for a given model. Each specific model prediction is then considered as a function of input features which deviates the prediction from the baseline model output. The feature's influence on each prediction is taken into consideration based on different combinations of input features [33]. SHAP values use an additive feature attribution approach, meaning that the output is a linear combination of the input variables [33]. In this manner, SHAP empirically determines the influence each feature has on the prediction output [32]. Computing the exact solution for the SHAP values is, by nature, an exponential problem, typically leaning it toward being infeasible [34]. However, a SHAP approximation can be made using an explanation model. The original model is associated with the explanation model with simplified inputs, and is expressible as:

where M is the number of input features, ϕ represents the baseline model output, and ϕ_i represents the SHAP values. The SHAP values themselves can be approximated through a variety of methods, including Kernel SHAP, Deep SHAP, and Tree SHAP. A more detailed explanation of the SHAP value formulation and interpretation can be found in [35].

SHAP values have primarily been used to try to explain how a specific feature affects the model's output. For example, SHAP values were used in conjunction with the Cox hazards model to identify the most

important features that increase your odds of death over the next 20 years (e.g., age, sex, systolic blood pressure, and poverty index) [36], and were also used to identify the most important features in predicting shear wall failure modes [33]. [37] used SHAP values as a feature selection method before implementing ML methods (e.g., extreme gradient boosting and support vector regression [SVR]) to predict the total energy consumption of electric vehicles under realistic conditions, using parameters such as trip distance, tire type, power, and air conditioning. [33] predicted failure modes of reinforced concrete walls by employing RF ML models, then used SHAP values to rank the most influential input variables. ML is often criticized as a black-box approach, but SHAP values can lead to more interpretable models. For example, [38] used SHAP values to analyze feature contribution after prognosing the remaining useful life of turbofan engines. Our use of SHAP values, however, is to improve model feature selection by eliminating unimportant signals before they are inputted to the final model. Irrelevant or redundant inputs increase both the dimensionality of the data and the computational cost of finding the global minimum [39].

3. SHORT-TERM FORECASTING MODEL SELECTION

This report focuses on applying feature selection to data-driven approaches for short-term forecasting of plant parameters. Data-driven methods are excellent because every component interaction need not be modeled to produce usable results. These methods are generally inexpensive to create, and are quicker to develop than their physics-based counterparts [40]. The primary limitation of data-driven methods is the data themselves. The data must be plentiful and cover the entire expected range of operations. Extrapolation that leads to non-physical results can occur for predictions outside the range of the training dataset [40]. Data may not exist for all desired conditions. For example, new or mission-critical systems may not have the run-to-failure or operational data required to produce adequate forecasting models for certain operating conditions. Even with these shortcomings, data-driven models are widely used and implemented. This research focuses on three particular models for the short-term forecasting of plant parameters: LSTM neural networks, SVR, and RF. These models were selected due to their time series forecasting capabilities and ease of implementation within the Python coding environment.

3.1 Long Short-Term Memory

LSTM is a type of recurrent neural network that incorporates unique memory cells to learn long-term relationships between the inputs and outputs [4]. As with classical recurrent neural networks, LSTM networks process temporal information to develop the relationship between previous inputs and the current output. However, LSTM networks have a hidden state that serves as memory and interacts with the current output. This stored, hidden state is updated as new inputs arrive. A forget gate is also common in LSTM architectures as a means of forgetting some of the previous memory cell states, thus helping boost performance. The output of the forget gate is used to inform and update the hidden state of the LSTM. Many different types of activation functions can be used within the LSTM layers. Number of hidden layers, batch size, and number of epochs are examples of hyperparameters that must be optimized before implementing the fitted model into the system. In this report, a grid search approach was used to optimize the LSTM hyperparameters, but random and Bayesian search methods could also have been used [41], [42]. A more detailed guide to constructing LSTM networks is found in [39].

3.2 Support Vector Regression

SVR is a kernel-based regression technique featuring two primary components: a kernel function and an optimization routine. The kernel function first transforms the data into a higher dimensional feature space. The optimization routine then tries to minimize the generalization error. The solution then depends only on a subset of the training data (i.e., support vectors) that lies along the separation boundary [43]. In this report, a radial basis function kernel was implemented.

3.3 Random Forest

RF is a decision tree ensemble usable for either classification or regression [44]. The output is chosen based on a majority vote from the group of decision trees comprising the RF. Because a lone decision tree

is subject to high variance and noise, the RF addresses this by generating multiple trees, using bootstrapped samples from the training data [45]. Overall, RF is a straightforward, easy-to-implement ML model with relatively few hyperparameters to optimize (e.g., total number of trees to generate, minimum number of samples to split, and split criteria). The RF models generated in this research used an ensemble of 100 decision trees, each with at least two samples to split. The split criteria was based on the weighted impurity decrease equation given in Equation 4:

where n is the number of samples at the current node, N is the total number of samples, I is the impurity, n_r is the number of samples in the right child, I_r is the right impurity, n_l is the number of samples in the left child, and I_l is the left impurity [5]. This weighted impurity calculates whether or not the subsequent split would be beneficial.

4. MULTI-STEP FORECASTING

The simplest forecast is one-step ahead, since it represents the least amount of extrapolation from the known data. Intuitively, we expect better results, as there are fewer opportunities for unseen transients, depending on the size of the step. The 24-hour-ahead predictions in the previous report [22] were created using one-step-ahead predictions. This was accomplished by resampling the data so that each time step was 24 hours apart rather than the original 1-hour frequency. This method yielded acceptable results for that dataset, primarily due to the amount of data available: 5 years worth. However, the other two validation and verification (V&V) datasets discussed in this report only contain a year worth of data. By resampling the data from an hourly to a daily frequency, and then further dividing the data into training, testing, and validation sets, the V&V dataset becomes sparse for learning trends. This resampling strategy is therefore inappropriate for these datasets. With limited data, each point is critical for training the model. Several methods for making multi-step-ahead time-series predictions were tested to determine the optimal strategy for forecasting. These prediction strategies include direct multi-step forecasting, recursive multi-step forecasting, and direct-recursive hybrid (DirRec) multi-step forecasting [10]. These methods are discussed in further detail in the subsections below.

4.1 Direct Multi-step Forecasting

The first method for making predictions that are multiple time steps ahead is the direct multi-step forecasting method. In this method, the model is trained to directly predict component measurements at a set time into the future. This method is similar in appearance to a one-step-ahead prediction, but with an increased prediction horizon. Equation 5 shows an example of a one-step-ahead model:

where the previously observed data ($x_t, x_{t-1}, \dots, x_{t-n}$) is used to make a prediction that is a single time step ahead, x_{t+1} , and n is the number of previously observed time steps to be inputted into the model. In direct multi-step forecasting, this prediction horizon of one time step is extended to the desired length. Equation 6 shows an example of a model predicting two steps ahead with the same data previously observed in Equation 5. The difference between Equations 5 and 6 is the model's predicted outcome (x_{t+1} vs. x_{t+2}), due to the extended prediction horizon.

This prediction horizon can be extended even further. In this report, hourly recorded data were used to predict component measurements twenty-four time steps (i.e., one day) ahead. However, there are limitations to this technique. Extending the prediction horizon will, in most cases, diminish the accuracy of the predictions. This type of model has no opportunity to learn the dependencies between forecasted predictions. Changes in one variable may directly affect another, since this type of model might overlook this type of interaction, as it directly predicts future values.

Similarly, a direct, multi-output, multi-step forecast casting strategy can be implemented in which one model attempts to predict each consecutive future step, as seen in Equation 7:

This type of model will have similar or exacerbated drawbacks in comparison with direct multi-step forecasting, due to the additional predicted outputs. This will likely lead to a more complex model that requires more time to train, along with more data to prevent overfitting. Therefore, to recap, model 1 (Equation 5) attempts to predict one step ahead, model 2 (Equation 6) attempts to predict two steps ahead using the same inputs as model 1, and model 3 tries to predict both one and two steps ahead while still using the same input parameters.

4.2 Recursive Multi-step Forecasting

Another method for making multi-step-ahead predictions is recursive multi-step forecasting, which entails a single model that makes one-step-ahead predictions. After a prediction is made, the input window is shifted by one step, and the model's own prediction is used as one of the inputs. This process of recursively using the model's output as an input is repeated until the desired prediction horizon is reached. Equations 8 and 9 show a single step in the recursive multi-step forecasting process. In Equation 8, model 1 makes a single one-step-ahead prediction, $x_{(t+1)}$. This prediction is then fed back into itself, as seen in Equation 9, with model 1 then being used to make another one-step-ahead prediction, $x_{(t+2)}$.

This model has its disadvantages—namely, the accumulation of prediction errors. There will inevitably be some prediction error between the predicted output of Equation 8 and the ground truth. This error is then inserted back into the model in Equation 9, where the error can increase the deviations from the ground truth, thus resulting in accumulated prediction errors. Longer prediction horizons lead to greater accumulation of errors.

4.3 Direct-Recursive Hybrid Multi-Step Forecasting

DirRec multi-step forecasting combines direct and recursive forms of prediction in order to mitigate the disadvantages of each. In DirRec multi-step forecasting, several models are trained, each with a minimal prediction horizon. However, the prediction from one model is then fed as an input to another model that makes the same length prediction. In this way, a series of models are used in succession to extend the prediction horizon. The first model, seen in Equation 10, is used to predict a single step ahead, $x_{(t+1)}$. This prediction is then used as an input for model 4, seen in Equation 11, which is used to make another one-step-ahead prediction, $x_{(t+2)}$.

This approach aims to minimize the disadvantages of both the direct and recursive approaches. With a series of models, dependencies between predicted outcomes can be observed, thus reducing the disadvantage of direct approaches. With multiple models being created, there is also an opportunity to correct some of the accumulating errors that one specific model might make, thus reducing the drawbacks of the recursive model. However, numerous models can be computationally expensive to train. With each model making one-step (i.e., one hour)-ahead predictions, 24 models would need to be trained to predict a single day in advance.

In this report, the recursive multi-step forecasting method produced prohibitively large accumulations of prediction errors. And though the DirRec multi-step forecasting method avoided such error accumulation, training multiple models became computationally expensive. A sample case from each forecasting method is shown below to demonstrate that method's forecasting ability. In this report, the direct multi-step forecasting method was chosen as the primary method for forecasting, due to its relatively low degree of prediction error and computational expense.

5. RESULTS

As mentioned earlier, the NPP data used for the analysis in this section originate from a BWR system and have been anonymized to protect the privacy of the plant. As a result, all data shown remain in standardized form, with zero mean and unit variance, and have been shifted slightly.

5.1 Metrics Used and Model Parameters

The model outputs were compared using the root mean square error (RMSE), as calculated via Equation 12:

where N is the total number of predictions, \hat{y}_i is the model's predicted output, and y_i is the observed output.

The hyperparameters for LSTM are important components that help determine the quality of the model's predictions and thus the usefulness of the end product. However, this report primarily focuses on how the choice of input features affects model performance. Optimal LSTM hyperparameters were identified by using a grid search method for one set of input variables and prediction horizons from this dataset, then the best results for the hyperparameters were used for all the LSTMs networks thereafter. The chosen hyperparameters are given in Table 1. The model was trained for a maximum of 250 epochs, or until the validation loss did not improve for 10 consecutive epochs. Early stoppage of training was implemented to avoid overfitting the model to the training data. The dropout layer and L1 & L2 regularizers were all added to improve the LSTM network's robustness by reducing overfitting. The hyperparameters for the RF model can be seen in Table 2. The hyperparameters for the SVR, found through a grid search, are shown in Table 3.

Table 1. LSTM parameters.

Hyperparameter	Value
Number of LSTM units	1000
Number of layers	4
Batch size	64
Epochs	Up to 250
Dropout	20%
Validation split	10%
Optimizer	Adam
Activation function	ReLu
Learning rate	0.0001
Loss function	Mean square error
L1 & L2 regularizer	1e-5

Table 2. RF parameters.

Hyperparameter	Value
Number of trees	1000
Measure of split quality	Squared error
Bootstrapping	True

Table 3. SVR parameters.

Hyperparameter	Value
C	46.416
ϵ	0.044
γ	0.464

5.2 Datasets Based on VIF and SHAP Analysis

The target variable to predict in this analysis was pump bearing temperature, using different combinations of the 78 recorded variables (e.g., reactor power, feedwater flows, temperatures, pressures, and other parameters recorded throughout the plant). The parameter values from associated systems such as the turbine control systems are included in the 78 recorded variables. The possible input features were the recorded variables post-preprocessing. These variables were grouped and used as predictors, as per the criteria listed in Table 4. No further feature engineering occurred, as this report's primary objective was to identify the best combination of features from a select set of features. The first feature set used all 78 variables, regardless of their relationship to the pump's bearing temperature. The second feature set started with all 78 variables, then removed those with a VIF of five or more to eliminate the multicollinearity within this feature set. By removing the multicollinearity from the set by using VIF as the basis for feature selection, the total number of features dropped from 78 to 36. Note that the current pump bearing temperature was never removed from any of the feature sets, as it is the single best predictor of future pump bearing temperatures. The next feature set consisted solely of the variables that had a correlation of 0.9 or greater with the predicted pump bearing temperature. More specifically, these variables were the current pump bearing temperature as well as three bearing temperatures from other pumps. All these variables highly correlated with each another. After removing multicollinearity from this feature set, the only remaining variable was the current bearing temperature itself. Used on its own, the current pump bearing temperature can be seen as a control group for determining whether the information added by the other variables actively helps or hurts model performance.

Table 4. Comparison of model performance (RMSE * 1e3).

		1-hour ahead			24-hour ahead		
Selection Method	# Features	LSTM	SVR	RF	LSTM	SVR	RF
All	78	318 \pm 258	34	92 \pm 4	326 \pm 26	44	132 \pm 9
All with VIF reduction	36	206 \pm 18	18	83 \pm 4	365 \pm 19	32	85 \pm 6
Corr > 0.9	4	8 \pm 4	7	114 \pm 10	37 \pm 4	39	238 \pm 14
Feature by itself	1	7 \pm 4	8	6 \pm 0.2	36 \pm 2	27	51 \pm 1
Corr > 0.8	14	29 \pm 7	11	95 \pm 13	160 \pm 9	28	130 \pm 11
Corr > 0.8 with VIF	6	6 \pm 1	9	9 \pm 1	52 \pm 5	27	60 \pm 4
Location-based	11	15 \pm 3	7	61 \pm 10	117 \pm 15	33	153 \pm 19
Location-based with VIF	8	14 \pm 5	13	61 \pm 10	124 \pm 9	46	97 \pm 14
SHAP values	2	6 \pm 2	6.6	35 \pm 2	36 \pm 14	26	63 \pm 4

The next feature set contained variables that had a correlation of 0.8 or greater with the pump bearing temperature. These 14 variables included turbine exhaust temperatures, pump bearing temperatures, and pump motor temperatures. Using the VIF values to remove any multicollinearity from this feature set reduced the number of variables from 14 to six, while still retaining the same variety of features (i.e., redundant turbine exhaust and bearing temperatures were removed from the feature set). One feature set was location-based (i.e., variables within the same loop were grouped together) and contained variables within the same loop as the predicted variable, including feedwater temperatures/pressures, pump bearing and motor temperatures, and condenser pressure. Using the VIF values, the location-based feature set was reduced from 11 variables to eight.

The final feature set was based on the SHAP values calculated for every variable in order to predict the temperature of a specific bearing within pump 1. Because the mean SHAP value represents the average impact a variable has on the model output, it was utilized to define feature importance, as the most important features are those that most greatly affect the model's outcome. The four most important variables, according to the magnitude of the mean SHAP values, were all pump bearing temperatures, as shown in Table 5. Pump 1 had multiple bearings, each with recorded temperatures, so the target variable being predicted was labeled pump 1a. Pump 1b related to the temperature of a separate bearing on the same pump as pump 1a. Pump 1a's temperature had the greatest impact on the prediction outcome. Although the temperature measurement site for pump 1b was physically close to pump 1a, it contained little in the way of useful new information, thus leading a low mean SHAP value.

5.3 Short-Term Forecasting Model Performance

Each of the feature sets was used as an input to make one- and 24-hour-ahead predictions using LSTM, SVR, and RF. Since LSTM and RF results can vary based on the seed being used to train the weights or bootstrap, each model was then retrained 10 times in order to compute the average RMSE and its standard deviation. The mean RMSE and its standard deviation, each multiplied by 1,000 for easier viewing, can be seen in Table 4. A lower mean RMSE represents better prediction accuracy, while a lower standard deviation represents more consistent model results. Based on the mean RMSE for the one-hour-ahead predictions, Table 4 shows the three top models to be RF using one variable, LSTM using the correlation >0.8 with VIF reduction, and LSTM using SHAP-determined inputs.

LSTM, the structure of which is described in Table 1, seemingly benefits from selectively choosing which variables to pair with the response variable in the input space. Adding too many unrelated variables diminished the model performance. SVR does not seem to suffer as greatly from this limitation, since reasonable results were produced when using all variables. SVR also had results comparable to those of LSTM and RF when using only the current value of the response variable as an input. SVR outperformed LSTM and RF when using eight or more variables. RF had the most sporadic results of the three models. The most accurate and consistent model resulted from using only the response variable. Inclusion of additional, very highly correlated variables significantly increased the mean RMSE, as demonstrated by the feature sets with correlations of 0.8 or higher. In general, the reduced standard deviation for the RMSE indicates that utilizing VIF values to remove multicollinearity invariably improves model consistency.

The one-hour-ahead predictions made by LSTM using SHAP-determined inputs can be seen in Figure 1. We again emphasize that the axes have been anonymized to protect the plant's privacy. The test data cover temperatures over a 10-month period. The average temperature is seen to decrease before slowly increasing again. This is a seasonal effect caused by the temperature of the local water source used as the plant's heat sink. This seasonal effect can be seen in the bearing temperature, which ran cooler in the winter months and hotter in the summer months. The zoomed-in portion of the inset graph in Figure 3 better captures the model estimate and its 95% confidence interval. The LSTM captures the pump temperature trends very well, as seen by the low mean RMSE, due to the actual measurement being within the 95% confidence intervals.

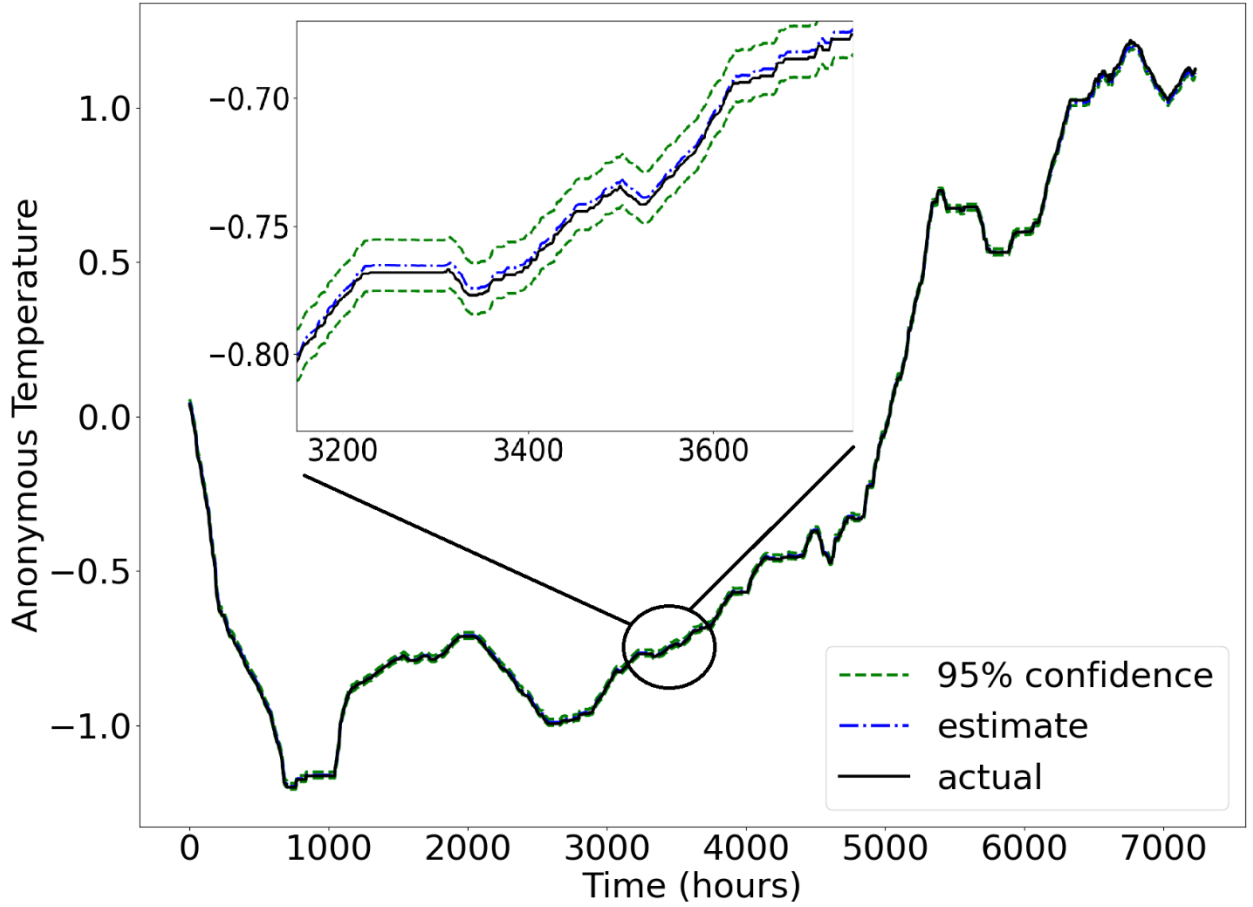


Figure 3. Predictions of pump temperature 1 hour ahead, using the SHAP-determined input to an LSTM model. This includes a closer view of the pump temperature predictions in order to provide a clearer view of the 95% confidence interval.

Each of the feature sets in Table 4 was also used as an input to predict the pump bearing temperature 24 hours ahead, using LSTM, SVR, and RF. Making a multi-step-ahead prediction can be more complicated than a one-step-ahead prediction, so two different methods were examined. First, a recursive method was tested. The LSTM model was used to make one-step-ahead predictions, then the output was looped into the input so the model could make another one-step-ahead prediction. By repeating this process multiple times, the one-step-ahead model can make 24-hour-ahead predictions. However, any error seen between the estimated and actual values during the one-step-ahead prediction was propagated, as the model uses the estimated prediction with its error to make the next prediction. Over a 24-step process, this led to poor results and a higher RMSE (i.e., 4.81) for the LSTM when just using the feature by itself. This is significantly larger than the direct method's mean RMSE of 0.036, reported in last three columns in Table 4.

For the direct method, the training and test data were first decimated to only include data recorded once every 24 hours. In this way, the models would still only be making one-step-ahead predictions, but that step would be for 24 hours rather than 1 hour. When a direct approach was taken, the 1- and 24-hour-ahead predictions generated similar results. In general, LSTM performs better with fewer, more focused features, and VIF reduction helps enhance model consistency by reducing the standard deviation of the RMSE. SVR consistently performed the best of the three models in regard to the 24-hour-ahead prediction. Again, each SVR's RMSE was on the same order of magnitude, suggesting that this model is robust with regard to inputs. RF saw its best performance by using the SHAP-determined inputs, but it once again demonstrated poor performance when using highly correlated variables.

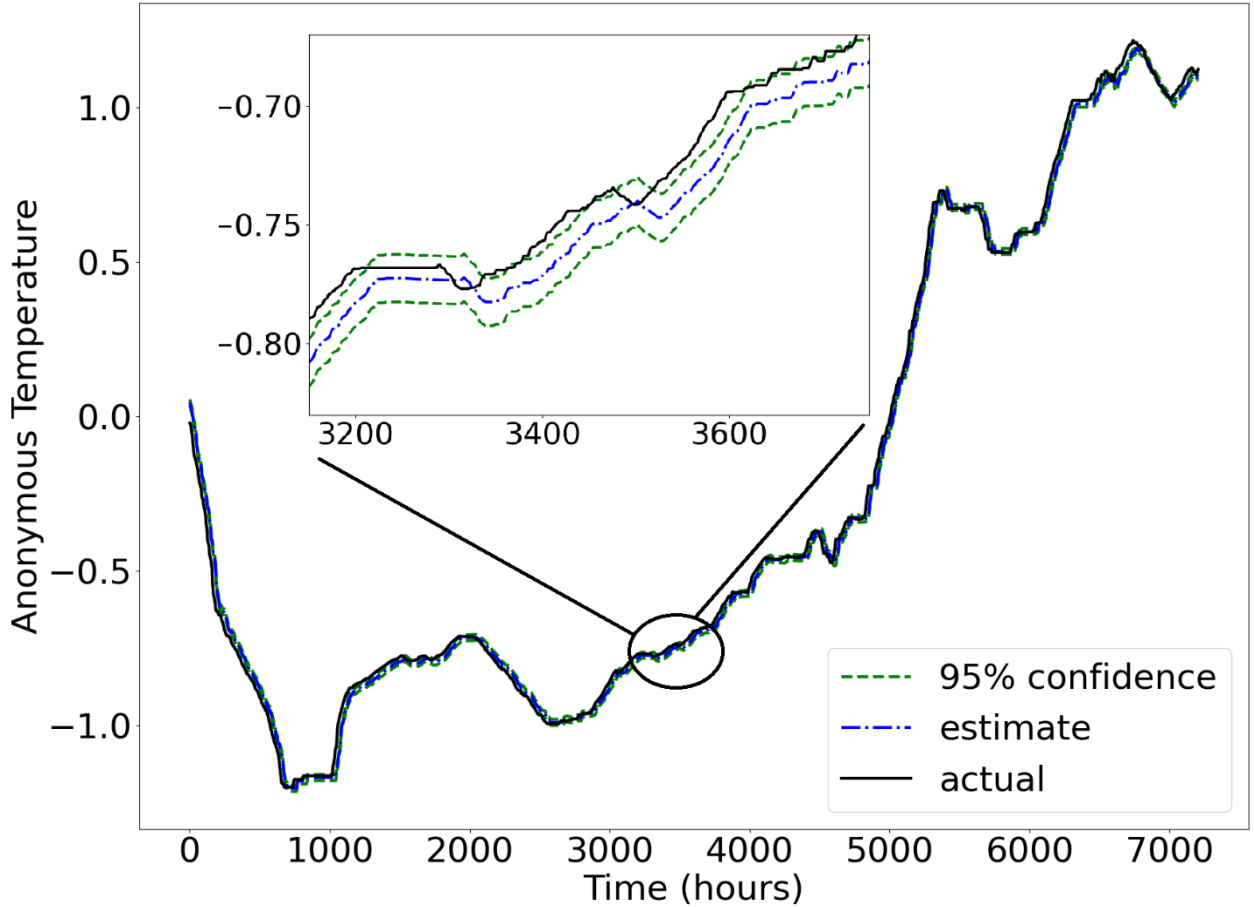


Figure 4. Predictions of pump temperature 1 day ahead, using the SHAP-determined input to an LSTM model. This includes a closer view of the pump temperature predictions in order to provide a clearer view of the 95% confidence interval.

To better compare the results from the 1- and 24-hour-ahead predictions, the LSTM results from using SHAP-determined inputs are given in Figure 4. Unlike the previous 1-hour-ahead predictions, the 24-hour-ahead predictions do not always reflect the actual measurements closely enough to fall within the 95% confidence interval.

For LSTM and SVR, the SHAP-determined inputs produced the best results with respect to the mean RMSEs for both prediction horizons. However, the best RF models contained only the response variable as an input. By adding more variables, a long train of outliers began to crop up. The outliers for the 1-hour-ahead prediction using RF can be seen in Figure 5. The majority of the model's predictions are close to the actual measurement. However, a large, prolonged deviation can be seen between hours 1,000 and 2,000, and another large deviation can be seen around hour 6,200 (both indicated by red arrows). These deviations were not seen in the results from the LSTM and SVR, which used the same training dataset. This suggests that LSTM and SVR are more adept at learning the relationships between multiple predictor variables in a regression analysis.

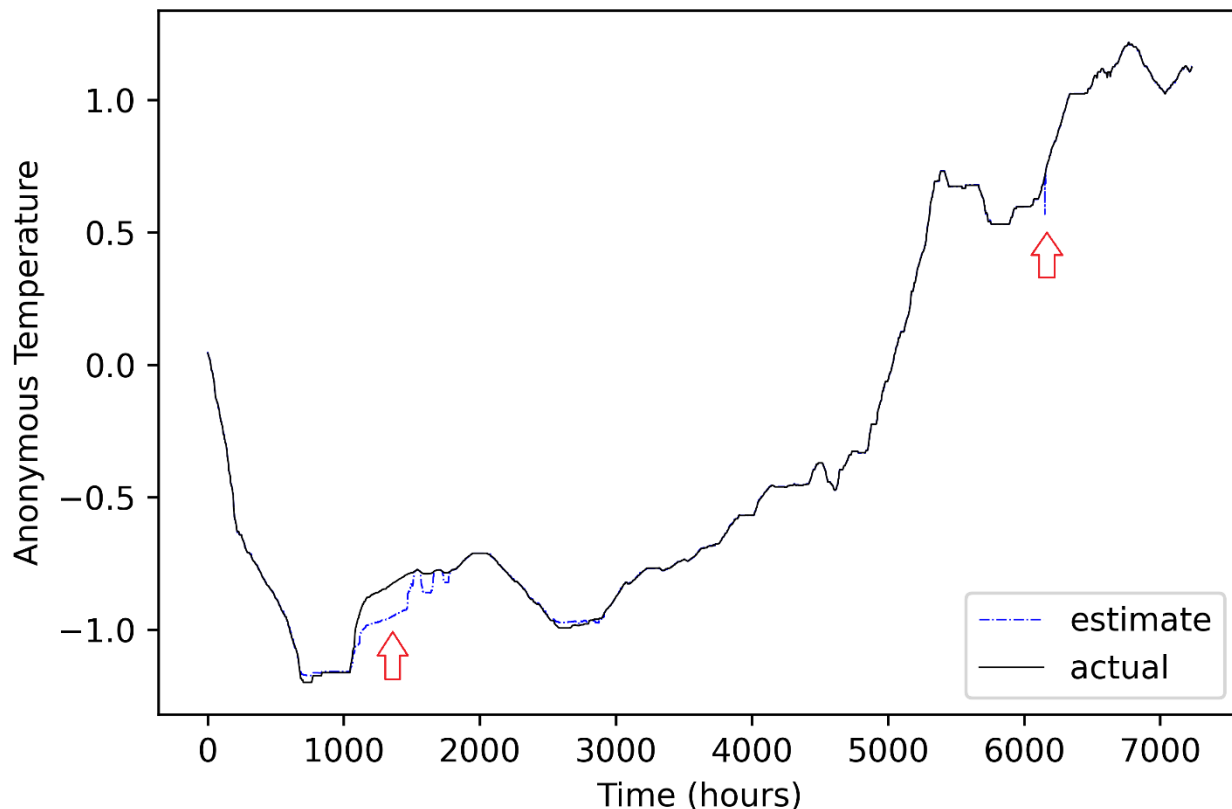


Figure 5. RF using SHAP inputs and making 1-hour-ahead predictions showed a higher RMSE, due to poor predictions between time steps 1000 and 2000, as well as for the time step at time 6,200.

The FCS in this plant contained three primary loops. These trains of pumps operate in loops that are parallel to one another. If one loop decreases its flow, the other must take on more flow so that the reactor core receives a constant supply of coolant during steady-state operation. The site of pump 2's bearing temperature measurement was located on a pump in a loop parallel to pump 1a. When the pump works harder to push more coolant, the temperature increases. It is suspected that this temperature fluctuation can be used to indirectly describe the amount of flow through the pump—and subsequently, the amount of flow through that particular loop. Pump 2's temperature may then be indirectly providing new information on how the train of pumps within each loop is being operated. The site of the pump bearing temperature measurement for pump 3 was located within the third loop of the system. Pump 3's bearing temperature seemed to provide very little new information for the analysis, as indicated by the low mean SHAP value in Table 5. With two out of three flows being indirectly calculated via these temperatures, the third flow may also be assumed, given an absence of leaks or degradation, since the overall flow to the core remains constant during steady-state operation. Pumps 2 and 3 may then be providing some redundant information about the operation of the other loops. Based on the mean SHAP values, the

SHAP-determined feature set contained two variables: the bearing temperature of pump 1a (the target variable being predicted) and the bearing temperature of pump 2.

Table 5. Mean SHAP values for determining feature importance.

Component	Mean (SHAP value)
Pump 1a	0.810
Pump 2	0.110
Pump 3	0.015
Pump 1b	0.011

LSTM, SVR, and RF each showed comparable results for the 1-hour-ahead predictions when the optimal input features were selected. For the 24-hour-ahead predictions, SVR was the clear winner, followed by LSTM and then RF. This does not necessarily speak to LSTM's prediction capabilities in general. For LSTM, the model's hyperparameters were chosen via grid search, and these same hyperparameters were used for the entirety of the study, regardless of inputs. By re-optimizing the hyperparameters based on each feature set, the results may improve. However, in the context of this study, it would be harder to distinguish whether this improvement was based on the input or on hyperparameter selection.

The SHAP values determined the best set of inputs for both LSTM and SVR in this study. This is a wrapper-based method that considers both the correlations and dependencies between the inputs. VIF is a filter-based method that primarily focuses on the input variables' relationship with the response variable, not the information contained within the other predictor variables. Although VIF reduction successfully reduced the total amount of multicollinearity within the input space, as well as the variability in the model predictions, this does not always yield the optimal set of inputs in comparison to other methods.

5.4 Validation

For the V&V process, the same methodologies and workflow were performed on three independent datasets. The first came from a BWR, while the other two came from pressurized water reactors (PWRs). The specific systems of interest include a BWR's CP and CBP, a PWR's steam generator feedpump, and another PWR's main turbine. The fundamentals of each component and the specific signals recorded are explained below.

5.4.1 BWR Condensate Pumps and Condensate Booster Pumps

CPs and CBPs are found within the FCS, the primary purpose of which is to condense steam and collect the drainage in the main condenser before purifying, preheating, and pumping the water back to the reactor vessel [25]. The CPs provide the driving force for pushing the condensate through auxiliary systems such as the steam jet air ejectors condenser, steam packing exhaust condenser, off-gas condenser, and demineralizers, all of which work to condition the condensate. Afterwards, the CBPs are the driving force of the flow as the condensate travels through a string of low-pressure heaters that work to preheat the water to the correct temperature. In the BWR system of interest, the CP and CBP are driven by a shared motor.

5.4.2 PWR Steam Generator Feedpump

In a PWR system, the primary coolant flows from the reactor to the steam generator to transfer its thermal energy to the secondary coolant via numerous tubes [46]. With sufficient heat, the secondary coolant starts to boil into steam before being sent to the turbine system. The steam generator feedpump is what supplies the driving force for the secondary side's coolant to flow through the steam generator.

5.4.3 PWR Turbine

In a PWR system, the steam formed in the steam generator is passed to the main turbine generator, where it is used to make electricity [46]. The steam is then directed to low-pressure turbines before being routed to the main condenser. Throughout this process, the steam is piped through moisture separators and reheaters that dry and reheat the steam to prevent potential turbine damage due to the moisture content of the steam, while also improving the efficiency of the turbine.

5.4.4 Validation Model Performance

This section focuses on how well LSTM and SVR predicted plant process datasets over two prediction horizons (1 hour and 1 day) using different datasets. Three different multi-step approaches were tested for the main turbine thrust bearing temperature in order to determine which approach would be used for all other predictions. The direct, recursive, and DirRec multi-step approaches produced RMSE's of 0.294, 4.816, and 0.301, respectively, for the anonymized bearing temperature dataset. For this model, the recursive method results were considered poor, due to the much larger RMSE than seen for the other two methods. This larger RMSE was due to prediction error accumulation while recursively looping the output from the one-step-ahead model back into the input. The predictions obtained by applying the direct and Dir-Rec methods on the anonymized temperature data can be seen in Figure 6. The direct method, seen in green, more closely follows the peaks as the temperature of the bearing oscillates. The cause of this oscillation, which is on the order of 1–2°C, is unknown, due to a lack of information on how the unit was operated or if the system had any underlying conditions. Although the DirRec method had comparable results, the direct method was chosen thanks to having the lowest RMSE, and was used for the remainder of the V&V process.

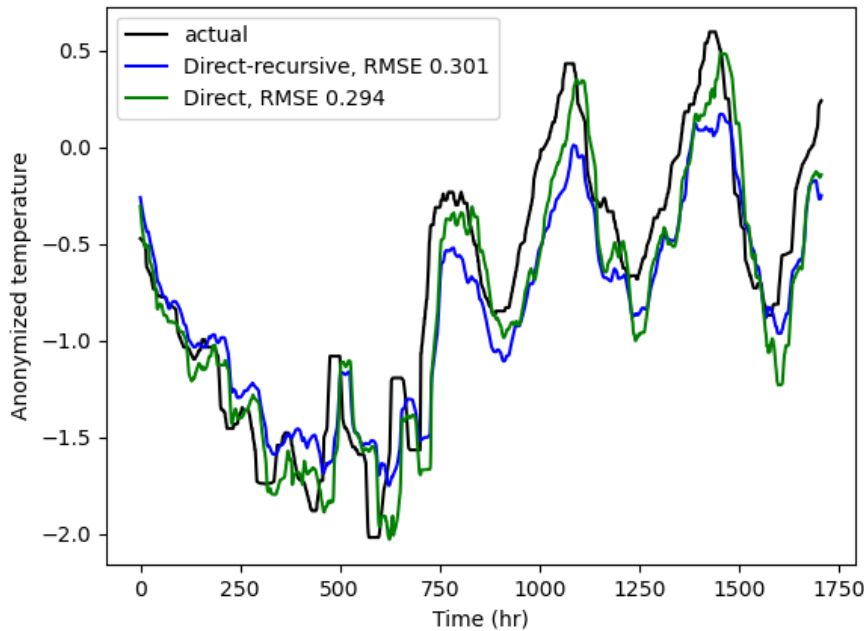


Figure 6. Comparison of direct vs. direct-recursive forecasting strategies.

LSTM and SVR were used make 1-hour (i.e., one step) and one-day (i.e., 24 step)-ahead predictions, using data from three different NPPs (i.e., PWR 1, PWR 2, and BWR). The PWR 1 and PWR 2 datasets contained a year worth of data, while the BWR dataset contained 5 years worth. The mean of the RMSEs from the cross-validation approach demonstrates the modeling method's expected accuracy when used to make predictions. The RMSEs' means and standard deviations from the LSTM and SVR models for the 1-hour- and 1-day-ahead predictions are shown in Table 6.

Table 6. 1-hour- and 1-day-ahead predictions for both LSTM and SVR.

Dataset			1 step ahead		24 steps ahead	
Plant	Parameter Predicted	Model	Mean RMSE	Std Error	Mean RMSE	Std Error
PWR 1	Main Turbine Bearing Temp	LSTM	0.0796	0.0411	0.7932	0.5450
		SVR	0.0214	0.0080	0.3194	0.1202
PWR 1	Generator Output	LSTM	0.2871	0.2031	2.2636	2.8338
		SVR	0.0806	0.0422	1.5611	1.2424
PWR 2	Steam Generator Flow	LSTM	2.4792	3.0455	12.435	17.333
		SVR	1.4070	2.4270	5.6299	5.3154
BWR	Condensate Pump Bearing Temp	LSTM	0.0792	0.0722	0.2991	0.2724
		SVR	0.0323	0.0496	0.2238	0.2184

In Table 6, the RMSEs were calculated after converting the predictions back into their respective engineering unit of measurement. Temperatures were captured in °C, generator output in MW, and steam generator flow in thousands of gallons per minute. The first point of comparison regards the prediction accuracy of SVR compared to that of LSTM. In each dataset, SVR outperformed LSTM. This may be due to how each model was optimized. SVR has a convex optimization, so choosing the optimal hyperparameters (i.e., regularization parameter, epsilon tube, and kernel coefficient) can be done efficiently. For LSTM, there are many hyperparameters to consider, some of which are listed in Table 1. This makes optimization more difficult and computationally expensive. After the hyperparameters are chosen, the LSTM model still must be trained to the dataset of interest. In this report, the optimal hyperparameters were found for one dataset and then applied to every other dataset. Of all the datasets, the BWR CP bearing temperature dataset, in which the LSTM hyperparameters were optimized, showed results that were closest, in terms of mean RMSE, with those of the SVR. This suggests that further improvements may be seen if, with each new dataset, the optimal architecture is selected for each LSTM.

In comparing the results found within Table 6, one notices that 24-step-ahead predictions perform worse than one-step-ahead predictions. This is logical, as more events and transients can occur over a longer prediction horizon, making forecasting more difficult. Another trend that sometimes becomes apparent is a lag in the 24-step-predictions (see Figure 7). The one-step-ahead predictions for both LSTM and SVR seem to trend well with the actual measurements; however, the 24-step-ahead predictions produce a noticeable delay. The predictions mimic the same patterns as the temperature fluctuations, but do not accurately predict them as they occur. This is most likely due to a lack of predictor information for informing the model of the cause of these 1–2°C fluctuations that seemingly oscillated over a two-week period. The lack of information (e.g., operating or maintenance logs) prevented the source of the fluctuations from being determined. With the derates removed, the 24-step-ahead model was still expected to predict 24 steps ahead, even though this no longer corresponded with the basis of the original 24-hour-ahead prediction. This could be one cause of the inflated mean RMSE values.

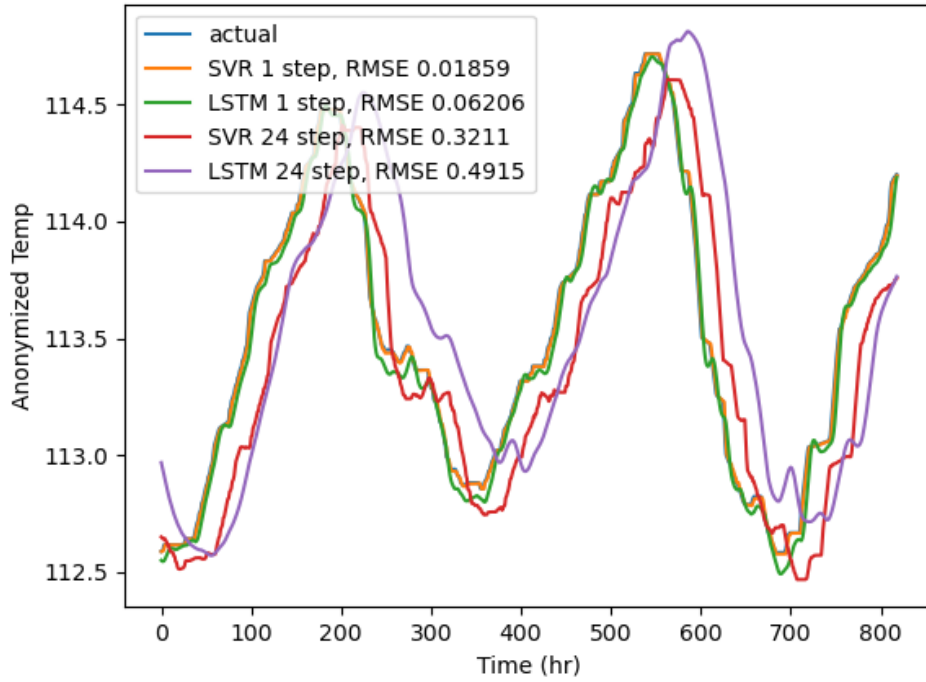


Figure 7. Forecasting the main turbine thrust bearing temperature using LSTM and SVR for both one- and 24-step-ahead predictions.

Another interesting comparison relates to LSTM's 1-step- and 24-step-ahead predictions of the PWR 1 main turbine bearing temperature, as compared to LSTM's 1-step- and 24-step-ahead predictions of the BWR CP bearing temperature. The mean RMSEs for both of the one-step-ahead predictions are very close to each other: 0.0796 and 0.0792, respectively. However, the 24-step-ahead predictions deviate significantly with mean RMSEs of 0.7932 and 0.2991, respectively. The difference seen between the mean RMSEs of the 24-step-ahead predictions most likely attributable to the increased amount of data in the BWR dataset (5 years worth vs. 1 year worth) having yielded increased prediction accuracy. Although LSTM still did not beat SVR in terms of overall accuracy, it did see a bigger improvement when moving from 1-step- to 24-step-ahead predictions for this variable.

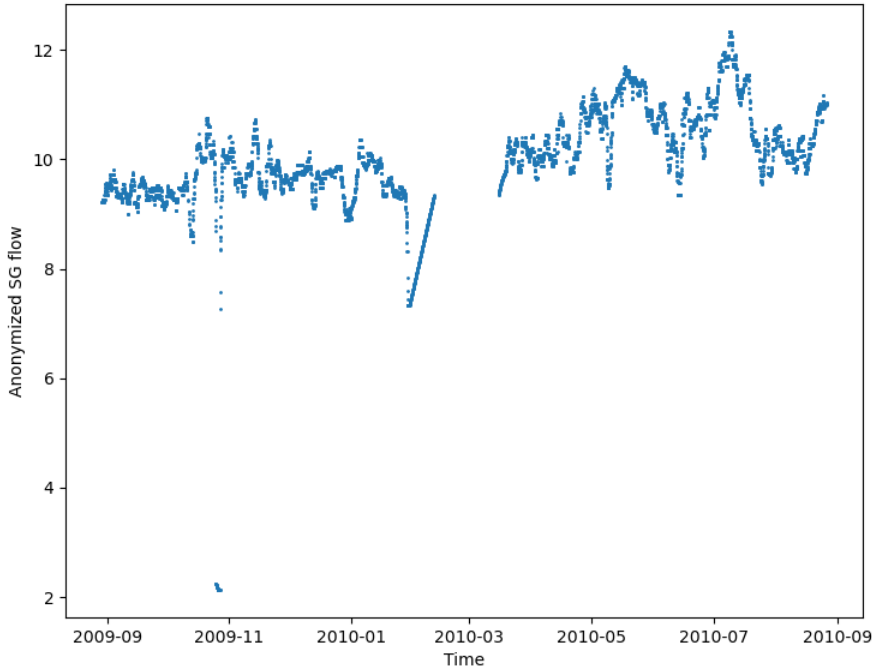


Figure 8. Anonymized steam generator flow and time, showing a significant derate at 2009-11 and an outage at 2010-03.

For both the models and prediction horizons in Table 6, the RMSE standard deviation for the PWR 2 flow predictions are large. This is because two separate sections of the flow data represent abnormal behavior (see Figure 8, where the y-axis was anonymized and the x-axis was shifted to protect the plant's identity). In Figure 8, the first large deviation is seen at time 2009-11. This large reduction in flow occurred when the reactor's thermal power exceeded 90%. The other section of abnormal behavior occurred at around 2010-02, when the reactor was being ramped down toward a regularly scheduled outage that lasted through 2010-03. The flow appears to linearly increase in a strict fashion. Each of these sections were poorly predicted, as their behaviors were not captured in the training data. With these two, operational anomalies present, the mean RMSE for the SVR 24-step-ahead-predictions was 5.629. After removing these outliers from the testing data, the mean RMSE fell to 1.744. In the data cleaning process, each of these abnormal sections were missed. The large derate in flow remained within three standard deviations, and the linearly increasing section occurred while the reactor's thermal power was above 90%.

Overall, each model made good predictions based on the new datasets. However, as seen with the two flow outliers, data-driven models can only effectively be relied upon to predict activity that has been seen within the training dataset. Model performance can be improved by supplementing the training set with more information if available; although, unfortunately, this is rarely the case.

6. SUMMARY AND PATH FORWARD

This report demonstrated the expected accuracy and consistency of short-term forecasting ML models when given new data from different systems. Prognostic and short-term forecasting models are a key enabling technology that can be used to reduce operations and maintenance costs by enabling condition-based predictive maintenance. This research showed that a model's prediction accuracy depends on several factors, including the model's architecture, the prediction horizon, and the training dataset itself. Each ML model must be optimized in some way to the dataset of interest, as the model's performance

directly relies on it. SVRs have fewer hyperparameters than LSTMs, making them less computationally expensive to optimize. This research also showed that model performance degraded as the prediction horizon increased. In addition, this research showed the importance of having a representative training dataset. When the plant experiences transients not within the dataset, as seen in the PWR 2 flow predictions, the model cannot be relied on to predict such unseen transients accurately. The analysis in this report only covered operating conditions at steady-state conditions (as defined by the reactor's thermal power). There is a need to expand this work to non-steady-state operating conditions in order to cover the full range of the reactor's operating lifecycle (i.e., derates, trips, and ramp downs toward outages). During these transient operating conditions, the system is vulnerable to degradation, and knowledge of the future condition of critical operating equipment is essential for enabling predictive maintenance.

This report also described the SHAP feature importance values for selecting a representative dataset, based on feature importance. Other feature selection techniques (i.e., filter, wrapper, and embedded methods [47]) and dimensionality-reduction methods (e.g., principal component analysis) should be explored to determine whether an optimal input variable space can be developed. Feature importance values can potentially identify the features that most contribute when predicting future plant parameters. This could be incredibly useful when predicting certain parameters, such as gross load. In the Results section (i.e., Section 5), low pressure export flow was a major influencer for forecasting gross load. As low-pressure steam is exported to other areas of the plant, the gross load is reduced. With a wide array of sensors across many components and systems, new critical components can be determined based on how they influence other system parameters. These components would also then need to be analyzed by subject matter experts to determine the extent of the causal relationship.

This research will continue to investigate how feature selection can improve model performance, and perhaps more importantly, how to display the model's results to the human-in-the-loop who is required to make decisions. This research will also begin to look at the data-to-human pipeline and how that is affected by cloud services.

7. ACKNOWLEDGEMENTS

This report was made possible through funding from the U.S. Department of Energy's Office of Nuclear Energy under the Nuclear Energy Enabling Technologies Program. We are grateful to Melissa Bates at the U.S. Department of Energy and Patrick Calderoni at Idaho National Laboratory for championing this effort. The data used in this research were provided by Exelon Generation Company. We are also thankful to the plant engineers for their technical discussions on the data.

8. REFERENCES

- [1] Zhang, Y., Z. Peng, Y. Guan, and L. Wu. 2021. "Prognostics of battery cycle life in the early-cycle stage based on hybrid model." *Energy* 221, 119901. <https://doi.org/10.1016/j.energy.2021.119901>.
- [2] Gohel, H. A., H. Upadhyay, L. Lagos, K. Cooper, and A. Sanzetenea. 2020. "Predictive maintenance architecture development for nuclear infrastructure using machine learning." *Nucl. Eng. Technol.* 52 (7): 1436–1442. <https://doi.org/10.1016/j.net.2019.12.029>.
- [3] Müller, I. M. 2021. "Feature selection for energy system modeling: Identification of relevant time series information." *Energy AI* 4, 100057. <https://doi.org/10.1016/j.egyai.2021.100057>.
- [4] Salcedo-Sanz, S., L. Cornejo-Bueno, L. Prieto, D. Paredes, and R. García-Herrera. 2018. "Feature selection in machine learning prediction systems for renewable energy applications." *Renew. Sustain. Energy Rev.* 90, (March): 728–741. <https://doi.org/10.1016/j.rser.2018.04.008>.
- [5] Remeseiro, B., and V. Bolon-Canedo. 2019. "A review of feature selection methods in medical applications." *Comput. Biol. Med.* 112 (July): 103375. <https://doi.org/10.1016/j.compbiomed.2019.103375>.

- [6] Castanedo, F. 2013. "A review of data fusion techniques." *Sci. World J.* 2013. <https://doi.org/10.1155/2013/704504>.
- [7] Atamuradov, V., K. Medjaher, P. Dersin, B. Lamoureux, and N. Zerhouni. 2017. "Prognostics and Health Management for Maintenance Practitioners - Review, Implementation and Tools Evaluation." *Int. J. Progn. Heal. Manag.* 8. <https://doi.org/10.36001/ijphm.2017.v8i3.2667>.
- [8] Song, F., Z. Guo, and D. Mei. 2010. "Feature selection using principal component analysis." *Proc. - 2010 Int. Conf. Syst. Sci. Eng. Des. Manuf. Informatiz. ICSEM 2010* 1, 27–30, 2010. <https://doi.org/10.1109/ICSEM.2010.14>.
- [9] Davò, F., S. Alessandrini, S. Sperati, L. Delle Monache, D. Airolidi, and M. T. Vespucci. 2016. "Post-processing techniques and principal component analysis for regional wind power and solar irradiance forecasting." *Sol. Energy* 134, 327–338. <https://doi.org/10.1016/j.solener.2016.04.049>.
- [10] Zebari, R., A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed. 2020. "A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction," *J. Appl. Sci. Technol. Trends* 1 (2): 56–70. <https://doi.org/10.38094/jastt1224>.
- [11] Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. 2002. "SMOTE: synthetic minority over-sampling technique." *J. Artif. Intell. Res.* 16 (March): 321–357. <https://doi.org/10.1002/eap.2043>.
- [12] Karasu, S., A. Altan, S. Bekiros, and W. Ahmad. 2020. "A new forecasting model with wrapper-based feature selection approach using multi-objective optimization technique for chaotic crude oil time series." *Energy* 212, 118750. <https://doi.org/10.1016/j.energy.2020.118750>.
- [13] Niu, T., J. Wang, H. Lu, W. Yang, and P. Du. 2020. "Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting." *Expert Syst. Appl.* 148, 113237. <https://doi.org/10.1016/j.eswa.2020.113237>.
- [14] Monirul Kabir, M., M. Monirul Islam, and K. Murase. 2010. "A new wrapper feature selection approach using neural network." *Neurocomputing* 73 (16–18): 3273–3283. <https://doi.org/10.1016/j.neucom.2010.04.003>.
- [15] Shahidi, P., D. Maraini, and B. Hopkins. 2020. "Railcar Diagnostics Using Minimal-Redundancy Maximum- Relevance Feature Selection and Support Vector Machine Classification." *Int. J. Progn. Heal. Manag.* 7 (4): 1–13. <https://doi.org/10.36001/ijphm.2016.v7i4.2524>.
- [16] Sendlbeck, S., A. Fimpel, B. Siewerin, M. Otto, and K. Stahl. 2021. "Condition monitoring of slow-speed gear wear using a transmission error-based approach with automated feature selection." *Int. J. Progn. Heal. Manag.* 12 (2): 1–15. <https://doi.org/10.36001/IJPHM.2021.V12I2.3026>.
- [17] Sthle, L., and S. Wold. 1989. "Analysis of variance (ANOVA)." *Chemom. Intell. Lab. Syst.* 6 (4): 259–272. [https://doi.org/10.1016/0169-7439\(89\)80095-4](https://doi.org/10.1016/0169-7439(89)80095-4).
- [18] Bechhoefer, E., R. Schlanbusch, and T. I. Waag. 2016. "Techniques for large, slow bearing fault detection." *Int. J. Progn. Heal. Manag.* 7 (1): 1–12. <https://doi.org/10.36001/ijphm.2016.v7i1.2358>.
- [19] Ozturk, T., M. Talo, E. Azra, U. Baran, and O. Yildirim. 2020. "Automated detection of COVID-19 cases using deep neural networks with X-ray images." *Comput. Biol. Med.* 121, 103792. <https://doi.org/10.1016/j.combiomed.2020.103792>.
- [20] Manjunatha, K., and V. Agarwal. 2021. "Multi-Band Heterogeneous Wireless Network Architecture for Industrial Automation: A Techno-Economic Analysis." <https://doi.org/10.1007/s11277-021-09303-4>.
- [21] Manjunatha, K., and V. Agarwal. 2019. "Review of Wireless Communication Technologies and Techno-Economic Analysis." INL/EXT-19-53966, Idaho National Laboratory.

- [22] Ramuhalli, P., C. Walker, V. Agarwal, and N. J. Lybeck. 2020. "Development of Prognostic Models Using Plant Asset Data." ORNL/TM-2020/1697, Oak Ridge National Laboratory.
- [23] Walker, C., V. Agarwal, and N. Lybeck. 2021. "Nuclear Power Fault Diagnostics and Preventative Maintenance Optimization."
- [24] Walker, C., V. Agarwal, N. Lybeck, P. Ramuhalli, and M. Taylor. 2022. "Development of Short-Term Forecasting Models Using Plant AssetData and Feature Selection." Under review.
- [25] NRC. 1998. "BWR/4 Technology Manual (R-104B)." ML022830867, 1–442, Accessed February 2, 2022. <https://www.nrc.gov/docs/ML0228/ML022830867.pdf>.
- [26] Akinwande, M. O., H. G. Dikko, and A. Samson. 2015. "Variance Inflation Factor: As a Condition for the Inclusion of Suppressor Variable(s) in Regression Analysis." *Open J. Stat.* 5, (7): 754–767. <https://doi.org/10.4236/ojs.2015.57075>.
- [27] Daoud, J. I. 2018. "Multicollinearity and Regression Analysis." *J. Phys. Conf. Ser.* 949 (1). <https://doi.org/10.1088/1742-6596/949/1/012009>.
- [28] Yildirim, H., and M. Revan Özkale. 2019. "The performance of ELM based ridge regression via the regularization parameters." *Expert Syst. Appl.* 134: 225–233, <https://doi.org/10.1016/j.eswa.2019.05.039>.
- [29] Pan, F., et al. 2020. "Factors associated with death outcome in patients with severe coronavirus disease-19 (Covid-19): A case-control study." *Int. J. Med. Sci.* 17 (9): 1281–1292. <https://doi.org/10.7150/ijms.46614>.
- [30] Yu, J., B. Hong, J. Y. Park, J. H. Hwang, and Y. K. Kim. 2021. "Impact of Prognostic Nutritional Index on Postoperative Pulmonary Complications in Radical Cystectomy: A Propensity Score-Matched Analysis." *Ann. Surg. Oncol.* 28 (3): 1859–1869. <https://doi.org/10.1245/s10434-020-08994-6>.
- [31] O'Brien, R. M. 2007. "A caution regarding rules of thumb for variance inflation factors." *Qual. Quant.* 41 (5): 673–690. <https://doi.org/10.1007/s11135-006-9018-6>.
- [32] Booth, A. L., E. Abels, and P. McCaffrey. 2021. "Development of a prognostic model for mortality in COVID-19 infection using machine learning." *Mod. Pathol.* 34 (3): 522–531. <https://doi.org/10.1038/s41379-020-00700-x>.
- [33] Mangalathu, S., S. H. Hwang, and J. S. Jeon. 2020. "Failure mode and effects analysis of RC members based on machine-learning-based SHapley Additive exPlanations (SHAP) approach." *Eng. Struct.* 219 (May): 110927. <https://doi.org/10.1016/j.engstruct.2020.110927>.
- [34] Marcilio, W. E., and D. M. Eler. 2020. "From explanations to feature selection: Assessing SHAP values as feature selection mechanism." *Proc. - 2020 33rd SIBGRAPI Conf. Graph. Patterns Images, SIBGRAPI 2020*, 340–347. <https://doi.org/10.1109/SIBGRAPI51738.2020.00053>.
- [35] Lundberg, S. M., and S. I. Lee. 2017. "A unified approach to interpreting model predictions." *Adv. Neural Inf. Process. Syst.* 2017-Decem (2): 4766–4775.
- [36] Lundberg, S. M., G. G. Erion, and S. I. Lee. 2018. "Consistent individualized feature attribution for tree ensembles."
- [37] Pokharel, S., P. Sah, and D. Ganta. 2021. "Improved prediction of total energy consumption and feature analysis in electric vehicles using machine learning and shapley additive explanations method." *World Electr. Veh. J.* 12, no. 3. <https://doi.org/10.3390/wevj12030094>.
- [38] Hong, C. W., C. Lee, K. Lee, M. S. Ko, and K. Hur. 2020. "Explainable artificial intelligence for the remaining useful life prognosis of the turbofan engines." *Proc. 3rd IEEE Int. Conf. Knowl. Innov. Invent.* 144–147. <https://doi.org/10.1109/ICKII50300.2020.9318912>.

- [39] Alzubi, J., A. Nayyar, and A. Kumar. 2018. "Machine Learning from Theory to Algorithms: An Overview." *J. Phys. Conf. Ser.* 1142, no. 1. <https://doi.org/10.1088/1742-6596/1142/1/012012>.
- [40] Diez-Olivan, A., J. Del Ser, D. Galar, and B. Sierra. 2019. "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0." *Inf. Fusion* 50 (October): 92–111. <https://doi.org/10.1016/j.inffus.2018.10.005>.
- [41] Cabrera, D., et al. 2020. "Bayesian approach and time series dimensionality reduction to LSTM-based model-building for fault diagnosis of a reciprocating compressor." *Neurocomputing* 380, 51–66. <https://doi.org/10.1016/j.neucom.2019.11.006>.
- [42] Greff, K., R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber. 2017. "LSTM: A Search Space Odyssey." *IEEE Trans. Neural Networks Learn. Syst.* 28 (10): 2222–2232. <https://doi.org/10.1109/TNNLS.2016.2582924>.
- [43] Jain, R. K., K. M. Smith, P. J. Culligan, and J. E. Taylor. 2014. "Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy." *Appl. Energy* 123, 168–178. <https://doi.org/10.1016/j.apenergy.2014.02.057>.
- [44] Moon, J., Y. Kim, M. Son, and E. Hwang. 2018. "Hybrid short-term load forecasting scheme using random forest and multilayer perceptron." *Energies* 11(12): 1–20. <https://doi.org/10.3390/en11123283>.
- [45] Chornovol, O., G. Kondratenko, I. Sidenko, and Y. Kondratenko. 2020. "Intelligent forecasting system for NPP's energy production." *Proc. 2020 IEEE 3rd Int. Conf. Data Stream Min. Process*, 102–107. <https://doi.org/10.1109/DSMP47368.2020.9204275>.
- [46] U.S. NRC. 2007. "Reactor Concepts Manual: Pressurized Water Reactor (PWR) Systems." 603, 28. Accessed February 2, 2022. <https://www.nrc.gov/reading-rm/basic-ref/students/for-educators/04.pdf>.
- [47] Ahmed, A., and M. Khalid. 2018. "An intelligent framework for short-term multi-step wind speed forecasting based on Functional Networks." *Appl. Energy* 225, (April): 902–911. <https://doi.org/10.1016/j.apenergy.2018.04.101>.