CEAS Aeronautical Journal (2020) 11:277–287
https://doi.org/10.1007/s13272-019-00422-0

ORIGINAL PAPER

Modeling of aircraft fuel consumption using machine learning algorithms

S. Baumann[1] · **U. Klingauf**[1]

Received: 28 February 2018 / Revised: 9 May 2019 / Accepted: 16 September 2019 / Published online: 30 September 2019

Abstract
With the aid of recording systems such as the flight data recorder, information from aircraft sensors can be transmitted in-situ to airlines or maintenance providers during the flight in form of reports, or stored for subsequent analysis in the postprocessing. This allows the determination of current aircraft performance with regard to fuel consumption or emissions. At present, data in a highly aggregated form (predominantly averages) are used for statistical estimations or physical models. The metrics are calculated on a rolling basis as performance indicators of the aircraft and are then compared with book values from the manuals or with data from the performance monitoring system. However, this procedure represents only a situational, aggregated point evaluation of stable flight conditions. The data aggregation is based on strict validity limits for parameter variations. Furthermore, trigger conditions of the recording logic determine the number and quality of the transmitted reports, such, that only a few data points are available for performance analyses. To improve realistic performance analyses, the time series of all aircraft sensors over the entire flight mission (full-flight data) can be used. In contrast to physical models, this study presents data-based approaches using machine-learning tools from the field of artificial intelligence, to develop fuel flow models based on full-flight data. The proposed methods result in detailed statements for the diagnosis of aircraft fuel consumption. The paper deals with the model development and results of different analyses based on a variety of an airline's operational flight data records. This study describes the learning methods and shows results for two different data-based models, including neural networks and decision trees. Finally, future applications for the models and an outlook on the authors' activities will be provided.

Keywords Machine learning · Aviation · Fuel efficiency · Data-based models · Grey box modeling

1 Introduction

Fuel consumption accounts for about 20% of the total operating costs of an airline (see [1]). Monitoring the fuel required for a specific flight mission can increase the efficiency of aircraft operations, resulting in cost savings for an airline. Aircraft performance monitoring (APM) is used to determine consumption, to evaluate drag of an aircraft, and to monitor fuel consumption metrics over the aircraft life cycle. The benefits of fuel efficiency retrofits can therefore be quantified by comparing APM before and after applying a measure. For planning and performance monitoring of aircraft and fleets, statistical analyses of historical data sets and physical estimation models are primarily used. Performance indicators that are determined from dedicated operating conditions during cruise are compared with book values from the flight crew operating manual (FCOM). The performance indicators represent point evaluations in the form of highly aggregated parameters. However, these are based on strict recording logics, so only a few reports with data from preselected sensors provide any kind of aircraft status. In this way, only one to four reports are generated during a flight. For performance monitoring, these methods therefore exhibit considerable inaccuracies compared to real, dynamic flight operations and provide idealized assessments of aircraft performances.

To optimize fuel consumption analyses and better forecast fuel profiles, the Technical University of Darmstadt is developing diagnosis and prediction models using machine learning algorithms. In contrast to physical model calculations

✉ S. Baumann
baumann@fsr.tu-darmstadt.de

[1] Institute of Flight Systems and Automatic Control, Technische Universitaet Darmstadt, Otto-Berndt-Str. 2, 64289 Darmstadt, Germany

and estimations, this paper deals with data-based modeling of fuel profiles based on the fuel flow for individual flight phases as well as for entire flight missions. With this method, flight route, aircraft and environmental characteristics can be mapped to create individual diagnoses and forecasts. We use flight operational data records as the basis for the data-based models.

In the scientific literature, there are already a few papers that address the topic of estimating and forecasting fuel consumption by means of data-based methods in machine learning. For example, Chati and Balakrishnan [2] developed a machine learning model based on physical relationships in the form of a classification and regression tree, which provides more precise estimations of fuel flow in comparison with databases of the International Civil Aviation Organization (ICAO) and the Base of Aircraft Data (BADA) of EUROCONTROL. Horiguchi et al. [3] use various decision tree techniques such as random forest and XGBoost, as well as deep neural networks to analyze low-cost airline flight planning and passenger data, and determining possible improvements by comparing values of the fuel planning and airline scheduling. Trani et al. [4] presented an estimation method for the aircraft fuel consumption using neural networks. However, the data basis for the training and test of the neural network is information from the aircraft performance manual. The results achieved were also compared with BADA calculations. Turgut and Rosen [5] developed a genetic algorithm approach to identify the relationship between fuel flow and altitude during the descent flight phase. Here, an optimization problem is analyzed, addressed with the help of a stochastic search technique.

In contrast to this paper, the studies are based on databases that are not all publicly accessible. Furthermore, information from manuals was also used instead of real flight data recordings from the aircraft's quick access recorder. In addition, the studies cited include model comparisons with different benchmarks. Thus, the studies can only be compared with each other as proof of method and not as proof of performance.

The present work is subdivided into the description of the machine learning methods in Sect. 2. The available data base is presented in Sect. 3 and the regression method is explained in Sect. 4 in more detail. The results of data-based modeling are presented and discussed in the form of progressions and validation metrics in Sect. 5. The paper concludes with a summary and outlook of future work in Sect. 6.

2 Learning methods

Iterative learning models have long been used in research and applications of artificial intelligence. The application can be simple to use, but the model behavior is analytically difficult to describe and highly complex. The diagnosis and prognosis results also depend on the data basis and the settings of process parameters. This includes, for example, the network topology. Due to a large number of possible variations and the limited intervention in the calculation process (black-box model), these methods require precise model development and evaluation. In accordance with the principle of parsimony, the models have to be kept as simple as possible with reasonable losses of accuracy (see [6]). Nevertheless, with the help of machine learning methods in the area of algorithm-based modeling, classical physical and statistical models can be surpassed for a large number of applications [7].

Choosing an optimal regression method for the given task is not trivial. For an first overview, a test was carried out examining the performances of different regression methods including multi-layer perceptron networks, regression trees and ensemble methods like random forests, support vector machines, Gaussian processes as well as baseline methods such as linear regressions, decision tables, decision stumps and zero rule classifiers.

In spite of the character of the regression problem and the statistical basis of support vector machines and Gaussian process regressors, it was not possible to use these methods for the present study. Due to the very extensive database these algorithms are not initially feasible for large data sets. Because of storing the kernel matrices large amounts of memory and training time are necessary that scale exponentially with the number of data samples. For their application, further optimization of the database is necessary, such as approximations of the eigenvalues/eigenvectors for large matrices using smaller submatrices or approximations of the optimization problem using subproblems (see e.g. [8]). For future studies of the authors these developments and a bootstrapping approach are considered for the use of support vector machines and Gaussian process regressors. In the literature, studies such as from Haifeng et al. in Ref. [9] and from Chati and Balakrishnan in Ref. [10] have investigated the use of support vector machines and Gaussian process regressors on small data sets for the addressed purpose of fuel consumption modeling.

In the following, the most promising methods, which were identified in the preliminary study and used in this article, will be introduced in more detail. These include neural networks and decision trees. Latter ones are also statistically comprehensible for evaluate decision rules and identify key features of the data set.

2.1 Neural networks

Artificial neural networks enable the description of complex interdependencies. Similar to the human brain, the functional principle is based on serial and parallel interconnections of

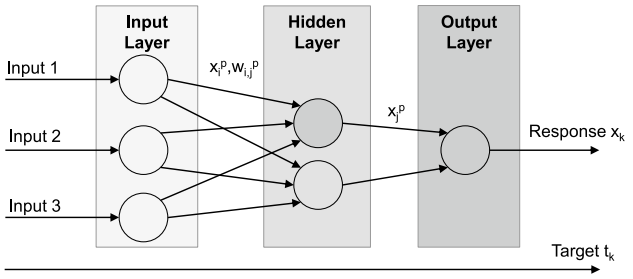

Fig. 1 A neural network with three inputs, a hidden layer with two neurons and an initial layer

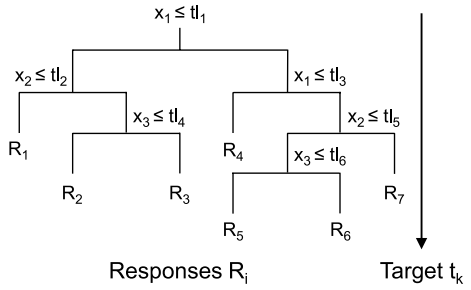

Fig. 2 Decision tree with test conditions ($x_j \leq t_k$) at each internal node and learned classification values (R_i) at the leaves of the tree

perceptrons (analogous to neurons in the human brain). Each perceptron can be supplied with several inputs and forms a dedicated initial state. For the cross-linking of individual perceptrons, compounds (analogous to synapses in the human brain) are used, so that input values depending on the contribution to the regression target are weighted. The performance of a neural network is determined by the network architecture, for example the number of hidden layers and the number of perceptrons in each layer, the type of interconnections, the training method and activation function. During training, the weights change to reduce the error of the initial values to a minimum. Alternative stop conditions result from the number of generations (epochs) passing through the neural network [6]. Figure 1 shows an exemplary artificial neural network with three inputs and one intermediate layer for mapping an output.

At each perceptron, the output parameters x_j^p are presented by a combination of the input parameters x_i^p and the weights $w_{i,j}^p$ by

$$x_i^p = f\left(\sum_{j=1}^{n} w_{i,j}^p \cdot x_i^p\right). \tag{1}$$

This is exemplified in Fig. 1 at the first perceptron in the hidden layer. The perceptron represents a simple classifier, which calculates the values one or zero via the simulation of a sigmoid function depending on a threshold value Θ by the activation function a^p at each perceptron with

$$a^p := \begin{cases} 0 & \text{if } \sum_{j=1}^{n} w_{i,j}^p \cdot x_j^p < \Theta \\ 1 & \text{else} \end{cases}. \tag{2}$$

As a quality criterion of the neuronal network, the function J is given by

$$J = \min\left(E(w_{i,j}^p) = \frac{1}{2}\sum_{k=1}^{n}\left(t_k - x_k\right)^2\right), \tag{3}$$

where the quadratic error E between the target values t_k in the training data and the calculated response x_k of the net. The learning rule adjusts the weights of the respective inputs to the perceptrons at each iteration. The learning adaptation of the weights is carried out proportionally to the error via the so-called delta rule, until the changes become suitably small or until a calculation limit is reached (e.g. by the number of iterations). The procedure is referred to as backpropagation (see [6, 11]).

The neural network used in this article features an architecture with a maximum number of 176 inputs for multiple input single output calculations. One hidden layer with a number of ten perceptrons is used (for further explanations see Sect. 4.3). The information flow of the network is directed forward from the input to the output layer (feed-forward network). To prevent an over-adaptation (overfitting) of the neural network, the so-called Bayes regularization is used. Thus, models can be optimized to lower complexity and a good generalizability can be achieved. Furthermore, correction terms are stored in the optimization functional, which, for example, penalize the size of weights or favor the selection of a network with the best representation of legality behind the data and not the best reproduction of the data points (see [12]).

2.2 Decision trees

Decision trees are straightforward procedures for classification and regression tasks. The results can be visualized by means of a learned tree structure, whereby the leaves (points without further division of the tree) are interpreted as regression results and the branches as calculation results at the nodes in the tree (see [6]). Figure 2 shows an exemplary decision tree structure.

The structure of a decision tree is divided into the decision nodes and the leafs. The learning procedure with decision trees selects a characteristic x_i (see Fig. 2) at each node (also called attribute or feature) on the basis of the description variables of the training data set, which best divides the data set. Data-based, decision rules on the basis of threshold

levels tl_i are being trained at the internal nodes of the tree. The nodes usually represent binary classifications where exactly two descendants occur. The classification or regression results are then presented as classes R_i on the leaves. This is not the construction of a global regression model, but rather predictor intervals are stretched out, which build a superior, mean result for each leaf of the tree (see [6, 13]).

The quality of the tree architecture can be described by the gained information. This is composed of the entropy E by

$$E(P) = E(P_1, P_2, \ldots, P_n) = -\sum_{i=1}^{n} P_i \cdot \log_2(P_i), \tag{4}$$

and corresponds to a probability distribution for the different outputs n. Entropy is also a measure for uncertainty. The smaller the probability of occurrence of a sample, the higher its information is. Imagine that every data sample in the training set generates a leaf or a class as result for the decision tree. The entropy inevitably increases, the tree overfits the training data, the gained information concerning the training data is high but decreases if same matters occur frequently. Additionally, the quality decreases and the global behavior of the problem cannot be modeled. Thus, the trade-off between accuracy and complexity must be taken into account. The information content I of a finite data set D can thus be described as

$$I(D) = 1 - E(D). \tag{5}$$

To be able to select a characteristic feature A with the greatest information gain G at each node i, this node is calculated with

$$G(D, A) = E(D) - \sum_{i=1}^{n} \frac{|D_i|}{|D|} \cdot E(D_i), \tag{6}$$

whereby the second part of Eq. (6) represents the information content by the characteristic A in the data subsets D_i. The information content is used to decide for each node which characteristic contributes to the division of a tree. When an attribute A splits the set D into subsets D_i the average entropy is compared to the sum of the entropy of the original set D. Then, the attribute A that maximizes the difference of Eq. (6) is selected to reduce disorderliness and to maximize information gain. The first division is based on the greatest amount of information. The characteristics with descending information content are then used to divide the tree at the attributes. Building decision trees is called divide and conquer or greedy algorithm (see [6, 13, 14]).

For regression tasks, entropy calculations are replaced by mean squared error calculations and class labels are replaced with numerical averages. In line with the quality function of neural networks, the mean squared error between the target value in the training data and the averaged calculation result of the tree structure at a leaf is used as degree of impurity for the evaluation to divide the tree structure for the regression task. A weighted sum is calculated on the basis of the regression result for each node. If the error at the node is acceptable (under a defined threshold), then a leaf node is created. Therefore, decision trees can be used for regression of continuous target values. Then, each leaf refers to an average numerical attribute for each class. The tree division is carried out up to a defined stop criterion, which can be determined by the tree size and depth. Based on the classification, the system searches for those characteristics as split criteria for which minimum errors occur (see [15]).

To prevent a tree from being under or over adapted, the structure is optimized on the basis of the test data set. This is known as pruning of decision trees. Based on the structure of the different decision trees, one can select a tree with regard to the weighing of errors and complexity (see [16]).

Random Forest learns several uncorrelated decision trees for classification tasks. The regression result is then determined in the form of a mean classification/regression in a majority decision procedure under the decision trees (see [17]). Model trees (also known as M5 according to Quinlan [18]) represent an extension of the classical regression trees and differ, that linear models for approximation of the classes are developed on the leaves of the trees instead of discrete classes (see [18]).

3 Database

A large number of full-flight data records are used as the data basis for this investigation, which correspond to the time series of all aircraft sensors. The data is provided by the National Aeronautics and Space Administration (NASA) [19] for research purposes. The database comprises more than 250 gigabytes of data and contains over 180,000 flight missions from the years 2001 to 2003 in North America from an unknown airline. There are no details on aircraft identification codes (tail signs) or aircraft types given. However, based on the nature of the data, it can be assumed that it is one aircraft type in a single fleet. Another special feature of the data is that it must be a four-jet aircraft. The recorded parameters can be divided into groups of air data (e. g. temperature, pressure, altitude, climb/sink rate, angle of attack, speed), engine parameters (e.g. engine speed, exhaust gas temperature, engine number and cycle, oil pressure/temperature, fuel flow, vibration), control inputs (e.g. control surface positions, inputs in the flight management system), navigation data (e.g. latitude and longitude, course, aircraft orientation, distance to waypoint, acceleration), environmental data (e.g. wind direction/speed) and system parameters (brake pressure, (fire) warnings, errors and status parameters). In

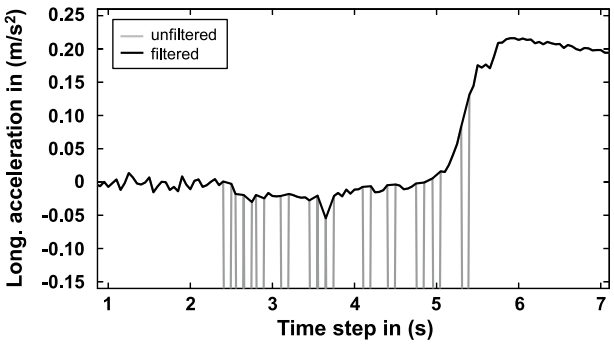

Fig. 3 Preprocessing of an exemplary aircraft parameter progression (take-off acceleration) for the elimination of outliers using median filters or k means clustering

total, 186 parameters with sampling rates from 0.25 to 16 Hz are recorded. The quantitative records contain non-metric and metric continuous data.

4 Regression method

This chapter discusses aspects of the data mining process and provides the metrics used to evaluate the model results.

4.1 Data preprocessing

Due to faulty data recordings, the modeling parameters have to be appropriately prepared for processing iterative learning methods. Otherwise, the unprocessed input data would have a negative effect on the model quality, for example, as the learning methods can be sensitive to outliers [20].

Ranking filters are suitable for the data basis used, which determine the mean absolute deviation from the median of a sample using scaled position of the standard deviation of the ensemble values. These filters contribute to the fact that parameter-dependent characteristics can be considered by variable window widths in the filters without identifying the system dynamics as outliers. The method sets a moving median filter, which evaluates the deviation based on the standard deviation σ between median and ensemble value [known as median absolute deviation (MAD)]. Similar results in preprocessing can be obtained by learning a clustering algorithm, for example k means, for the classification of nominal conditions and outliers. In Fig. 3, an exemplary result of the preprocessing is shown.

In contrast with conventional smoothing algorithms such as spline curve interpolations, an outlier treatment removes outliers from the data set. In addition, care is taken during the pretreatment to ensure that no phase shift effects are introduced, as they can occur in the case of floating statistical metrics due to short-term level jumps (see [21]).

Based on the different recording rates of the data acquisition, the data sets are subjected to a down-sampling procedure to the lowest sampling rate after filtering and elimination of outliers.

Furthermore, the data can be transformed in the context of preprocessing by applying a normalization or $(\mu - \sigma)$ standardization. It has been shown that neural networks converge faster when input data is normalized. In a first step, the mean value is subtracted from the values of the characteristics. In the second step, the variance of the features is scaled to the standard deviation. In this case, the inputs are transformed to follow a normal distribution, with a mean of zero and a standard deviation of one. If the input characteristics of the neural network show strong differences in their value ranges, a normalization is a reasonable option. In addition, many weight initialization methods assume normalized input data (see [22]).

Data points that differ greatly from each other are advantageous for the training of neural networks. In addition, it can be useful to randomly mix the time-dependent instances when processing the data prior to the application of machine learning methods. This process is known as shuffling.

The previously presented methods contribute to an optimization of the machine learning algorithms by adapting different parameter scales and avoiding learning effects due to temporal allocation in favor of a higher generalizability of the models, especially when using neural networks (see Sect. 4.4; see [20]).

4.2 Selection of model predictors

In addition to all available parameters of the data set with a number of 176, selected subsets in relation to the underlying flight physics and engine behavior are used as manually selected predictors for modeling. The subsets each comprise about 40 parameters. In this study, the modeling objective is the fuel flow summed over all engines. In addition, linear and non-linear correlation and variance analyses based on statistical methods can be used to select modeling parameters. The best 30 correlated parameters based on a Spearman rank correlation coefficient with respect to the modeling target fuel flow are then selected (see [23]). An algorithm-based, automated selection of input characteristics is also used as a sequential feature subset selection (see [20, 24]). These methods reduce model complexity by neglecting irrelevant characteristics for data-based modeling.

4.3 Selection of the neural network architectures

The addressed analysis question is a regression problem. A feed-forward network (FFN) as multi-layer perceptron architecture is therefore used as regressors with a single output node for the fuel flow. FFN represent a simple type

of neural networks, where flow of information is unidirectional from the input layer to the output layer. As a fully connected net, each perceptron is linked to all neurons in the following layer. The net consists of an input layer, a certain number of hidden layers and an output layer.

There are no exact rules for the selection of suitable numbers of hidden layers and perceptrons. The numbers are mainly based on the accuracy to be achieved. In addition, the complexity of the problem to be achieved as well as the quantity and quality of the training data must be taken into account. One hidden layer is sufficient for the large majority of problems (see [25]). To use a simple topology, one hidden layer was chosen on the basis of a related topic analyzed by Du et al. in Ref. [26] to predict the fuel consumption of a vehicle. Moreover, this choice coincides with the statement "a feed-forward network with a single layer is sufficient to represent any function, but the layer may be infeasibly large and may fail to learn and generalize correctly" made by Goodfellow et al. in Ref. [27].

A-priori determinations of the number of neurons in the hidden layers of the neural network are not feasible due to the black-box character. A feed-forward structure with a total of three layers was chosen as the architecture of the multi-layer perceptron network. The neurons of the input layer are connected to the input parameters. It should be noted that the number of neurons in the input layer was varied during the investigation. Following the decision rules given by Du et al. in Ref. [26], ten perceptrons in the hidden layer were chosen. This represents a trade-off between the estimation formulas (7) and (8) as

$$n_h = \sqrt{n_{\text{Input}} \cdot n_{\text{Output}}}, \tag{7}$$

and

$$n_h = n_{\text{Input}} \cdot \log(2), \tag{8}$$

where n_h is the number of perceptrons in the hidden layer, n_{Input} is the number of perceptrons in the input layer and n_{Output} the number of perceptrons in the output layer. For reasons of comparability for subsequent studies, the number of perceptrons was not further optimized. However, Baklacioglu's work in Ref. [28] is also pointed out for completeness, in which a genetic algorithm was used to determine the number of perceptrons in the hidden layer.

The Bayesian regularization was used as the training function. This is a development of the Levenberg–Marquardt optimization, which minimizes a combination of quadratic errors and weights. A nonlinear sigmoid activation function for the perceptrons in the hidden layer was chosen. A maximum of 1000 epochs were carried out each run.

4.4 Learning process and generalizability

When learning data-based methods, a distinction is made between the training and test phase. In the training phase, the relationships between the nodes of the network are modified using a training data set. In the test phase, new data records are used to test how well the learned model can map the test data. For example, two thirds of the available data can be used as training data for the training procedure and the remaining third as test data. The shares can be selected randomly.

However, in the context of an n fold cross-validation, it is advisable to use the entire available amount of data for the training procedure. First, the data set for training process is randomly split into n folds. $(n-2)$ partitions are used for the training to adjust the network in terms of weights and biases. The remaining amount of data is retained as a validation set to measure network generalization and to halt training when generalization stops improving, and as a test set. These provide independent measures of network performance during and after training. The error rates and learning parameters of the network are then determined on average for the entire data set with all partitions (see [23]).

The data was subdivided into these three parts in the ratios of 70%, 15% and 15% for training, validation (during training for neural nets) and test data. The samples were chosen randomly. The present study uses a tenfold cross-validation to determine an overall performance metric for the training.

However, it may happen that data points have already been used in training and neighboring points from the same flight are used for testing as well. Once the training is completed, the generalization of the models can be checked against fully unknown data. This so-called hold out or leave one out data set has not already been used for training or test. For a further validation step, another data set was used from the flight operations data. This ensures that independent flights are used for a further evaluation step for the trained models.

The generalization of the models is opposed to the so-called bias variance problem, where the bias is a measure of accuracy and variance is a scattered variable of the regression results based on the test and validation data. The regression results depend heavily on the data basis, the learning method, the complexity of the model and, in some cases, the learning process. For the respective data mining problem, a balance between an over and under adapted model must be found. It is also worth mentioning that effects such as noise, which cannot be influenced during data acquisition, overlay the data. These are implicitly mapped in the learning model. Compensation for these fixed errors with the help of additional expert knowledge is only partially or not at all possible and is not dealt with in this article (see [6, 12, 20]).

4.5 Validation metrics

The focus of this article is on data-based modeling of the aircraft fuel flow and its calculation with new, unknown input data sets. The following metrics are used to evaluate the model quality and the calculation results. Metrics used in this article include (for more details see [29]):

- Mean absolute error (MAE): represents the absolute deviation between the calculated result of the model $x_{m,i}$ and real target $x_{r,i}$, averaged over the number of observations N, and given by

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|x_{r,i} - x_{m,i}\right|. \tag{9}$$

- Mean relative error (MRE): Relative error based in the mean value $\overline{x_r}$ of the real target samples, and given by

$$\mathrm{MRE} = \frac{\mathrm{MAE}}{\overline{x_r}}. \tag{10}$$

- Mean squared error (MSE): Square root of the mean quadratic error given by

$$\mathrm{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_{r,i} - x_{m,i})^2}. \tag{11}$$

- Coefficient of determination (R^2): Quality measure to explain the data variation of the regression model (unexplained variation of the residuals in relation to the overall variation) and given by

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(x_{r,i} - x_{m,i})^2}{\sum_{i=1}^{N}(x_{r,i} - \overline{x_r})^2}. \tag{12}$$

5 Results and discussion

Based on iteratively learning models, different progressions for the fuel flow of an aircraft can be calculated: the results for entire flight missions and the cruise phase are presented hereafter.

5.1 Fuel flow modeling for entire flight missions

Figure 4 shows an example of a fuel flow profile over an entire flight mission, which is calculated with the already introduced neural network. The duration of the mission shown in Fig. 4 is approximately 2 h.

The training process required an average computing time of about 3–4 h on a computer with four cores (Intel® Core™ i5-4440 CPU @ 3.1 GHz) and eight gigabytes of RAM. It should be noted that the training time for neural networks is exponentially linked to the number of training instances and input features. In addition, the selection of the learning process (e.g. Bayes regularization) determines the required resources. Compared to neural networks, decision tree procedures only require about one tenth of the training time. The (preliminary) analyses were performed in MathWorks Matlab 2017a using the statistics and machine learning as well as the neural network toolbox, and in Python using the libraries tensorflow, keras and scikit-learn.

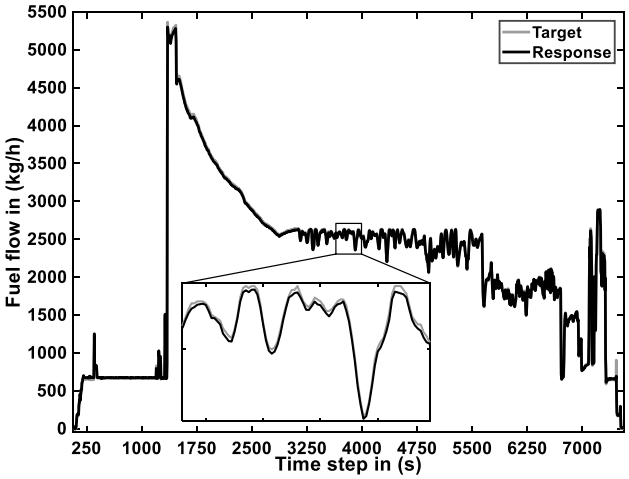

Fig. 4 Real and (by means of a neural network) calculated fuel flow profile for an entire flight mission

To evaluate a robust training of the models and to prove an independence of the data sets used for train the models, several (up to ten) models were trained in an upstream step. With this, a sufficient robustness of the learned networks can be shown. A scatter and density distribution of the evaluation metrics was evaluated. From these models the best was selected and used for the diagnosis task. It can be shown that consistently very good averaged determination coefficients, means and medians of the individual neural networks over the cross-validated data folds with the test data can be achieved. On average the (root) mean squared error is 35.25 kg/h with a standard deviation of 12.5 kg/h, the median is 29.8 kg/h with a 25th percentile at 27.75 kg/h and a 75th percentile at 38.4 kg/h. The scattering range of the coefficient of determination behaves similarly and is 98.5% in the worst case.

For the generated decision trees similar position and scatter values for the performance can be determined. Between 4000 and 6500 leaf nodes as regression results and about 4000 decision rules were learned. For the training, only a maximum number of splits in the order of the sample size and a minimum of four leafs were specified for the tree architecture.

Table 1 Comparison of the validation metrics for the learning methods used to model the fuel flow for entire flight mission

	Training	Validation
Neural net		
MAE in (kg/h) | MRE in (%)	7.25 | 0.4	15.88 | 0.8
MSE in (kg/h)	12.7	22.23
R^2 in (–)	0.9998	0.9995
Decision tree		
MAE in (kg/h) | MRE in (%)	24.04 | 1.3	39.46 | 2.1
MSE in (kg/h)	36.74	66.67
R^2 in (–)	0.9987	0.9952

Table 1 lists the validation metrics for calculating the fuel flow profiles for entire flight missions in comparison of the best models of a neural network and a decision tree. It should be noted that the databases used to validate the learned models have approximately the same number of time instances as the training data (approximately 200,000 each).

A comparison of the two learning methods shows that fuel flow profiles over entire flight missions can be modeled using neural networks with mean relative errors of 0.8% (validation result with a hold out data set). The average fuel flow over a total flight mission is approximately 2000 kg/h. For the entire validation data set, the statistical averaged error for the cumulative fuel consumption per flight (integrated fuel flow over the duration of the mission) are less than 5 kg for the application of a neural network.

On the basis of the square error values and the coefficient of determination, a very good generalizability of the learned models for entire flight missions can be determined. The error between test and validation data increases approximately by the factor of two. Furthermore, there are no significant losses in model quality.

A comparison between the different decision tree procedures (not shown in this study) reveals, that model trees (M5) have a slightly better performance with fewer errors (with comparable model quality) than regression trees and random forests.

5.2 Fuel flow modeling for cruise flight

Figure 5 depicts real target and calculated fuel flow during cruise flight. A regression tree was used for modeling. In comparison with a neural network, the result has a stepped profile, which is due to the classification method of the tree.

In Table 2, metrics for modeling the fuel flow during cruise flight are given in comparison with neural network modeling.

The results for a data-based modeling of the cruise flight show fewer errors than for entire flight missions during training. Performance investigations are carried out by means

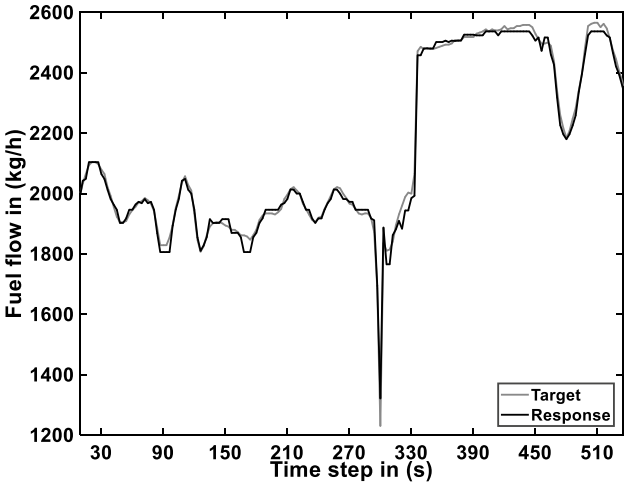

Fig. 5 Real and (by means of a decision tree) calculated fuel flow during cruising flight

Table 2 Comparison of the validation metrics for the learning methods used to predict the fuel flow for cruise flight

	Training	Validation
Neural net		
MAE in (kg/h) | MRE in (%)	4.5 | 0.13	41.3 | 1.9
MSE in (kg/h)	4.08	68.95
R^2 in (–)	0.995	0.880
Decision tree		
MAE in (kg/h) | MRE in (%)	8.4 | 0.74	27.3 | 1.26
MSE in (kg/h)	14.3	36.74
R^2 in (–)	0.998	0.963

of a cross-validation with ten partitions (tenfold cross-validation). For validation, however, inconsistencies with the previous results are found, since the decision tree method has fewer errors than the neural network. For the neural network, the error metrics show considerable losses in generalizability for the validation data set. This is particularly evident in reduction of the coefficient of determination from very good to good model quality as well as an increase of error values based on validation data. In addition, the mean squared error increases tenfold. This is used to evaluate the error susceptibility with respect to outliers, since these are weighted more heavily due to the squaring of the absolute error. Despite the selection of the learning process and the application of avoidance strategies for an over adaptation of the model, this cannot be excluded.

For models with smaller validation datasets (not explicitly presented in this article), there are no significant differences between the metrics for training and validation. The database used is split up in a ratio of 70:30 for training and validation. This means that less than half of the instances are available

for validation than for training. On the basis of this result, it cannot be ruled out, that the flight data used may change over time. These can include, for example, degradations of aircraft efficiency due to external influences, changes in environmental conditions such as changes in flight planning and operations, or short-term changes in efficiency due to maintenance events and repairs carried out. These influences can have direct impacts on the regression results.

5.3 Summary of modeling results

The results from the two preceding sub chapters show, that data-based procedures can calculate the aicraft fuel flow, even in the case of data with uncertainty and outliers, with very good model quality and generalizability. Individual flight phases (climb, cruise and descent) can be modeled with fewer errors in comparison to entire flight missions, while cruise shows the smallest errors under the considered flight phases. Decision trees may occasionally have larger errors, but they require significantly less computing time during the training process (a factor of ten in this study). Neural networks can be prone to higher errors and poorer generalizability when modeling individual flight phases. However, the decisive influence is not definitively clarified. The appropriate selection and reduction of the input parameters could serve as optimization (see Sect. 5.4).

A preprocessing step before the training process of the neural network included the normalization of the training data base. The shuffling of the training data during the learning process has proven to be very useful. Based on the modeling of cruise flight by means of neural networks, a slight increase in the mean absolute error (MAE) during training can be observed. However, error metrics are reduced by about a factor of two compared to validation data sets without mixing. In addition, the model quality can be increased considerably.

5.4 Reduction of modeling parameters

To improve the performance of the regression results, modeling with fewer input parameters is already carried out for this contribution. Initial results show that with only about a quarter of all available parameters comparable calculation results can be achieved, independent of the learning method. Thus, the computing time can be reduced by one third to one tenth. Characteristics are selected manually by the authors and data-based on feature subset algorithms such as principle component analysis, factor analysis, mutual information or wrapper (see [24]). A further selection procedure is also carried out with a ranking correlation coefficient according to Spearman (see [20]) with regard to the fuel flow. With these methods, the input data base can be reduced to about 30–70 parameters for modeling the target, the aircraft fuel flow.

6 Summary and outlook

This study deals with the application of machine learning algorithms for data-based modeling of the fuel consumption of an aircraft in different phases of flight and for entire flight missions. Two non-parametric methods, namely neural networks and decision trees, are applied to freely available flight operation data and compared with each other. Neural networks show the best results for modelings of fuel profiles over an entire flight mission, but have a high resource effort in terms of computing time. This article can serve as a benchmark for future activities in this area.

Future activities of the authors refer to the topological optimization of neural networks and decision trees (e.g. with genetic algorithms). In this regard, algorithms are also to be used to automate the process and to find suitable initial values for topologies and hyper parameter of the models. This is intended to increase the model accuracy. An application of competing networks is also conceivable. In addition, suitable methods for the selection of relevant characteristics and the generation of features from flight operational data are to be identified and developed. The first results of the feature reduction are to be optimized and automated using wrapper algorithms.

For a project of the German Aeronautical Research Programme LuFo V-2 (see below), which is currently being processed by the authors, the method applied in this article will prospectively be used for examining aircraft fuel efficiency. For this purpose, different models for flight route, aircraft and environmental characteristics are to be trained and validated. New flight operation data will then be used to resolve and evaluate efficiency differences in dynamic flight operations for the characteristics. The aim is to carry out an economic technology assessment of retrofit measures on aircraft. These include, for example, the performance increase due to engine washings, the application of modern wing tip fences such as sharklets or the application of resistance reducing foils and structures such as riblets on the wing surfaces.

The commercial application of machine learning methods is becoming increasingly important for aviation. Individual, digital products for the performance monitoring of aircraft and fleets can now be found as applications in decentralized platforms such as General Electric (Predix[1]), Honeywell

[1] See: https://www.predix.io/ (Accessed 8 February 2018).

(GoDirect[2]) and Airbus (skywise[3]). In autumn 2017, Lufthansa Technik (AVIATAR[4]) took up the service of another open platform. Against this backdrop, the present work can contribute to new digital products for fuel consumption modeling and the assessment of aircraft fuel efficiency. A study by Ryerson [30] shows, that the transport of unused fuel results in about two to four and a half percent additional fuel burn of the total fuel consumption (trip fuel) per flight. In addition to the prescribed minimum quantities and reserves, this is also taken along during flight (see [30]). An additional fuel consumption per kilogram of extra weight can be estimated at about 0.045 kg of fuel related to a distance of 1000 km (see [30, 31]). This results in considerable saving potentials over the course of a year of operation of an aircraft or an entire fleet.

Acknowledgements The authors would like to thank the Federal Ministry of Economics and Energy (BMWi) for their support in the RetroEff project (Retrofit Technology Assessment for Efficient and Economical Aircraft Fleets, sub-funding 20Y1513C) as part of the German Aeronautical Research Programme (LuFo V-2). In this context, large parts of this contribution could be worked out by Technische Universitaet Darmstadt. In addition to the Institute of Flight Systems and Automatic Control at Technische Universitaet Darmstadt, the project RetroEff includes the partners Lufthansa Technik AG and DLR institute Air Transportation Systems. The subproject of Technische Universitaet Darmstadt ends on 31.10.2019.

References

1. International Air Transport Association (IATA): Fact sheet—fuel. https://www.iata.org/pressroom/facts_figures/fact_sheets/Documents/fact-sheet-fuel.pdf (2017). Accessed 5 Jan 2018
2. Chati, Y. S., Balakrishnan, H.: Statistical Modeling of Aircraft Engine Fuel Flow Rate. International Council of the Aeronautical Sciences (ICAS). http://www.icas.org/ICAS_ARCHIVE/ICAS2016/data/papers/2016_0619_paper.pdf (2016). Accessed 5 Jan 2018
3. Horiguchi, Y., Kayahara, H., Maeno, J.: Predicting Fuel Consumption and Flight Delays for Low-Cost Airlines. Association for the Advancement of Artificial Intelligence (AAAI). https://www.aaai.org/ocs/index.php/IAAI/IAAI17/paper/view/14591/13722 (2017). Accessed 5 Jan 2018
4. Trani, A., Wing-Ho, F., Schilling, G., Baik, H., Seshadri, A.: A Neural network model to estimate aircraft fuel consumption. In: 4th AIAA aviation technology, integration and operations (ATIO) forum, paper 2004-6401. American Institute of Aeronautics and Astronautics (AIAA) (2004)
5. Turgut, E.T., Rosen, M.A.: Relationship between fuel consumption and altitude for commercial aircraft during descent: preliminary assessment with a genetic algorithm. Aerosp. Sci. Technol. **17**(1), 65–73 (2012)
6. Ertel, W.: Grundkurs Künstliche Intelligenz. Springer, Wiesbaden (2016)
7. Wickramanayake, S., Bandara, D.: Fuel consumption prediction of fleet vehicles using machine learning: a comparative study. In: 2016 Moratuwa Engineering Research Conference (MERCon), pp. 90–95. Institute of Electrical and Electronics Engineering (IEEE) (2016)
8. Menon, A.K.: Large-Scale Support Vector Machines: Algorithms and Theory. http://users.cecs.anu.edu.au/~akmenon/papers/research-exam/research-exampaper.pdf (2009). Accessed 8 Jan 2018
9. Haifeng, Z., Xu-hui, W., Xin-feng, C.: Support vector with ROC optimization method based fuel consumption modeling for civil aircraft. J. Procedia Eng. **99**, 296–303 (2015)
10. Chati, Y., Balakrishnan, H.: A Gaussian process regression approach to model aircraft engine fuel flow rate. In: Proceedings of the 8th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS2017). Institute of Electrical and Electronics Engineering (IEEE) (2017)
11. Sturm, M.: Neuronale Netze zur Modellbildung in der Regelungstechnik. Technische Universitaet Muenchen, Institut für Informatik. https://www.mediatum.ub.tum.de/doc/601680/601680.pdf (2000). Accessed 5 Jan 2018
12. Bol, G., Nakhaeizadeh, G., Vollmer, K.-H.: Datamining und Computational Finance. Springer, Berlin (2000)
13. Moisen, G.G.: Classification and regression trees. Encycl. Ecol. **1**, 582–588 (2008)
14. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classification and Regression Trees. Taylor & Francis Ltd., London (1984)
15. Alpaydin, E.: Introduction to Machine Learning. MIT Press, Cambridge (2004)
16. Flach, P.: Machine Learning: the Art and Science of Algorithms that Make Sense of Data. Cambridge University Press, Cambridge (2012)
17. Breiman, L.: Random Forest. Mach. Learn. **45**(1), 5–32 (2001)
18. Quinlan, J.R.: Learning with continuous classes. In: Adams, D.E., Sterling, E. (eds.) Proceedings AI 1992, pp. 343–348. World Scientific, Singapore (1992)
19. NASA DASHlink: National Aeronautics and Space Administration (NASA). https://www.c3.nasa.gov/dashlink/ (2017). Accessed 5 Jan 2018
20. Runkler, T.A.: Data Mining. Springer Vieweg, Wiesbaden (2015)
21. Baumann, S., Gnisia, M., Feifel, P.: Identifikation und Behandlung von Ausreissern in Flugbetriebsdaten für Machine Learning Modelle. Deutsche Gesellschaft für Luft- und Raumfahrt—Lilienthal-Oberth e.V., urn:nbn:de:101:1-201812211207138374462 4 (2018)
22. McCaffrey, J.: How to standardize data for neural networks. Vis. Studio Mag. https://visualstudiomagazine.com/articles/2014/01/01/how-to-standardizedata-for-neural-networks.aspx (2014). Accessed 8 Jan 2018
23. Ross, S.M.: Statistik fuer Ingenieure. Elsevier, Muenchen (2006)
24. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)
25. Nunes da Silva, I., Hernane Spatti, D., Andrade Flauzino, R., Bartocci Liboni, L.H., Franco dos Reis Alves, S.: Artificial Neural Networks. Springer, Berlin (2017)
26. Du, Y., Wu, J., Yang, S., Zhou, L.: Predicting vehicle fuel consumption patterns using floating vehicle data. J. Environ. Sci. **59**, 24–29 (2017)
27. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)
28. Baklacioglu, T.: Modeling the fuel-flow rate of transport aircraft during flight phases using genetic algorithm-optimized neural networks. J. Sci. Technol. **49**, 52–62 (2016)

[2] See: https://aerospace.honeywell.com/en/pages/godirect/ (Accessed 8 February 2018).

[3] See: https://services.airbus.com/maintenance/expertise-and-other-services/skywise/skywise/ (Accessed 8 February 2018).

[4] See: https://www.lufthansa-technik.com/aviatar/ (Accessed 8 February 2018).

29. Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., Schwabacher, M.: Metrics for evaluating performance of prognostic techniques. In: 2008 International Conference on Prognostics and Health Management, pp. 1–17. Institute of Electrical and Electronics Engineering (IEEE) (2008)
30. Ryerson, M. G., Hansen, M., Hao, L., Seelhorst, M.: Landing on empty: estimating the benefits from reducing fuel uplift in US Civil Aviation. Environmental Res. Lett. **10**, 1–11 (2015). http://www.iopscience.iop.org/article/10.1088/1748-9326/10/9/094002/pdf. Accessed 8 Jan 2018
31. Rebhahn, E.: Energiehandbuch. Springer, Berlin (2002)