Development of Short-Term Forecasting Models Using Plant Asset Data and Feature Selection

Cody Walker[1], Pradeep Ramuhalli[2], Vivek Agarwal[1], Nancy J. Lybeck[1], and Michael Taylor [3]

[1] *Idaho National Laboratory, Idaho Falls, ID 83415, USA*
{*cody.walker, vivek.agarwal, nancy.lybeck*}*@inl.gov*

[2] *Oak Ridge National Laboratory, Oak Ridge, TN 37830, USA*
ramuhallip@ornl.gov

[3] *SANS Technology Institute, Philadelphia, PA 19104, USA*
mtaylor@sans.org

ABSTRACT

Nuclear power plants 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 future equipment condition. Maintenance operations can then be planned in advance whenever degraded performance is predicted, thus resulting in fewer unplanned outages and the optimization of maintenance activities. This enables lower maintenance costs and improves the overall economics of nuclear power.

This paper focuses on developing 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 of 0.036 ± 0.014, 0.0026 ± 0, and 0.063 ± 0.004 RMSE, 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.

1. INTRODUCTION

Nuclear plant sites collect and store large volumes of data 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 developing 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 set of features are selected. It is well known that input features significantly impact the model's prediction performance and training time (Zhang, Peng, Guan, & Wu, 2021; Gohel, Upadhyay, Lagos, Cooper, & Sanzetenea, 2020; Müller, 2021; Salcedo-Sanz, Cornejo-Bueno, Prieto, Paredes, & García-Herrera, 2018). When facing a large number of in-

https://doi.org/10.36001/IJPHM.2022.v13i1.3120

put features, the dimensionality may be reduced through feature extraction, feature selection, or a combination of the two (Remeseiro & Bolon-Canedo, 2019; Godwin & Matthews, 2013).

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 (Atamuradov, Medjaher, Dersin, Lamoureux, & Zerhouni, 2017). One example of feature extraction, as found in wind turbine health monitoring, is the absolute difference in blade angle position (Godwin & Matthews, 2013). Another example of feature extraction is principal component analysis (PCA), in which input features are combined to produce a new set of orthogonal features (Song, Guo, & Mei, 2010; Davò et al., 2016). 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 (Davò et al., 2016). 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 (Zebari, Abdulazeez, Zeebaree, Zebari, & Saeed, 2020). Filters are open-loop methods that measure feature characteristics (e.g., information, dependency, consistency, and distance) while being fast and scalable (Zebari et al., 2020; Remeseiro & Bolon-Canedo, 2019; Hall & Smith, 1998). 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 (Karasu, Altan, Bekiros, & Ahmad, 2020; Niu, Wang, Lu, Yang, & Du, 2020). However, wrappers are prone to overfitting and can be computationally expensive (Zebari et al., 2020). Hybrid and ensemble methods integrate filters and wrappers alike, thereby benefiting from their complementary approaches (Monirul Kabir, Monirul Islam, & Murase, 2010; Zebari et al., 2020).

Other common feature selection techniques—apart from the ones used in this paper—include mutual information (MI) (Shahidi, Maraini, & Hopkins, 2020), recursive feature elimination (Sendlbeck, Fimpel, Siewerin, Otto, & Stahl, 2021), and analysis of variance (ANOVA) tests (Bechhoefer, Schlanbusch, & Waag, 2016). MI uses entropy as a means of determining the amount of information" gained by each input feature. MI has been used as the basis for a minimally redundant, maximally relevant feature selection method for multi-class support vector machine classification of railcar conditions (Shahidi et al., 2020). 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 (Sendlbeck et al., 2021). 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, recursive feature elimination was used to reduce a set of 5,650 features down to the top 15 (Sendlbeck et al., 2021). ANOVA is a statistical method that uses hypothesis testing to determine whether a given result or feature is significant (Sthle & Wold, 1989). ANOVA was used to determine whether various wind turbine bearing temperature measurements significantly differed from each other (Bechhoefer et al., 2016). If the variables are not significantly different, one of them may be removed from the analysis. Other methods for feature extraction utilize deep learning to automate the feature extraction step altogether. For example, an end-to-end architecture fully automated the feature extraction process for diagnosing COVID-19 using x-ray images (Ozturk, Talo, Azra, Baran, & Yildirim, 2020).

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.

The main contributions of this paper are as follows:

1. A case study comparing the short-term forecasting capabilities of three different ML techniques for predicting a nuclear power plant's feedwater and condensate system (FCS) parameters, and how various input features affect the ML model's performance.

2. Formalization of the preprocessing steps required to integrate heterogeneous nuclear plant data. These preprocessing steps include both feature selection and the necessary data cleaning.

The rest of this paper is arranged as follows: Section 2 gives the background for the paper, 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 model hyperparameters generated for this research and compares the performance of each model, as the input features vary; and Section 5 concludes by summarizing the paper and highlighting its significance.

2. SYSTEM AND DATA DESCRIPTION

Data in this analysis is primarily recorded from the condensate pumps (CP) and condensate booster pumps (CBP) 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 (NRC, 1998). 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. Afterward, 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 condensate and condensate booster pumps are driven by a shared motor.

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

1. Generator gross load (MW)
2. Average feedwater flow rates (million gallons/second)
3. Temperatures from the feedwater pumps, CPs, CBPs, and associated motors (°C)
4. Pressures within the condenser, CPs, CBPs, and turbines (psig)
5. Current to the CP and CBP drive motors (amps).

The recorded data were primarily found within 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 offer sufficient information to determine the cause of each derate. Data preprocessing was necessary, as the data contained missing values, outliers, 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:

1. Collect, clean, and explore the data.
2. Determine the relevant features.
3. Train and evaluate the models.
4. 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 nuclear plants can be seen in Figure 1. This research is a piece of the broader vision detailed in Figure 1. The overall project focuses on addressing digital monitoring challenges that range from the deployment of wireless sensor technology in the nuclear power plant environment, to feature selection and data analytics that drive online component monitoring, to visualization for decision-making purposes. Each task is vital, as it ensures that the right people get the right information at the right time.

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, there is an excellent review paper covering critical component analysis and sensor selection, along with prognostic methodologies and tool evaluation (Atamuradov et al., 2017).

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% 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 condensate pump 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 filtering, replacing, or scaling, thereby reducing their effects on model performance. Data preprocessing includes data cleaning, feature scaling, and feature selection (Li, Verhagen, & Curran, 2019). The data preprocessing steps implemented in this research are described in the subsections below.

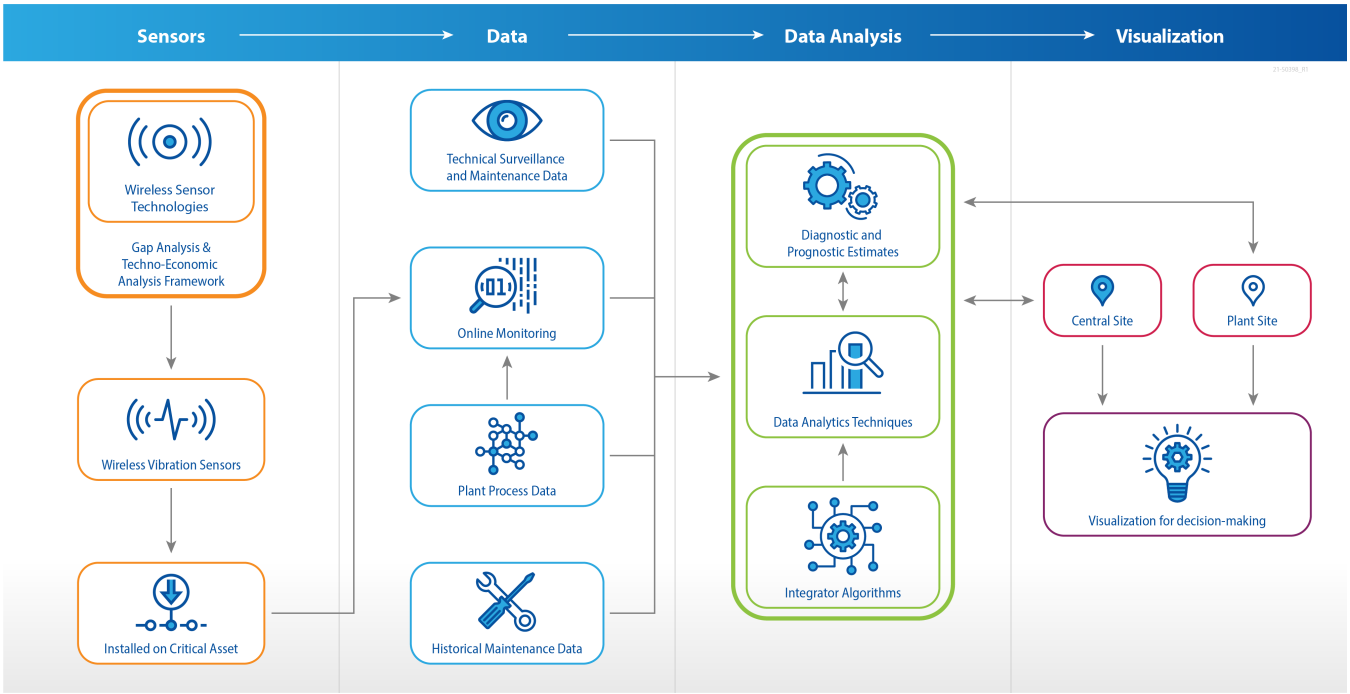

Figure 1. Steps for leveraging digital monitoring to enable cost-effective predictive maintenance for nuclear power plants.

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 are primarily noted when the system or component is offline, in which case the data are 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 paper, 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 29 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.

The two primary feature selection techniques covered in this paper 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 (Akinwande, Dikko, & Samson, 2015). 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, using PCA (Daoud, 2018; Davò et al., 2016), or by utilizing regularization techniques (e.g., ridge regression) (Yildirim & Revan Özkale, 2019). 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.

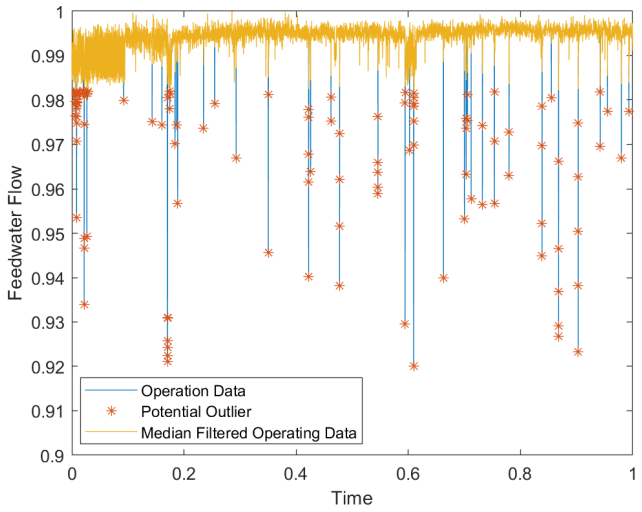

Figure 2. Outliers being identified and rectified for the average feedwater flow over time. The data range has been normalized so that the maximum value of the variable is 1, and the normalized horizontal axis represents the fraction of plant operation time.

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 (Daoud, 2018; Akinwande et al., 2015). 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 using the remaining variables, as per Eq. (1):

$$X_i = \alpha_0 + \alpha_{i+1}X_{i+1} + \alpha_{i+2}X_{i+2} + ... + e, \qquad (1)$$

where α_0 is a constant, e is an error term, X_i is the variable to check for multicollinearity, and $X_{i+1,i+2,...}$ are the other predictor variables. The second step in calculating the VIF is reflected in Eq. (2):

$$VIF_i = \frac{1}{1 - R_i^2}, \qquad (2)$$

where R_i^2 is the coefficient of determination from Eq. (1). 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. VIF has been used 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 (Pan et al., 2020). No multicollinearity was seen, as each variable had a VIF of less than two. A multivariate logistic regression was used to determine the prognostic nutritional index's impact on postoperative pulmonary complications (Yu, Hong, Park, Hwang, & Kim, 2021). They used a VIF threshold of 10 for determining highly correlated variables, then eliminated them from the analysis. VIF was used to remove multicollinear variables before estimating the state of health for lithium-ion batteries, using a gradient boosted decision tree (Zhang et al., 2021). 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, the 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 effect the variance of the i^{th} regression coefficient in Eq. (1) (O'Brien, 2007). Even though the dataset in this paper satisfies these rules of thumb, a threshold of five was selected so as to be conservative.

2.2.3. Shapley Additive Explanations

Shapley Additive Explanations, also known as 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 (Booth, Abels, & McCaffrey, 2021). 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 that deviates from the baseline model output. The feature's influence on each prediction that pushes the output either positively or negatively is taken into consideration based on different combinations of input features (Mangalathu, Hwang, & Jeon, 2020). SHAP values use an additive feature attribution approach, meaning that the output is a linear combination of the input variables (Mangalathu et al., 2020). In this manner, SHAP empirically determines the influence each feature has on the prediction output (Booth et al., 2021). Computing the exact solution for the SHAP values is, by nature, an exponential problem, typically leaning it toward being infeasible (Marcilio & Eler, 2020). However, a SHAP approximation can be made using an explanation model. The original model $f(x)$ is associated with the explanation model $g(x')$ with simplified inputs x', and is expressible as:

$$f(x) \simeq g(x') = \phi_0 + \sum_{i=1}^{M} \phi_i x'_i, \tag{3}$$

where M is the number of input features, ϕ_0 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 (Lundberg & Lee, 2017a). The SHAP library used in this research is readily available through Python's SHAP package (Lundberg & Lee, 2017b).

SHAP values have primarily been used to try to explain how a specific feature affects the model's output. For example, SHAP values with a Cox hazards model were used by Lundberg, Erion, and Lee (2018) 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), and were also used by Mangalathu et al. (2020) to identify the most important features in predicting shear wall failure modes. SHAP values were used by Pokharel, Sah, and Ganta (2021) 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. ML is often criticized as a black-box approach, but SHAP values can lead to more interpretable models. For example, SHAP values were used by Hong, Lee, Lee, Ko, and Hur (2020) to analyze feature contributions 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 (Alzubi, Nayyar, & Kumar, 2018).

3. SHORT-TERM FORECASTING MODEL SELECTION

This paper 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 (Diez-Olivan, Del Ser, Galar, & Sierra, 2019). 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 (Diez-Olivan et al., 2019). 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 represent a wide variety of ML capabilities, as one is a neural network, one is kernel-based, and one is composed of decision trees. Other ML models could have been chosen, but these 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 (RNN) that incorporates unique memory cells to learn long-term relationships between the inputs and outputs (Hochreiter & Schmidhuber, 1997). As with classical RNNs, 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 (Kong et al., 2017). 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 (Farzad, Mashayekhi, & Hassanpour, 2019). 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 paper, a grid search approach was used to optimize the LSTM hyperparameters, but random and Bayesian search methods could also have been used (Cabrera et al., 2020; Greff, Srivastava, Koutník, Steunebrink, & Schmidhuber, 2016). A more detailed guide to constructing LSTM networks is found in (Greff et al., 2016).

3.2. Support Vector Regression

Support vectors were originally introduced for classification, but later expanded to include regression (Drucker et al., 1997). 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 (Jain, Smith, Culligan, & Taylor, 2014). In this paper, a radial basis function kernel was implemented.

3.3. Random Forest

RF is a decision tree ensemble usable for either classification or regression (Moon, Kim, Son, & Hwang, 2018). 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 (Pham, Luo, & Finley, 2020; Chornovol, Kondratenko, Sidenko, & Kondratenko, 2020). 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 having at least two samples to split. The split criterion was based on the weighted impurity decrease equation given in Eq. (4):

$$\frac{N_t}{N} * (I - N_{tr}/N_t * I_r - N_{tl}/N_t * I_l) \geqslant 0, \qquad (4)$$

where N_t is the number of samples at the current node, N is the total number of samples, I is the impurity, N_{tr} is the number of samples in the right child, I_r is the right impurity, N_{tl} is the number of samples in the left child, and I_l is the left impurity (Pedregosa et al., 2011). This weighted impurity calculates whether or not the subsequent split would be beneficial.

4. RESULTS

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

4.1. Metrics Used and Model Parameters

The model outputs were compared using the root mean square error (RMSE) and mean absolute error (MAE), which can be calculated via Eq. (5) and Eq. (6), respectively.

The RMSE is given as:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|^2}, \qquad (5)$$

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 MAE is given as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|, \qquad (6)$$

where N is the total number of predictions, $\hat{y}_i$ is the model's predicted output, and y_i is the observed output. Lower RMSE and MAE values indicate better accuracy. RMSE tends to punish outliers more as the error is squared, while the MAE increases linearly.

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 paper primarily focuses on how the choice of input features affects model performance. Optimal LSTM hyperparameters were identified 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 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 prevent 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 Quality	squared error
Bootstrapping	True

Table 3. SVR parameters

Hyperparameter	Value
C	46.416
ϵ	0.044
γ	0.464

4.2. Datasets Based on VIF and SHAP Analysis

The target variable to predict in this analysis was the 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). Parameter values from associated systems such as turbine control systems were included among the 78 recorded variables. The possible input features were the recorded variables after preprocessing. These variables were grouped and used as predictors, per the criteria listed in Tables 4 and 5, respectively. The previous three time steps were used as model inputs for each model (e.g., the last 3 hours were used as inputs for the 1-step model.) No further feature engineering occurred, as this paper'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. This feature set represents the input feature space when no feature selection was used, and provides a comparison that highlights why feature selection is necessary. The second feature set started with all 78 variables, then those variables with a VIF of five or more were removed so as to eliminate the multicollinearity within this feature set. By eliminating 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 other. After removing multicollinearity from this feature set, the only remaining variable was the current pump bearing temperature itself. Used on its own, this temperature can be seen as a control group for determining whether the information added by the other variables actively helps or hurts model performance.

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 6. 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 to a low mean SHAP value.

4.3. Short-Term Forecasting Model Performance

Each of the feature sets was used as an input to make 1- 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 1-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. Table 5 shows the mean MAE and its standard deviation multiplied by 1,000. These results show that SVR provided more consistent results than the other models for this dataset.

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. The addition of 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 and MAE, as demonstrated by the feature sets with correlations of 0.8 or higher.

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 ± 258	34	92 ± 4	326 ± 26	44	132 ± 9
All with VIF reduction	36	206 ± 18	18	83 ± 4	365 ± 19	32	85 ± 6
Corr >0.9	4	8 ± 4	7	114 ± 10	37 ± 4	39	238 ± 14
Feature By itself	1	7 ± 4	8	6 ± 0.2	36 ± 2	27	51 ± 1
Corr >0.8	14	29 ± 7	11	95 ± 13	160 ± 9	28	130 ± 11
Corr >0.8 with VIF	6	6 ± 1	9	9 ± 1	52 ± 5	27	60 ± 4
Location-based	11	15 ± 3	7	61 ± 10	117 ± 15	33	153 ± 19
Location-based with VIF	8	14 ± 5	13	61 ± 10	124 ± 9	46	97 ± 14
SHAP Values	2	6 ± 2	6.6	35 ± 2	36 ± 14	26	63 ± 4

Table 5. Comparison of model performance (MAE * 1e3)

		1-hour ahead			24-hour ahead		
Selection Method	**# Features**	**LSTM**	**SVR**	**RF**	**LSTM**	**SVR**	**RF**
All	78	296 ± 27	15	83 ± 6	342 ± 35	99.8	95 ± 9
All with VIF reduction	36	226 ± 18	1.6	26 ± 1	332 ± 25	30.5	45 ± 1
Correlation >0.9	4	7 ± 3	16	65 ± 1	34 ± 4	18.9	127 ± 9
Feature By itself	1	5 ± 3	0.8	2 ± 1	25 ± 3	17.8	33 ± 2
Corr >0.8	14	41 ± 6	1.5	59 ± 5	149 ± 8	18.8	91 ± 8
Corr >0.8 with VIF	6	8 ± 4	1.5	5 ± 1	36 ± 4	27.2	55 ± 2
Location-based	11	5 ± 2	2.3	38 ± 2	86 ± 17	18.9	59 ± 8
Location-based with VIF	8	14 ± 6	2.5	31 ± 2	106 ± 11	23.9	61 ± 15
SHAP Values	2	5 ± 1	0.7	11 ± 1	24 ± 1	17.4	35 ± 1

In general, the reduced standard deviation for the RMSE indicates that utilizing VIF values to remove multicollinearity invariably improves model consistency.

The 1-hour-ahead predictions made by LSTM, SVR, and RF using SHAP-determined inputs can been seen in Figure 3. 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 models' estimate. LSTM and SVR capture the pump temperature trends very well, as seen by the low mean RMSE and MAE. For LSTM and SVR, the SHAP-determined inputs produced the best results with respect to the mean RMSEs and MAEs 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 begins to crop up. The outliers for the 1-hour-ahead prediction using RF can be seen in Figure 3. 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.

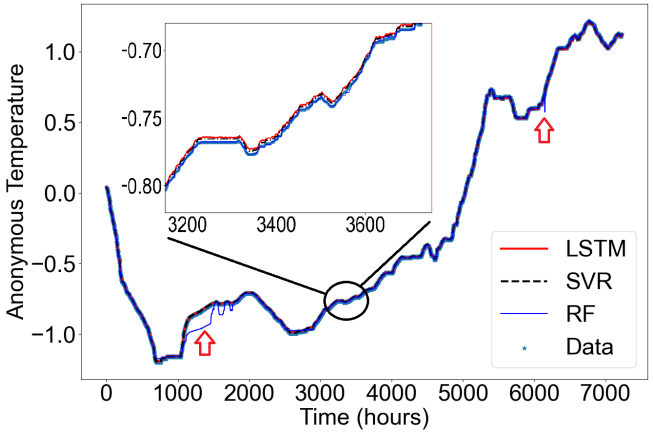

Figure 3. Predictions of pump temperature 1 hour ahead, using the SHAP-determined input to LSTM, SVR, and RF models.

Each of the feature sets in Tables 4 and 5 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 single-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 that 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 only the feature itself, and once again demonstrated poor performance when using highly correlated variables. It appears RF has a more difficult experience combining information contained within multiple variables for regression.

To better compare the results from the 1- and 24-hour-ahead predictions, LSTM, SVR, and RF results using SHAP-determined inputs are given in Figure 4. Unlike the previous 1-hour-ahead predictions, the 24-hour-ahead predictions are not as accurate, as seen by the variability in the zoomed-in portion. Larger changes in bearing temperatures can occur over a 24-hour period than over a 1-hour period.

The FCS in this plant contained three loops. A schematic of BWR system with a FCS system is show in Figure 5. These loops (i.e., CP, CBP, heaters, and feed pumps) operate in 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 6. 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.

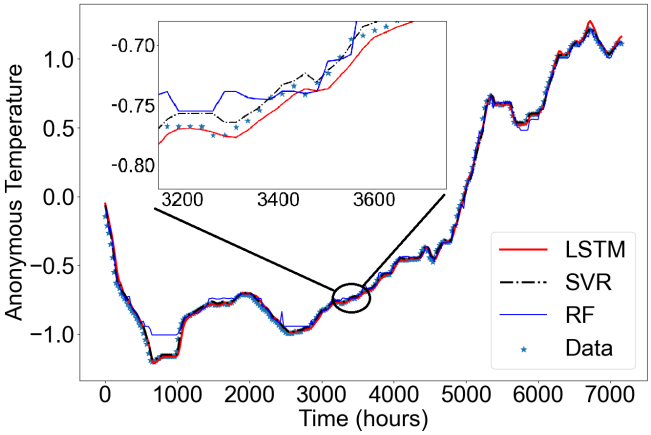

Figure 4. Predictions of pump temperature 1 day ahead, using the SHAP-determined input to LSTM, SVR, and RF models.

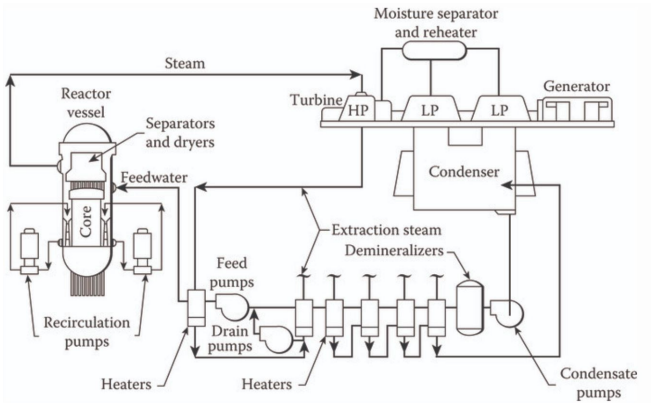

Figure 5. Schematic of a BWR system (Theriault, 2016).

Table 6. 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

4.4. Discussion

LSTM, SVR, and RF each showed comparable results for the 1-hour-ahead predictions when the optimal input features were selected. Although the feature by itself provided adequate results for the RF, the same results were not seen by the other models. With a different dataset in a more complicated setting (e.g., non-steady-state operation or under faulted conditions), additional features could provide additional insights. 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 input or 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 takes into consideration 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. CONCLUSION AND FUTURE WORK

The proposed data cleaning and feature selection process can be applied to many applications beyond the prediction of features within nuclear infrastructure. As big data and continuous online monitoring become more widely available, relevant features will need to be extracted to maximize the forecasting ability of these models. Proper and necessary steps should be implemented to ensure that the data are appropriately sanitized before use. The forecasting models can then be used to estimate future conditions, allowing for optimal operating strategies and/or more efficient maintenance scheduling. However, correlation does not always relate to causation. One limitation of the proposed methods, especially SHAP, is that the explanation of the feature is based on the correlation of the feature to the model's output. This correlation implies causation, but such may not always be the case. Further investigation would be required to examine the plausibility of the causation.

This research detailed a framework for data cleaning and feature selection that was then applied to several ML techniques to predict future feature conditions within a BWR FCS. The most accurate models were those trained on optimal inputs. In the future, we will look to expand these predictions to encompass not only steady-state operation but also non-steady operations such as derates and refueling ramp-downs. Additional contributions to the data cleaning and feature selection process may need to be taken into consideration, as the system would now undergo larger—and sometimes more abrupt—variations in operating conditions.

6. ACKNOWLEDGMENTS

This project was made possible through funding by the U.S. Department of Energy's Office of Nuclear Energy, under the Nuclear Energy Enabling Technologies Program. We are grateful to the plant engineers for their technical discussions on the system and for providing the data.

REFERENCES

Akinwande, M. O., Dikko, H. G., & Samson, A. (2015). Variance Inflation Factor: As a Condition for the Inclusion of Suppressor Variable(s) in Regression Analysis. *Open Journal of Statistics*, *05*(07), 754–767. doi: 10.4236/ojs.2015.57075

Alzubi, J., Nayyar, A., & Kumar, A. (2018). Machine Learning from Theory to Algorithms: An Overview. *Journal of Physics: Conference Series*, *1142*(1). doi: 10.1088/1742-6596/1142/1/012012

Atamuradov, V., Medjaher, K., Dersin, P., Lamoureux, B., & Zerhouni, N. (2017). Prognostics and Health Management for Maintenance Practitioners - Review, Implementation and Tools Evaluation. *International Journal of Prognostics and Health Management*, *8*.

Bechhoefer, E., Schlanbusch, R., & Waag, T. I. (2016). Techniques for large, slow bearing fault detection. *International Journal of Prognostics and Health Management*, *7*(1), 1–12. doi: 10.36001/ijphm.2016.v7i1.2358

Booth, A. L., Abels, E., & McCaffrey, P. (2021). Development of a prognostic model for mortality in COVID-19 infection using machine learning. *Modern Pathology*, *34*(3), 522–531. doi: 10.1038/s41379-020-00700-x

Cabrera, D., Guamán, A., Zhang, S., Cerrada, M., Sánchez, R. V., Cevallos, J., ... Li, C. (2020). Bayesian approach and time series dimensionality reduction to LSTM-based model-building for fault diagnosis of a reciprocating compressor. *Neurocomputing*, *380*, 51–66. doi: 10.1016/j.neucom.2019.11.006

Chornovol, O., Kondratenko, G., Sidenko, I., & Kondratenko, Y. (2020). Intelligent forecasting system for NPP's energy production. *Proceedings of the 2020 IEEE 3rd International Conference on Data Stream Mining and Processing, DSMP 2020*, 102–107. doi: 10.1109/DSMP47368.2020.9204275

Daoud, J. I. (2018). Multicollinearity and Regression Analysis. *Journal of Physics: Conference Series*, *949*(1).

doi: 10.1088/1742-6596/949/1/012009

Davò, F., Alessandrini, S., Sperati, S., Delle Monache, L., Airoldi, D., & Vespucci, M. T. (2016). Post-processing techniques and principal component analysis for regional wind power and solar irradiance forecasting. *Solar Energy*, *134*, 327–338. doi: 10.1016/j.solener.2016.04.049

Diez-Olivan, A., Del Ser, J., Galar, D., & Sierra, B. (2019). Data fusion and machine learning for industrial prognosis: Trends and perspectives towards industry 4.0. *Information Fusion*, *50*, 92–111.

Drucker, H., Burges, C. J., Kaufman, L., Smola, A., Vapnik, V., et al. (1997). Support vector regression machines. *Advances in neural information processing systems*, *9*, 155–161.

Farzad, A., Mashayekhi, H., & Hassanpour, H. (2019). A comparative performance analysis of different activation functions in lstm networks for classification. *Neural Computing and Applications*, *31*(7), 2507–2521.

Godwin, J. L., & Matthews, P. (2013). Classification and detection of wind turbine pitch faults through scada data analysis. *IJPHM Special Issue on Wind Turbine PHM*, 90.

Gohel, H. A., Upadhyay, H., Lagos, L., Cooper, K., & Sanzetenea, A. (2020). Predictive maintenance architecture development for nuclear infrastructure using machine learning. *Nuclear Engineering and Technology*, *52*(7), 1436–1442. doi: 10.1016/j.net.2019.12.029

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, *28*(10), 2222–2232.

Hall, M. A., & Smith, L. A. (1998). Practical feature subset selection for machine learning.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8).

Hong, C. W., Lee, C., Lee, K., Ko, M. S., & Hur, K. (2020). Explainable artificial intelligence for the remaining useful life prognosis of the turbofan engines. *Proceedings of the 3rd IEEE International Conference on Knowledge Innovation and Invention 2020, ICKII 2020*(1), 144–147. doi: 10.1109/ICKII50300.2020.9318912

Jain, R. K., Smith, K. M., Culligan, P. J., & Taylor, J. E. (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. *Applied Energy*, *123*, 168–178.

Karasu, S., Altan, A., Bekiros, S., & Ahmad, W. (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. doi: 10.1016/j.energy.2020.118750

Kong, W., Dong, Z. Y., Jia, Y., Hill, D. J., Xu, Y., & Zhang, Y. (2017). Short-term residential load forecasting based on lstm recurrent neural network. *IEEE Transactions on Smart Grid*, *10*(1), 841–851.

Li, R., Verhagen, W. J., & Curran, R. (2019). Comparison of data-driven prognostics models: A process perspective. In *29th european safety and reliability conference.*

Lundberg, S. M., Erion, G. G., & Lee, S. I. (2018). Consistent individualized feature attribution for tree ensembles. In *31st conference on neural information processing systems (nips 2017).* Long Beach, CA, USA.

Lundberg, S. M., & Lee, S. I. (2017a). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, *2017-Decem*(Section 2), 4766–4775.

Lundberg, S. M., & Lee, S.-I. (2017b). A unified approach to interpreting model predictions. In I. Guyon et al. (Eds.), *Advances in neural information processing systems 30* (pp. 4765–4774). Curran Associates, Inc.

Mangalathu, S., Hwang, S. H., & Jeon, J. S. (2020). Failure mode and effects analysis of RC members based on machine-learning-based SHapley Additive exPlanations (SHAP) approach. *Engineering Structures*, *219*(May), 110927. doi: 10.1016/j.engstruct.2020.110927

Marcilio, W. E., & Eler, D. M. (2020). From explanations to feature selection: Assessing SHAP values as feature selection mechanism. *Proceedings - 2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2020*, 340–347. doi: 10.1109/SIBGRAPI51738.2020.00053

Monirul Kabir, M., Monirul Islam, M., & Murase, K. (2010). A new wrapper feature selection approach using neural network. *Neurocomputing*, *73*(16-18), 3273–3283. doi: 10.1016/j.neucom.2010.04.003

Moon, J., Kim, Y., Son, M., & Hwang, E. (2018). Hybrid short-term load forecasting scheme using random forest and multilayer perceptron. *Energies*, *11*(12), 1–20. doi: 10.3390/en11123283

Müller, I. M. (2021). Feature selection for energy system modeling: Identification of relevant time series information. *Energy and AI*, *4*, 100057. doi: 10.1016/j.egyai.2021.100057

Niu, T., Wang, J., Lu, H., Yang, W., & Du, P. (2020). Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting. *Expert Systems with Applications*, *148*, 113237. doi: 10.1016/j.eswa.2020.113237

NRC. (1998). BWR/4 Technology Manual (R-104B). *Ml022830867*, 1–442.

O'Brien, R. M. (2007). A caution regarding rules of thumb for variance inflation factors. *Quality and Quantity*, *41*(5), 673–690. doi: 10.1007/s11135-006-9018-6

Ozturk, T., Talo, M., Azra, E., Baran, U., & Yildirim, O. (2020). Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Computers in Biology and Medicine*(January).

Pan, F., Yang, L., Li, Y., Liang, B., Li, L., Ye, T., ... Zheng, C. (2020). Factors associated with death outcome in patients with severe coronavirus disease-19 (Covid-19): A case-control study. *International Journal of Medical Sciences*, *17*(9), 1281–1292. doi: 10.7150/ijms.46614

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pham, L., Luo, L., & Finley, A. (2020). Evaluation of Random Forest for short-term daily streamflow forecast in rainfall and snowmelt driven watersheds. *Hydrology and Earth System Sciences Discussions*(June), 1–33. doi: 10.5194/hess-2020-305

Pokharel, S., Sah, P., & Ganta, D. (2021). Improved prediction of total energy consumption and feature analysis in electric vehicles using machine learning and shapley additive explanations method. *World Electric Vehicle Journal*, *12*(3). doi: 10.3390/wevj12030094

Remeseiro, B., & Bolon-Canedo, V. (2019). A review of feature selection methods in medical applications. *Computers in Biology and Medicine*, *112*(July), 103375. doi: 10.1016/j.compbiomed.2019.103375

Salcedo-Sanz, S., Cornejo-Bueno, L., Prieto, L., Paredes, D., & García-Herrera, R. (2018). Feature selection in machine learning prediction systems for renewable energy applications. *Renewable and Sustainable Energy Reviews*, *90*(March), 728–741. doi: 10.1016/j.rser.2018.04.008

Sendlbeck, S., Fimpel, A., Siewerin, B., Otto, M., & Stahl, K. (2021). Condition monitoring of slow-speed gear wear using a transmission error-based approach with automated feature selection. *International Journal of Prognostics and Health Management*, *12*(2), 1–15. doi: 10.36001/IJPHM.2021.V12I2.3026

Shahidi, P., Maraini, D., & Hopkins, B. (2020). Railcar Diagnostics Using Minimal-Redundancy Maximum-Relevance Feature Selection and Support Vector Machine Classification. *International Journal of Prognostics and Health Management*, *7*(4), 1–13. doi: 10.36001/ijphm.2016.v7i4.2524

Song, F., Guo, Z., & Mei, D. (2010). Feature selection using principal component analysis. *Proceedings - 2010 International Conference on System Science, Engineering Design and Manufacturing Informatization, ICSEM 2010*, *1*, 27–30. doi: 10.1109/ICSEM.2010.14

Sthle, L., & Wold, S. (1989). Analysis of variance (ANOVA). *Chemometrics and Intelligent Laboratory Systems*, *6*(4), 259–272. doi: 10.1016/0169-7439(89)80095-4

Theriault, K. (2016). Boiling Water Reactors. In *Nuclear engineering handbook.* CRC Press. (ISBN: 10.1201/9781315373829-5)

Yildirim, H., & Revan Özkale, M. (2019). The performance of ELM based ridge regression via the regularization parameters. *Expert Systems with Applications*, *134*, 225–233. doi: 10.1016/j.eswa.2019.05.039

Yu, J., Hong, B., Park, J. Y., Hwang, J. H., & Kim, Y. K. (2021). Impact of Prognostic Nutritional Index on Postoperative Pulmonary Complications in Radical Cystectomy: A Propensity Score-Matched Analysis. *Annals of Surgical Oncology*, *28*(3), 1859–1869. doi: 10.1245/s10434-020-08994-6

Zebari, R., Abdulazeez, A., Zeebaree, D., Zebari, D., & Saeed, J. (2020). A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction. *Journal of Applied Science and Technology Trends*, *1*(2), 56–70. doi: 10.38094/jastt1224

Zhang, Y., Peng, Z., Guan, Y., & Wu, L. (2021). Prognostics of battery cycle life in the early-cycle stage based on hybrid model. *Energy*, *221*, 119901. doi: 10.1016/j.energy.2021.119901