Maciej PAWEŁCZYK
Szymon FULARA
Marzia SEPE
Alessandro DE LUCA
Maciej BADORA

INDUSTRIAL GAS TURBINE OPERATING PARAMETERS MONITORING AND DATA-DRIVEN PREDICTION

MONITOROWANIE ORAZ BAZUJĄCA NA DANYCH PREDYKCJA PARAMETRÓW ROBOCZYCH PRZEMYSŁOWEJ TURBINY GAZOWEJ

The article reviews traditional and modern methods for prediction of gas turbine operating characteristics and its potential failures. Moreover, a comparison of Machine Learning based prediction models, including Artificial Neural Networks (ANN), is presented. The research focuses on High Pressure Compressor (HPC) recoup pressure level of 4th generation LM2500 gas generator (LM2500+G4) coupled with a 2-stage High Speed Power Turbine Module. The researched parameter is adjustable and may be used to balance net axial loads exerted on thrust bearing to ensure stable gas turbine operation, but its direct measurement is technically difficult implicating the need to indirect measurement via set of other gas turbine sensors. Input data for the research have been obtained from BHGE manufactured and monitored gas turbines and consists of real-time data extracted from industrial installations. Machine learning models trained using the data show less than 1% Mean Absolute Percentage Error (MAPE) as obtained with the use of Random Forest and Gradient Boosting Regression models. Multilayer Perceptron Artificial Neural Networks (MLP ANN) models are reviewed, and their performance checks inferior to Random Forest algorithm-based model. The importance of hyperparameter tuning and feature engineering is discussed.

Keywords: *gas turbine, machine learning, data-driven prediction, HP recoup pressure analysis.*

W artykule przedstawiono przegląd klasycznych i aktualnych metod przewidywania parametrów operacyjnych oraz potencjalnych usterek turbin gazowych. Dodatkowo zaprezentowano porównanie wybranych modeli opartych o uczenie maszynowe, w tym modeli wykorzystujące sztuczne sieci neuronowe. Przeprowadzone badania dotyczyły analiz poziomu ciśnienia ze sprężarki turbiny gazowej LM2500 czwartej generacji (LM2500+G4) połączonej z dwustopniową turbiną roboczą. Badany parametr podlega sterowaniu i może posłużyć do wyrównania sił osiowych działających na łożysko główne wału wysokiego ciśnienia w celu zapewnienia stabilnej i niezawodnej pracy turbiny gazowej. Jednocześnie jego bezpośredni pomiar jest kosztowny stąd potrzeba dokonania pośredniego pomiaru z wykorzystaniem innych czujników zamontowanych na turbinie. Dane wejściowe do analiz otrzymano dzięki uprzejmości producenta turbin, firmy BHGE. Zawierają one parametry bezpośrednio pobrane z monitorowanych turbin gazowych. Modele uczenia maszynowego otrzymane w wyniku analizy charakteryzują się średnim błędem procentowym (MAPE) na poziomie poniżej 1%. Najmniejszym błędem charakteryzują się modele otrzymane przy zastosowaniu lasów losowych (Random Forest) oraz gradientowego wzmacniania regresji (Gradient Boosted Regression). Przetestowano także zastosowanie wielowarstwowych, w pełni połączonych sztucznych sieci neuronowych, których efektywność okazała się niższa od modelu opartego o algorytm lasów losowych. W podsumowaniu podkreślono wagę dostosowywania hiperparametrów i inżynierii cech.

Słowa kluczowe: *turbina gazowa, uczenie maszynowe, predykcja bazująca na danych, analiza pomiarów ciśnienia sprężarki.*

1. Introduction

Operational reliability of complex mechanical energy generation system is a key in assuring stable and cost-effective power supply for long-term commercial, industrial and communal purposes. One of the key systems used to deliver energy is currently, and for the foreseeable future, gas turbine. Gas turbine in-service monitoring proved to be useful in potential failure diagnosis and prevention as well as in its emission monitoring [36]. Appropriate gas turbine utilization, for example in de-rated temperature mode, may have significant contribution to decrease in harmful exhaust emissions [20] as well as to extend reliable operation of the system [1]. In this article accumulated gas turbine operating service data has been studied to predict key parameters.

Prediction of gas turbine performance is one of the key factors researched. In-service performance deterioration and system unreliability are significant contributors to a gas turbine utilization planning. System failure or even single percent of power reduction can result in significant impact throughout life cycle. Overhaul is the optimum remedy to restore desired parameters, but for the planning purposes, it should also be modelled and postponed if possible. Moreover, performance deterioration (and system failure) proved to be difficult to monitor and predict using installed gas turbine due to sensor accuracy, instrumentation aging, assembly constraints and control challenges. Therefore, maintenance planning should be based on multiple parameter's analysis and results integration using specialized model [12].

There are several typical examples of monitoring systems, that have been studied in the past. Historically, standard gas turbine tests

have been used to create performance maps and model typical gas turbine behavior. Instrumentation proved to be one of the most important challenges, limiting data acquisition capability [11]. Gas turbine integrity issues, cost of engine testing and limited acquisition capabilities are responsible for this restraint [35]. Moreover, tests of the heavyweight gas turbines (75 MW and more) are not carried out before gas turbine delivery due to cost and assembly reasons.

Multiple statistical methods have been developed over time to preview, model and decrease gas turbine life cycle cost. Traditional analytical methods include, among others: Failure Mode and Effects Analysis (FMEA), Fault Tree Analysis (FTA), and Markov Analysis. An example of the Failure Mode and Effects Analysis (FMEA) supplemented by statistical failure rate model has been discussed by Mingazov and Korobitsin [29]. The results of such model application may allow to assess different gas turbine systems reliability. Non-traditional reliability monitoring methods and novel approaches to reliability monitoring include vague lambda-tau methodology [23], fuzzy sets [37] and numerical assessment based on Piecewise Deterministic Markov Process and Quasi Monte Carlo methods [10]. The Petri Net model fed by Failure Tree Analysis (FTA) has also been presented and discussed by Verma and Kumar [35]. To address system unreliability, the fuzzy sets application has been studied by several authors, example of which is described by Huang et all [18]. The results obtained using these methods are promising, however they also require rough estimation of expected reliability to proactively choose appropriate modelling method. The dynamic nature of gas turbine can also be taken into consideration. The power generation system deteriorates over time and thus its reliability is time dependent. Binary Decision Diagram (BDD) has been proposed to address a multi-phase network system as discussed by Lu et al. [26]. The industrial gas turbine system operation may also be considered as phased-mission system (PMS) since their operation mode usually consists of numerous repetitive phases. Therefore, a flexible truncation limit may be applied to BDD problem explosion. While the truncation application is necessary to model increasing number of phases, the flexible truncation allows retention of the truncation error at low level as presented by Lu et al. in [27].

The statistical-based analytical models are the most frequently used methods for Engine Health Monitoring (EHM). Trend analysis is one of the standard methods used to compare current gas turbine parameters with warning levels [32]. Pattern recognition can be employed in limited time frames (or sliding windows) to detect gas turbine behavior anomaly and potential failure. Kalman filters are applied to define parameter changes that result in least square error. Artificial Neural Networks (ANN), support vector machines (SVM) and particle swarm optimization (PSO) techniques are also utilized to improve the accuracy of the predictions [6,7,16,21,33]. It has been proven that a hybrid PSO-SVM based model can result in a regression accuracy of approximately 95% [13]. It is also worth noting that described methods can achieve top results only when applicable dataset is available in vast quantities for multiple sensors (implicating Big Data solutions). In general, it can be concluded that the physical models derived from full or partial operating parameters provide the most accurate results [22, 24].

As discussed, performance optimization and hardware ageing modelling are crucial to maintain high efficiency of the propulsion system and to monitor health condition of the critical components. Usually, only selected parameters are remotely monitored. The limited data acquisition requirement allows minimization of costs associated with the installation of additional sensors in hardly accessible turbine locations. In addition, the benefit of installing advanced sensors to monitor system performance and components health status must be balanced

against the risk of gas turbine downtime due to sensor malfunctions and potential secondary damages of gas turbine components caused by sensor failure itself. A scenario of gas turbine life-cycle optimization with several operational parameters used as input to advanced models estimating online gas turbine performance and components ageing has been presented by Baker Hughes team [28]. The team concluded that algorithm's prediction accuracy together with data quality and proper expertise are important to model accurately long-term gas turbine behavior and to provide correct maintenance insights to on-site operators [28].

To ensure consistently high accuracy required, data exploration of each input and corresponding data processing is required [17]. While main gas turbine control parameters are redundant and transducer outputs are processed by the control software according to robust selection criteria and technical board approved logic, some parameters are obtained directly from a single sensor and hence a predictive modelling may help to create a baseline for consistency check.

2. Presentation of the problem

This paper discusses the use of machine learning (ML) algorithms to estimate chosen parameter (in this case, the secondary flow pressures, particularly the HP recoup pressure) with respect to the main gas turbine control parameters in order to troubleshoot instrumentation anomalies or detect deviations from the expected baseline. The primary goal is to predict long-term HP recoup pressure and hence enable consistent performance prediction and potential early detection of components degradation.

The HP recoup pressure is an adjustable parameter, that enables balancing of net axial loads exerted on thrust bearing and ensures operation within desired threshold. The control method and applicable system have been described and patented by Badeer [2].

The presented research is based on data obtained from 4th generation LM2500 gas generator (LM2500+G4) coupled with a 2-stage High Speed Power Turbine Module GE Challenge competition [19]. A cross section of LM2500 gas turbine is shown on Fig. 1. The LM2500 is an industrial derivative of the General Electric (GE) CF6 aircraft engine (engine originally developed for aviation purposes and then refurbished for stationary operation) [3, 14]. This study focuses on PGT25+G4 gas turbine delivering 34 MW with thermal efficiency of 41% [5].

The presented study shows analytical comparison of the gas turbine engine pressure parameter prediction with respect to other operational (numerical) and geometrical (categorical) parameters. Categorical parameter is represented by separately provided pressure orifice size, available only for limited number of gas turbines. The orifice plate is built in the HP Recoup pressure line to allow flow rate measurement and is causing an irrecoverable pressure loss [30]. The
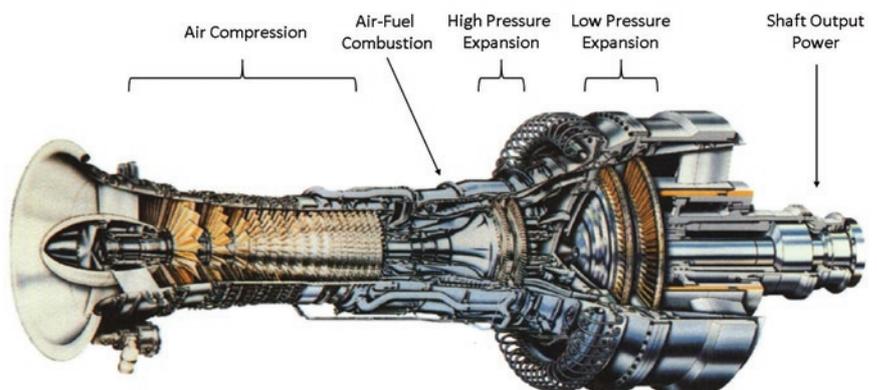

Fig. 1. Researched industrial & marine gas turbine configuration, based on [8]

potential impact of the orifice size on the researched HP Recoup pressure parameter prediction has been also studied.

3. Mathematical modelling and models implementation

Since turboshaft engine consists a close-loop system, by design, researched recoup pressure is mutually depended on other turbine engine parameters. This feature makes it suitable for multistep prediction, as the best model can be identified in such condition [31].

3.1. Mathematical models

Typical starting point for all machine learning challenges are linear models. This include linear regression (logistic regression in instance of models with categorical values included) with linear coefficients $\kappa = (\kappa_1, \kappa_2, \ldots, \kappa_n)$ and biases assigned to each of the input variables to minimize the output sum of squares between the training labels and predictions via linear approximation:

$$\hat{\kappa} = \min_{\kappa} X\kappa - y_2^2 \qquad (1)$$

where $X\kappa = y$ is linear equation with coefficient κ and $\hat{\kappa}$ is predicted value.

While continuous numerical inputs for the model are normalized (as discussed later in this article) to prevent data range discrepancies from affecting the resultant model, additional options are available to prevent high coefficients from being assigned to selected parameters, while keeping the others small (and effectively eliminating them from the prediction process). This is implemented by imposing additional penalty parameter on the size of coefficient in the cost function calculation process. For ridge regression (also known as Tikhonov based regression or L2) the penalty is assigned to a sum of squared weights. For LASSO (Least Absolute Shrinkage and Selection Operator or L1) regression penalty value is only assigned to the sum of absolute values of the model weights (which can allow removal of redundant feature from the model), while elastic net model allows utilization of both routines at the same time. This process forces linear model weight to remain small, which typically leads to less overfitting [34]. The penalty value is tuned to obtain minimum testing set prediction error:

$$\hat{\kappa} = \min_{\kappa} \frac{1}{2m_j} \|X\kappa - y\|_2^2 + \alpha \|\kappa\|_1 \qquad (2)$$

where m_j is a number of samples, α is constant and $\|\kappa\|_1$ is ℓ_1-normal to coefficient κ.

LASSO/ridge/elastic net CV models have additional cross validation routines built into them to automatically split the entire dataset into all possible train/test combinations to allow model generalization on the entire dataset, while maintaining dataset split. This happens on the expense of model performance as cross validation split of 5 will require training of 5 models, rather than one. LassoLarsIC is computationally cheaper alternative to this process.

Bayesion Ridge Regression models provides a probabilistic estimate for a regression problem. The model utilizes multiple parameters to obtain similar regularization pattern to ridge regression. The parameters are estimated by maximizing the marginal log likelihood. Bayesian Ridge Regression provides a different weight set than those obtained using Ordinary Least Squares method:

$$p(\kappa|\lambda) = \mathcal{N}\left(\kappa|0, \lambda^{-1}\mathbb{I}_p\right) \qquad (3)$$

where λ is Gaussian distribution for probabilistic model of output parameter κ.

Kernel models allow for dataset dimensionality modification/reduction, which allows model to create an artificial hyperspace split between model parameters, which can result in better model performance. This method can be applied on models such as Support Vector Machine (SVM), which can be used for either classification (Support Vector Classifier or SVC) or regression (Support Vector Regression) tasks. SVM works by finding a set of optimum linear split in high dimensional space, using a subset of training points in the decision function (support vectors). While different kernels can be used to achieve optimum model performance nonlinear kernels are computationally expensive, especially for large datasets.

Random Forest Model consists of multiple decision trees (with exact decision tree numbers determined based on hyperparameter tuning), which are averaged to classify target feature properly. During construction of each tree a node split is randomly generated to best fit random sub-sample of the features. The randomness results in increase of forest classifier bias, but, due to averaging, the model variance also decreases. This compensates the adverse effect leading to higher quality prediction. The algorithm processes original dataset by proposing a randomly generated shallow trees, usually bootstrapping (randomly sampling with replacement) original dataset for to augment original training dataset. Once trained, a prediction is made for every tree in the forest and individual estimates are combined to achieved optimum testing set performance. For regression problem (no categorical feature included) the overall result is a mean of individual predictions. For classification problem (orifice size included) each tree provides a weighted confidence for each class, which are then averaged across all trees. The result is the class with the highest confidence to minimize general mean-squared generalization error L for the numerical prediction $\hat{y}_i$ and its corresponding true label y_i:

$$L = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \qquad (4)$$

The binary nature of the Random Forest Model can be partially compensated by iteratively reducing the error rate for each of the created trees, by building regression tree on negative gradient of a loss function, quite similarly to neural network training process. This method is called gradient boosted decision trees (GBDT) [25,38]. It works by processing a small number of decision trees (which in this instance are base learners) and using a learning rate to adjust tree definition to minimize aggregation of the loss function $L(y_i, F(x_i))$, by reducing individual tree error:

$$T^* = arg \min_{T} \sum_{i=1}^{N} L(y_i, F(x_i)) \qquad (5)$$

Artificial Neural Networks (ANN) model training process, often called Deep Learning (for 3 hidden layers or more). Typical neural network relies on a predefined input and output vector and inner stacked layers utilizing mathematical transformations of input data as shown on Fig. 2. ANNs work by processing input vector through a set of linear functions followed by nonlinear activation (recently almost popular being Rectified Linear Unit or ReLU) in a process called feed forward propagation. This is followed by loss (for a single batch) and cost (for entire dataset) value computation based on predefined cost function. Then, for each weight, a gradient with respect to specific

parameter is computed using a chain rule and cost value. Finally, all weights are updated sequentially using process called backpropagation.
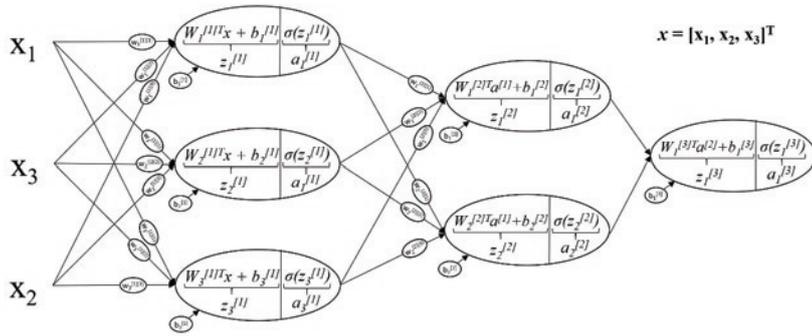

Fig. 2. Overview of Artificial Neural Network (ANN) densely connected architecture with 2 hidden layers

Considering that deep neural networks, especially convolutional neural networks for image processing, (utilizing a combination of convolutional, pooling and dense neural networks) usually consist of 10+ layers, it is useful to represent given neural network via high level representation. The example of 3 hidden layer network with 4 nodes in each layer, 3 inputs and 2 outputs has been shown on Fig. 3.
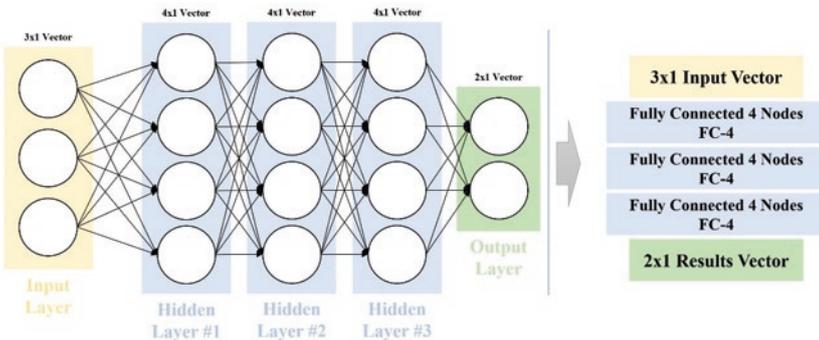

Fig. 3. An example high level representation on densely connected neural network. For the purpose of analyzed research 3 hidden layer network with up to 20 inputs, up to 100 nodes for each layer and 1 output has been used

This high level, generalized representation is often used in Deep Learning frameworks such as Keras, PyTorch. While Scikit-learn offers simple neural network implementations, those frameworks, while working on top of low level backends (Tensorflow), offer much more flexibility and significant reduction in training and inference (model implementation) time, typically due to effective processing routines for direct implementation on CPU/GPU (and recently, TPU).

3.2. Models Implementation

Models obtained in the following analysis were developed using Python 3.6, modules such as Numpy/Pandas (data processing), Scikit-learn (machine learning) and Keras/Tensorflow (Artificial Neural Networks) frameworks. Model was iteratively developed, with certain analysis thresholds established through set of tests.

In order to properly establish acceptability threshold for model's results a benchmark method has been proposed. It consists of three metrics used to compare results of applied machine learning models: Mean Absolute Percentage Error (MAPE), Root-Mean-Square Error (RMSE) and Coefficient of Determi-

nation (R2). The three parameters have been chosen as they represent a basis for qualitative and quantitative comparison of the researched methods. MAPE metric allows valuable insight into the absolute magnitude of the error, while RMSE provides additional intuition into scaled residual size. R2 score represents a statistical measurement of model fit into predicted dataset. Target feature mean has been assumed as baseline prediction, resulting in RMSE of 0.83, MAPE of 9.24% and R2 score of 0 (per R2 score definition).

The first step in predictive modelling is data exploration, which allowed to limit the number of features used as predictive maintenance system input. Data cleaning pipeline implementation allowed dataset feature size reduction from 4568 columns to 382. Any parameter with more than 3% missing data has been removed. For remaining parameters missing data has been replaced with median for all other features. Additionally, all outliers (exceeding +/− 6 Sigma deviation) were removed (entire data row removal) to prevent errors further on in the analysis.

Furthermore, categorical geometry feature – flow orifice size – has been provided and added to the dataset. Subsequent analysis has shown that geometry have significant predic-
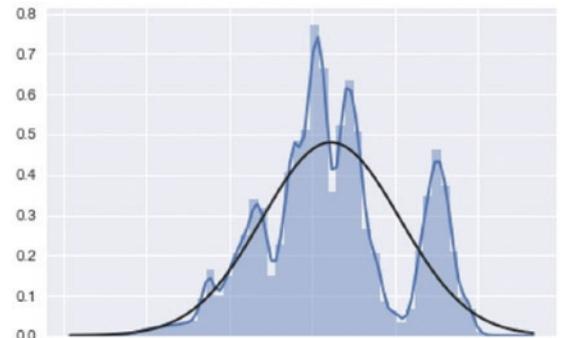

Fig. 4. Distribution of the target feature (gas turbine pressure) with fitted normal distribution. As input dataset (after initial preprocessing) consisted of multiple serial numbers, the resultant dataset is multinomial. No ensemble subgrouping has been used due to sufficient model predictive capacity
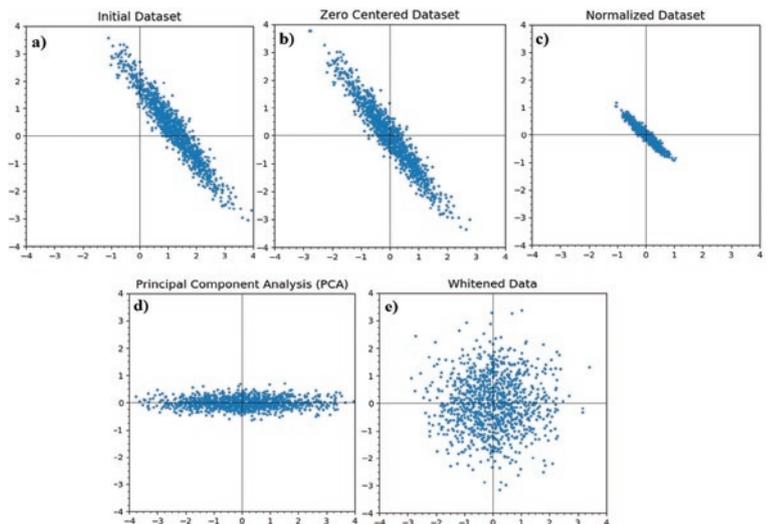

Fig. 5. Data normalization techniques: a) N=2000 normally distributed input dataset; b) centering procedure applied on the dataset (dimensional mean subtracted from each datapoint); c) Standard deviation-based normalization of centered data; d) Principal Component Analysis (PCA) applied on the initial dataset; e) Whitening transformation applied on input dataset

tive power and as such all rows, where geometrical data was not available, were dropped. Resultant target feature (pressure) distribution is shown on Fig. 4.

High predictive power of the model has been ensured by ranking remaining features based on Pearsons's correlation coefficient calculated between each of 380 analyzed parameters and target feature. Feature independence has been ensured by analyzed the relationship between pairs and picking only sensors, which would allow physics-based explanation and higher correlation value with target feature. This approach has been chosen (instead of proceeding with typical correlation tests like Pearson's or Chi-Squared) to streamline dataset reduction through project approval board. This allowed reduction of input dataset to 45 features. As such remaining 45 features were additionally tested with random forest feature estimator. Combined feature mark considered both regression correlation and random forest estimator mark. Best 20 features were used in the further analysis and combined with geometry categorical feature. Random 5% of the dataset were retained as model test set.

Input data set could be normalized in order to allow more effective computation paradigm and better convergence for neural network processing. Typical normalization steps consist of combination of zero-centering/normalization. Similarly, Principal Component Analysis (PCA) is used as an efficient way of both data normalization and dimensionality reduction. Finally, data whitening transformation ensures that mean data mean value is concentric with coordinate system origin point as well as that all the axis in the dataset have the same variance. These normalization methods have been presented on Fig. 5. Due to variable nature of turbine operating parameters input dataset has not been normalized. This approach has been tested on validation dataset and has shown more consistent results then normalizing input data.

For each of the predefined model combination data has been preprocessed to better fit specified model, then a test/train split has been created using random number seed and test set size of 0.25 (25% of the dataset was retained for testing) followed by classifier training, model storage and result view for postprocessing. Example of Linear Regression model implementation is shown on Figure 6. It shows Linear Regression model implementation.

```python
modelNames = [["Linear Regression With Orifice", "WO", "sklearnRegression"],
              ["Linear Regression with No Orifice", "NO", "sklearnRegression"]]

for modelNameIndex, (modelName, orificeStatus, saveName) in enumerate(modelNames):

    print("{}/{} Analyzing model '{}' with orifice status: {}".format(modelNameIndex+1, len(modelNames), modelName, orificeStatus

    # Copy dataframe and drop useless columns:
    LR_columns = df.columns.drop("DateTime")
    if orificeStatus=="NO":
        LR_columns = LR_columns.drop("OrificeSize")
    LR_df = df[LR_columns]
    LR_columns_validation = df_validation.columns.drop("DateTime")
    if orificeStatus=="NO":
        LR_columns_validation = LR_columns_validation.drop("OrificeSize")
    LR_df_validation = df_validation[LR_columns_validation]

    # Randomly shuffling the data (usually a good idea):
    np.random.seed(33)
    LR_df = LR_df.reindex(np.random.permutation(LR_df.index))
    LR_df.reset_index(inplace=True, drop=True)

    if orificeStatus=="WO":
        orifice = encode_text_index(LR_df, "OrificeSize")
        orificeValidation = encode_text_index(LR_df_validation, "OrificeSize")

    # Encode to a 2D matrix for training
    x,y = to_xy(LR_df, targetFeature)
    xValidation,yValidation = to_xy(LR_df_validation, targetFeature)

    # Split into train/test
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.25, random_state=45)

    # Create linear regression:
    regressor = sklearn.linear_model.LinearRegression()

    # Fit/train linear regression:
    regressor.fit(x_train,y_train)

    # Predict:
    pred = regressor.predict(x_test)
    predValidation = regressor.predict(xValidation)

    # Saving model description and its score to the main array:
    currentAnalysisResult = provideAnalysisResults(pred, y_test, predValidation, yValidation)
    modelSummary.append(currentAnalysisResult)
    print(currentAnalysisResult)

    # Saving the model to disk:
    modelFileName = ("{}\\{}{}.sav".format(folderForTheAnalysis, saveName, orificeStatus))
    pickle.dump(regressor, open(modelFileName, "wb"))

    plt.clf() # clearing all the plots from previous analysis
    # Plot the chart:
    expandedRegressionChart(pred, y_test, rollingWindowWidth=100)

    # Remove target feature from the array with all the column names:
    names = list(LR_df.columns.values)
    names.remove(targetFeature)

    # Plot regresssion coefficient importance:
    report_coef(names, regressor.coef_[0,:], regressor.intercept_)
    plt.show()
```

Fig. 6. Example of mathematical model implementation: Linear Regression

4. Machine Learning Simulation and Results Discussion

In order to properly establish acceptability threshold for model's results a benchmark method has been proposed. It consists of three metrics used to compare results of applied machine learning models: Mean Absolute Percentage Error (MAPE), Root-Mean-Square Error (RMSE) and Coefficient of Determination (R2). The three parameters have been chosen as they represent a basis for qualitative and quantitative comparison of the researched methods. MAPE metric allows valuable insight into the absolute magnitude of the error, while RMSE provides additional intuition into scaled residual size. R2 score represents a statistical measurement of model fit into predicted dataset. Target feature mean has been assumed as baseline prediction, resulting in RMSE of 0.83, MAPE of 9.24% and R2 score of 0 (per R2 score definition).
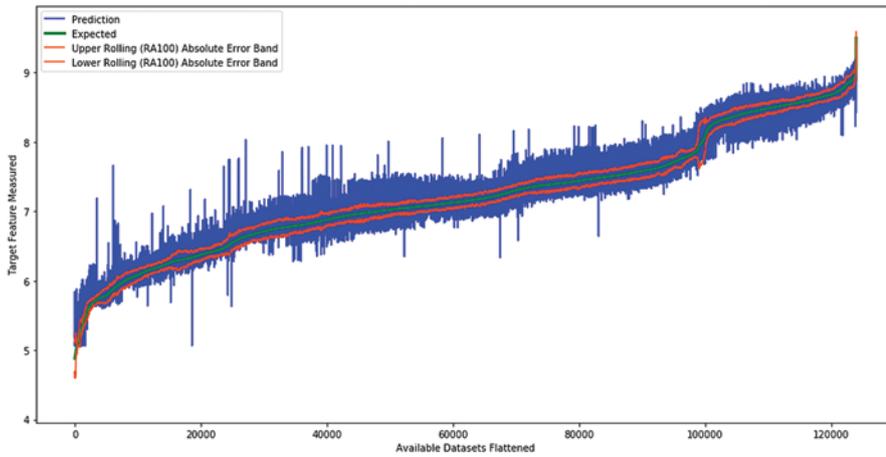

Fig. 7. Deep Learning model results. Due to large number of available datapoints (rows) Rolling Average of 100 samples (RA100) has been used to better visualize model results

Figure 7 shows an example of HP Recoup pressure parameter prediction postprocessing using Deep Learning model with 5-fold cross validation. Similar plots were created for each of the other 31 algorithms evaluated in this article. Figure 7 shows test set target feature range sorted ascending and its corresponding prediction using optimized Deep Learning model (as an example). The figure shows that while model error is generally consisted throughout entire model, the error increases significantly for certain feature inputs across entire target feature range. To prevent single set of measurements from triggering false negative alarms for the customer due to large model error it has been decided to smooth out the overall outcome using rolling average of the last 100 predicted values.

The result error has been shown on the chart in red and represents +/−MAPE band for the used rolling average window. As shown, it significantly reduces a single misprediction impact on overall maintenance system with a drawback of increase in system response time (up to 500 minutes of real time operations, depending on the parameter recording frequency). As overall system architecture is still being adjusted, the exact approach to filter out outlier predictions is not yet fully defined.

Altogether, 11 Scikit-learn machine learning models have been trained and used for comparison purposes followed by Keras (with Tensorflow backend) Artificial Neural Network (ANN) sequential dense models. Each model training process has been proceeded with extensive hyperparameter/parameter tuning on reduced dataset.

First, linear models, such as Linear Regression, Ridge Regression Model, Ridge Cross-Validation Regression Model, Kernel Ridge Regression, Lasso Regression, Lasso Cross-Validation and Elastic Net Model have been tested followed by Bayesian Ridge Regression Model (based on Bayes' theorem, that was applied to aviation gas turbines [4]). Then, more recently developed machine learning models, such as Support Vector Regression Model, Random Forest, and Gradient Boosting were analyzed. Finally, Artificial Neural Networks (ANN) with up to 3 hidden layers have been tested to allow comparison of established machine learning models to Deep Learning model.

The results of analysis are shown on Fig. 8 and 9 and summarized in Table 1. The most effective prediction has been achieved using Random Forest Regression model (0.018 RMSE, 0.122% MAPE and 0.9995 R2). Results acquired for the model, which included additional geometrical information (orifice size), were 0.01% higher (measuring the Coefficient of Determination, R2) than for the same model without additional data. Lower R2 results were achieved with Gradient Boosting Regression and simple Deep Learning model with 3 layers of Artificial Neural Networks (ANN). These models, however, achieved higher MAPE results (above 0.6% and 1.2% respectively). The first linear model that showed high R2 value was Support Vector Regression model achieving 0.155 RMSE, 1.258% MAPE and 0.965 R2 (additional geometrical data did not result in significant accuracy increase).

Random Forest Model has achieved highest performance using 80 estimators, while Gradient Boosted model shown lowest error rate with 245 estimators (and hinge at approximately 50 estimators).

As mentioned briefly in the previous chapter, in case of Artificial Neural Networks (ANN) models, training process consists of calculation of predefined loss function for given outputs and calculating
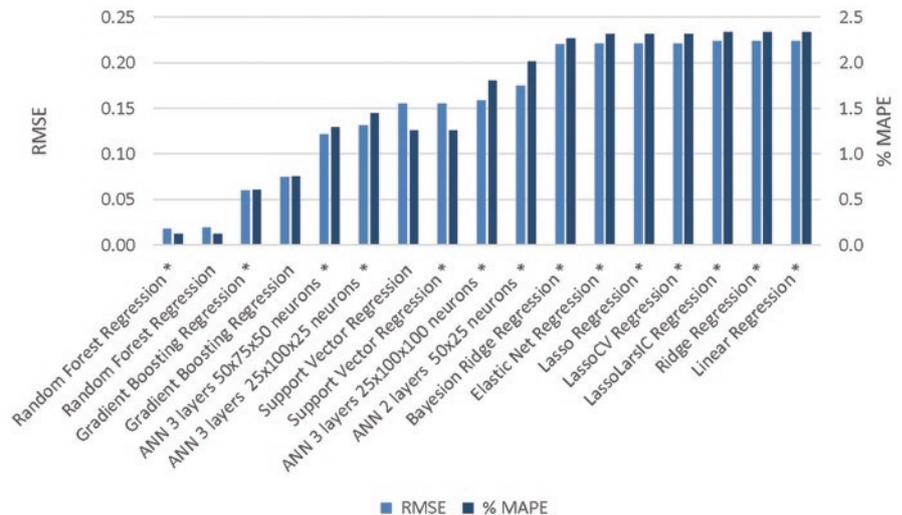

Fig. 8. The Root-Mean-Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) comparison for chosen Machine Learning models. Models with additional geometrical parameter (orifice size) are marked with asterix (*)
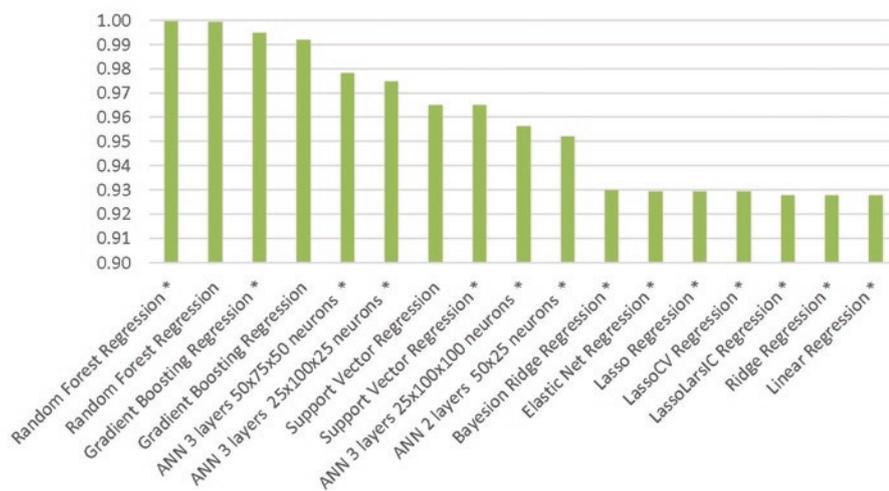
Fig. 9. Coefficient of Determination (R2) comparison for chosen Machine Learning models. Models with additional geometrical parameter (orifice size) are marked with asterix (*)

Table 1. Comparison of RMSE, MAPE and R2 for researched Machine Learning models. Models with additional geometrical parameter (orifice size) are marked with asterix (*)

	RMSE	% MAPE	R2 Score:
Random Forest Regression *	0,018	0,122	1,000
Random Forest Regression	0,019	0,125	0,999
Gradient Boosting Regression *	0,060	0,606	0,995
Gradient Boosting Regression	0,074	0,752	0,992
ANN 3 layers 50x75x50 neurons *	0,122	1,294	0,978
ANN 3 layers 25x100x25 neurons *	0,131	1,449	0,975
Support Vector Regression	0,156	1,259	0,965
Support Vector Regression *	0,156	1,259	0,965
ANN 3 layers 25x100x100 neurons *	0,159	1,803	0,956
ANN 2 layers 50x25 neurons *	0,175	2,013	0,952
Bayesion Ridge Regression *	0,220	2,270	0,930
Elastic Net Regression *	0,221	2,316	0,929
Lasso Regression *	0,221	2,315	0,929
LassoCV Regression *	0,221	2,315	0,929
LassoLarsIC Regression *	0,224	2,337	0,928
Ridge Regression *	0,224	2,335	0,928
Linear Regression *	0,224	2,336	0,928
Linear Regression	0,287	3,006	0,881
LassoLarsIC Regression	0,287	3,008	0,881
Ridge Regression	0,287	3,006	0,881
Bayesion Ridge Regression	0,287	2,998	0,881
Elastic Net Regression	0,288	3,030	0,880
Lasso Regression	0,288	3,037	0,880
LassoCV Regression	0,288	3,037	0,880
Kernel Ridge Regression	0,515	5,873	0,617
ANN 3 layers 25x25x100 neurons *	0,416	4,572	0,582
ANN 3 layers 50x100x100 neurons *	0,422	4,728	0,581
ANN 3 layers 50x25x75 neurons *	0,479	5,478	0,543
Kernel Ridge Regression *	0,646	7,606	0,398
RidgeCV Regression *	0,779	8,814	0,125
RidgeCV Regression	1,865	21,447	-4,023

partial derivatives across different layers for all weights (parameters representing connection strength between nodes) and biases (parameters used to adjust the model output to fit best data set). Recent years have shown increase in model efficiency through implementing new design of activation functions (including ReLU, used in this analysis), initialization methods, regularization methods, learning rate adjustment (including ADAM, also used in this thesis) and input sampling (SGD) while GPU/TPU based increase in computation power significant reduction in training time. While computer disc memory size consistently increases, introduction of Batch Gradient Descent allows training on theoretically infinite size dataset by subsampling it into smaller pieces capable of fitting into memory. As such Deep Learning allows user to operate and generalize much larger problems and enables end-to-end solution generation for any machine learning problem. This contrasts with machine learning models such as SVC, which are limited by memory size and can only operate on user predefined features.

Different size of 3 hidden layers have been tested within a range of 25-100 and an increment of 25 for each layer. Optimum performance has been obtained for a network with 25 nodes in the layer 1, 25 nodes in layer 2 and 100 nodes in layer 3.

Initially team considered utilization of additional categorical feature, HP Recoup pressure orifice size. The analysis has shown that it did not play significant role in case of complex models' application, such as Random Forest or Gradient Boosting. At the same time most of the linear models showed significantly higher R2 results for models with additional data. For example, Lasso Regression had 5.6% R2 score higher while Kernel Ridge Regression (generally one of the least performing models) achieved 54.9% difference.

Presented approach has been validated on gas turbines with operational range very similar to the training data, which partially skews presented results up. Expansion of available dataset for additional gas turbines with further hyperparameter tuning should increase model generalization capability while maintaining its high accuracy. Furthermore, analysis considered multivariate approach for single output, which results in time series information loss and relatively high point-to-point model results variation discovered during data postprocessing. Variation could be handled by simple methods like rolling average. Further increase in model accuracy can be achieved by extracting data sequence information. For example, Recurrent Neural Network could be used on top of preliminary model (ML/DL) results to reduce target feature variation and increase accuracy.

5. Conclusions

The presented study shows comparison of the gas turbine engine HP Recoup pressure prediction. It may be used for gas turbine predictive maintenance planning, potentially allowing gas turbine cost model improvement and optimization, as shown by Deloux et al. [9]. The applicability of numerical machine learning (ML) based prediction models for gas turbine operating parameters prediction has been demonstrated. To specify acceptable model threshold a basic (target data average) benchmark has been proposed and used. In order to allow quantitative comparison between different models 31 machine learning algorithms have been tested, including Artificial Neural Networks, random forests, boosted random forest and SVC. Best results were obtained for random forest regression due to its quick generalization capability enabling ensemble solution with relatively low computational power. Gradient boosting methods also have shown high accuracy due to residua minimization approach utilized in algorithm design. Artificial Neural Networks or deep learning methods were also shown application potential, showing high accuracy results with only 3 hidden layers. While computational requirement for deep learning hyperparameter tuning are significantly higher than those of random forest regressor [15], ANN model can be easily tuned and adjusted for other, similar problems (transfer learning), while majority of machine learning algorithms must be completely retrained for new purposes.

Moreover, data tests showed that additional geometrical data (such as orifice size, available for chosen gas turbines in researched dataset) is not always crucial to improve prediction quality, although it improves overall accuracy of the model and should be used if available, if only to check for model overfitting. Relatively simple numerical model result's comparison leads to most appropriate model, that guarantees high accuracy. Currently Artificial Neural Networks (ANN) architectures could also offer valuable insight such as prediction confidence, which would be critical for applications such as predictive maintenance.

Lastly, the developed methodology is applicable to any of the gas turbine parameter, when reference physics-based models and dataset from sufficiently large fleet are available to validate the accuracy of the data-driven algorithms developed. Achieved results showed that high accuracy may be obtained using the same input data, but with different machine learning algorithms after extensive hyperparameter tuning.

References

1. Augustyn S. Energy Model of Change in Technical Condition of Aircraft Power Plants and Space Propulsion Systems. Aviation Advances & Maintenance 2017; 40(2), https://doi.org/10.1515/afit-2017-0009.
2. Badeer G. H. Engine Thrust Bearing Condition Monitoring Method. US Patent US no. 6,637,932 B2, GE Schenectady, NY, US, Oct. 23, 2003.
3. Badeer G. H. GE Aeroderivative Gas Turbines - Design and Operating Features. GE Power Systems GER-3695E (10/00) available online: https://www.ge.com/content/dam/gepower-pgdp/global/en_US/documents/technical/ger/ger-3695e-ge-aero-gas-turbine-design-op-features.pdf
4. Batalha E. Aircraft Engines Maintenance Costs and Reliability. An Appraisal of the Decision Process to Remove an Engine for a Shop Visit Aiming at Minimum Maintenance Unit Cost 2012.
5. BHGE aeroderivative gas turbines portfolio available online at https://www.bhge.com/aeroderivative-gas-turbines
6. Carlevaro F, Cioncolini S, Sepe M, Parrella I, Escobedo E, Allegorico C, De Stefanis L, Mastroianni M. Use of operating parameters, digital replicas and models for condition monitoring and improved equipment health. ASME Turbo Expo 2018, https://doi.org/10.1115/GT2018-76849.
7. Chmielewski M, Fulara S, Gieras M. Numerical Prediction of GTD-350 Turboshaft Engine Combustor Deterioration. Journal of KONES 2017; 24(2): 47-58.
8. Cyrus B Meher-Homji C. B, Yates D, Weyermann H. P. Aeroderivative Gas Turbine Drivers for The ConocoPhillips Optimized CascadeSM LNG Process - World's First Application and Future Potential. 15th International Conference & Exhibition on Liquefied Natural Gas 2007 available online: http://www.ivt.ntnu.no/ept/fag/tep4215/innhold/LNG%20Conferences/2007/fscommand/PS2_6_Meher_Homji_s.pdf
9. Deloux E, Castanier B, Berenguer C. Predictive maintenance policy for a gradually deteriorating system subject to stress. Reliability Engineering and System Safety 2009; 94: 418- 431, https://doi.org/10.1016/j.ress.2008.04.002.
10. Demgne J, Mercier S, Lair W, Lonchampt J. Modelling and numerical assessment of a maintenance strategy with stock through Piecewise Deterministic Markov Processes and Quasi Monte Carlo methods. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 2017; 231(4): 429-445, https://doi.org/10.1177/1748006X17712121.
11. Fulara S, Chmielewski M, Gieras M. Experimental research of the small gas turbine with variable area nozzle. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 2019; 233(15): 5650-5659, https://doi.org/10.1177/0954410019853977.
12. Galar D, Gustafson A, Tormos B, Berges L. Maintenance Decision Making Based on Different Types of Data Fusion. Eksploatacja i Niezawodnosc - Maintenance and Reliability 2012; 14 (2): 135-144.
13. Garcia Nieto P.J, Garcia-Gonzales E, Sanches Lasheras F, de Cos Juez F.J. Hybrid PSO-SVM-based method for forecasting of the remaining useful life for aircraft engines and evaluation of its reliability. Reliability Engineering & System Safety 2015; 138: 219-231, https://doi.org/10.1016/j.ress.2015.02.001.
14. GE Power & Water Distributed Power. LM2500+ G4 BaseGas Turbine Fact Sheet available online: https://www.ge.com/content/dam/gepower-pgdp/global/en_US/documents/product/lm2500-plus-g4-fact-sheet.pdf accessed on September 2018
15. Heaton J. Automated Feature Engineering for Deep Neural Networks with Genetic Programming. College of Engineering and Computing Nova Southeastern University 2017.
16. Herbert L. Designing for Reliability, Maintainability, and Sustainability (RM&S) in Military Jet Fighter Aircraft Engines, Massachusetts Institute of Technology 2002.
17. Huang B, Wang Z. The Role of Data Prefiltering For Integrated Identification and Model Predictive Control. IFAC Proceedings Volumes 1999; 32(2): 6751-6756, https://doi.org/10.1016/S1474-6670(17)57153-0.

18. Huang H.Z, Tong X, Zuo M. J. Posbist fault tree analysis of coherent systems. Reliability Engineering & System Safety 2004; 84(2): 141-148, https://doi.org/10.1016/j.ress.2003.11.002.

19. Institute of Aviation. GE Challenge available online: https://ilot.edu.pl/tag/ge-challenge accessed on August 2018

20. Jankowski A, Kowalski M. Creating Mechanisms of Toxic Substances Emission of Combustion Engines. Journal of KONBiN 2015; 36(1): 33-42, https://doi.org/10.1515/jok-2015-0054.

21. Kopytov E, Labendik V, Yunusov S, Tarasov A. Managing and Control of Aircraft Power Using Artificial Neural Networks. Proceeding of the 7th International Conference "Reliability and Statistics in Transportation and Communication 2007.

22. Kozik P, Sęp J. Aircraft Engine overhaul demand forecasting using ANN. Management and Production Engineering Review 2012; 3(2): 21-26.

23. Kumar U. D, Crocker J, Knezevic J. Evolutionary Maintenance for Aircraft Engines. Proceedings Annual Reliability and Maintainability Symposium 1999: 62-68.

24. Liu D, Zhang H, Polycarpou M, Alippi C, He H. Elman-Style Process Neural Network with Application to Aircraft Engine Health Condition Monitoring. Advances in Neural Networks. Proceedings of the 8th International Symposium on Neural Networks 2011.

25. Louppe G. Understanding Random Forests: From Theory to Practice. ArXiv Preprint 2014; ArXiv:1407.7502: 35.

26. Lu J-M, Innal F, Wu X-Y, Liu Y, Lundteigen M. A. Two-terminal Reliability Analysis for Multi-Phase Communication Networks. Eksploatacja i Niezawodnosc - Maintenance and Reliability 2016; 18(3): 418-427, https://doi.org/10.17531/ein.2016.3.14.

27. Lu J-M, Lundteigen M. A, Liu Y, Wu X-Y. Flexible Truncation Method for The Reliability Assessment of Phased Mission Systems with Repairable Components. Eksploatacja i Niezawodnosc - Maintenance and Reliability 2016; 18 (2): 229-236, https://doi.org/10.17531/ein.2016.2.10.

28. Michelassi V, Allegorico C, Cioncolini S, Graziano A, Tognarelli L, Sepe M. Machine learning in gas turbines, from component design to asset management. ASME Journal: Global Gas Turbine News 2018; 140(09): 54-55, https://doi.org/10.1115/1.2018-SEP5.

29. Mingazov B. G, Korobitsin N. A. Use of Probability Indices for Assessment of Gas Turbine Power Station Reliability under Commercial Operation Conditions. Russian Aeronautics 2009; 53(2): 226-229, https://doi.org/10.3103/S1068799810020170.

30. Morris A.S, Langari R. Measurement and Instrumentation: Theory and Application 2016, https://doi.org/10.1016/B978-0-12-800884-3.00017-4.

31. Pawełczyk M, Bibik P. Wykorzystanie nowoczesnych narzędzi inżynierskich w projektowaniu bezzałogowego wiropłata czterowirnikowego. Materiały IX Krajowego Forum Wiropłatowego. Instytut Lotnictwa 2013, https://doi.org/10.5604/05096669.1106572.

32. Simon D.L. An Integrated Architecture for On-Board Aircraft Engine Performance Trend Monitoring and Gas Path Fault Diagnostics. 57th Joint Army-Navy-NASA-Air Force (JANNAF) Propulsion Meeting sponsored by the JANNAF Interagency Propulsion Committee 2010.

33. Soumitra P, Kapoor K, Jasani D, Dudhwewala R, Gowda V. B, Gopalakrishnan Nair T.R. Application of Artificial Neural Networks in Aircraft Maintenance. Repair and Overhaul Solutions, Analysis and Manufacturing Technologies 2008.

34. Tibshirani R, Friedman J, Hastie T. Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software. 2010; 33(1), https://doi.org/10.18637/jss.v033.i01.

35. Verma M, Kumar A. A novel general approach to evaluating the reliability of gas turbine system. Engineering Applications of Artificial Intelligence 2014; 28: 13-21, https://doi.org/10.1016/j.engappai.2013.10.001.

36. Walsh P, Fletcher P. Gas Turbine Performance. Blackwell Science 2004, https://doi.org/10.1002/9780470774533.

37. Zadeh L.A. Fuzzy sets. Information and Control 1965; 8(3): 338-353, https://doi.org/10.1016/S0019-9958(65)90241-X.

38. Zhang H, Keerthi S, Mahajan D, Dhillon I.S, Hsieh C.J. Gradient Boosted Decision Trees for High Dimensional Sparse Output. In ICML; 2017: 3182-3190.

Maciej PAWEŁCZYK
Szymon FULARA
Łukasiewicz Research Network – Institute of Aviation
al. Krakowska 110/114
02-256 Warsaw, Poland

Marzia SEPE
Alessandro DE LUCA
Baker Hughes
Via Felice Matteucci 2
50127 Firenze, Italy

Maciej BADORA
Baker Hughes
al. Krakowska 110/114
02-256 Warsaw, Poland

E-mails: maciej.pawelczyk@ge.com, szymon.fulara@ge.com,
marzia.sepe@bakerhughes.com, alessandro.deluca@bakerhughes.com,
maciej1.badora@bakerhughes.com