Aalto University
School of Engineering
Degree Programme in Mechanical Engineering

Tuomas Harju

Derivation of Aircraft Performance Parameters Applying Machine Learning Principles

Master's Thesis
Espoo, May 29, 2017

Supervisor: Professor Jukka Tuhkuri
Advisor: Pasi Koho M.Sc. (Tech.)

Aalto University
School of Engineering

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi

Abstract of master's thesis

Author	Tuomas Harju
Title of thesis	Derivation of Aircraft Performance Parameters Applying Machine Learning Principles
Degree programme	Mechanical Engineering
Major/minor	Aeronautical Engineering
Code	K3004
Thesis supervisor	Professor Jukka Tuhkuri
Thesis advisor(s)	M.Sc. (Tech.) Pasi Koho
Date	29.05.2017
Number of pages	42+3
Language	English

Abstract

To obtain fuel consumption reductions in margin of 5 %, at most, the functions that provide the performance parameters to the fuel consumption optimization problem require enhanced accuracy. The aircraft parameters used in calculation of the consumption of fuel during flight are usually provided in table form. Thus, their utilization in computer software calculations requires application of statistical methods. This thesis explores the usage of machine learning methods in modelling of the data to obtain more accurate models. The data tables are presented in the Aircraft Flight Manual. The datasets used in this thesis are Thrust Specific Fuel Consumption (TSFC) and Cruise Fuel Flow (CFF).

In this study, we select three candidate algorithms for analysis. The Enhanced Adaptive Regression Through Hinges (EARTH) algorithm, based on a trademarked Multivariate Adaptive Regression Splines (MARS) algorithm, Random Forest Regression (RFR) and Kernel Ridge Regression (KRR) are each used to analyze both datasets. An initial analysis gives insight to the algorithm, while a parameter optimization is conducted to obtain the optimal parameters for each algorithm. Additionally, the datasets are divided into training and testing sets in the optimization phase to reduce the effect of overfitting. With the optimal parameter combinations established, the machine learning models are validated using validation plots.

The optimal algorithm is proposed for both datasets according to the accuracy of the prediction. Also, the computational time required for each algorithm is evaluated, but it is not a deciding factor in algorithm selection, due to the nature of the problem. The KRR algorithm is found to not accurately model the dataset with chosen kernel, Radial Basis Function (RBF). Moreover, the optimal parameters obtained from the analysis for RFR render the algorithm used to deviate from accurately representing RFR. With these limitations, and the fact that EARTH algorithm modelled both datasets most accurately, EARTH is proposed as the optimal algorithm for these datasets.

Keywords parameter modelling, regression analysis, machine learning, fuel consumption

Aalto-yliopisto
Insinööritieteiden
korkeakoulu

Aalto-yliopisto, PL 11000, 00076 AALTO
www.aalto.fi
Diplomityön tiivistelmä

Tekijä Tuomas Harju		
Työn nimi Lentokoneen suoritusarvoparametrien selvittäminen käyttäen koneoppimisen periaatteita		
Koulutusohjelma Konetekniikka		
Pää-/sivuaine Lentotekniikka		**Koodi** K3004
Työn valvoja Professori Jukka Tuhkuri		
Työn ohjaaja(t) Diplomi-insinööri Pasi Koho		
Päivämäärä 29.05.2017	**Sivumäärä** 42+3	**Kieli** englanti

Tiivistelmä

Jotta saavutetaan 5 % marginaalissa olevia polttoainesäästöjä, vaaditaan polttoaineen kulutuksen optimoinnin suoritusarvoparametrifunktioissa suurta tarkkuutta. Lentokoneen polttoaineen kulutuksen suoritusarvoparametrit annetaan usein taulukkomuodossa. Tästä johtuen, niiden hyödyntäminen tietokonelaskelmissa vaati tilastotieteen menetelmien käyttöä. Tässä työssä tutkitaan koneoppimenetelmien käyttöä datan mallintamisessa tarkempien mallien saamiseksi. Käytetyt datataulukot on esitelty lentokäsikirjassa (Aircraft Flight Manual, AFM). Työn datasetit koostuvat työntövoimakohtaisesta polttoaineenkulutuksesta (Thrust Specific Fuel Consumption, TSFC) ja matkalennon polttoaineenkulutuksesta (Cruise Fuel Flow, CFF).

Työssä valittiin kolme algoritmia analyysiin. Datasetit analysoidaan RandomForest -regressiolla (Random Forest Regression, RFR), Kernel Ridge -regressiolla (Kernel Ridge Regression, KRR) ja EARTH-algoritmilla (Enhanced Adaptive Regression Through Hinges), joka pohjautuu patentoituun MARS-algoritmiin (Multivariate Adaptive Regression Splines). Alustava analyysi antaa tietoa algoritmien toiminnasta ja parametrien optimoinnilla selvitetään jokaiselle algoritmille optimikombinaatio parametreista. Lisäksi datasetit jaetaan koulutus ja testaus setteihin, jolla vähennetään ylisovittamisen (overfitting) vaikutusta. Kun optimaaliset yhdistelmät parametreille on selvitetty, validoidaan koneoppimalli kuvaajilla.

Lopuksi molemmille dataseteille suositellaan algoritmia ennusteen tarkkuuden perusteella. Laskenta-aika algoritmien välillä tarkastellaan, mutta sitä ei pidetä ratkaisevan tekijänä. Analyysissä huomattiin, että KRR-algoritmi ei mallinna dataa oikein valitulla kantafunktiolla (Radial Basis Function, RBF). Myös RFR:n optimaalisissa parametreissa huomattiin ongelmia, niiden muuttaessa käytetyn algoritmin toimintaa niin, että se ei enää mallintanut dataa kuten RFR:n todellisuudessa kuuluisi. Näiden rajoitusten ja EARTH-algoritmin paremman tarkkuuden johdosta, EARTH:ia suositellaan käytettäväksi näiden datasettien mallintamisessa.

Avainsanat parametrimallinnus, regressioanalyysi, koneoppi, polttoainekulutus

Acknowledgments

Firstly, I would like to thank my supervisor, professor Jukka Tuhkuri, for accepting to supervise the thesis and asking the right questions to help me proceed with the work to the right direction.

I would also like to thank Falconet Systems Oy, especially Pasi Koho and Leo Nyman, for giving me the opportunity to write this thesis to support their project and for the feedback they provided me throughout the process.

Lastly, I like to thank my family for the continuous support through my entire studies, and special thanks to Sara for encouragement and understanding during the thesis.

Espoo, May 29, 2017

Tuomas Harju

Abbreviations and Acronyms

AFM	Aircraft Flight Manual
BF	Basis Function
CFF	Cruise Fuel Flow
CS-25	Certification Specifications for Transport Category Aircraft
EARTH	Enhanced Adaptive Regression Through Hinges
EASA	European Safety Agency
FAA	Federal Aviation Authority
FCOM	Flight Crew Operational Manual
GCV	Generalized Cross-Validation
KRR	Kernel Ridge Regression
MARS	Multivariate Adaptive Regression Splines
MSE	Mean Squared Error
RFR	Random Forest Regression
RBF	Radial Basis Function
RSS	Residual Sum of Squares
SAM	Spectral Angle Mapper
SVR	Support Vector Regression
TSFC	Thrust Specific Fuel Consumption

Contents

Abbreviations and Acronyms

1 Introduction **1**
1.1 Problem Statement 1
1.2 Structure of the Thesis 2

2 Datasets and Methods **3**
2.1 Aircraft Performance in General 3
2.2 Fuel Consumption Datasets 5
2.3 Parameter Modelling Methods 6
2.4 Machine Learning Algorithm Selection 7
2.5 Data Partitioning 9

3 Machine Learning Methods **13**
3.1 Enhanced Adaptive Regression Through Hinges 13
3.2 Random Forest Regression 14
3.3 Kernel Ridge Regression 16
3.4 Algorithm Applications 17
3.5 Algorithm Performance Analysis 19

4 Results **23**
4.1 Thrust Specific Fuel Consumption 23
4.2 Cruise Fuel Flow 25
4.3 Summary 25

5 Evaluation **27**
5.1 Initial Analysis 27
5.2 Parameter Optimization 28
5.2.1 Optimization for TSFC Dataset 28
5.2.2 Optimization for CFF Dataset 30

6 Discussion **33**

6.1 Datasets 33

6.2 Machine Learning Algorithms 33

6.3 Results and Evaluation 34

6.4 KRR Algorithm with Polynomial Kernel 35

7 Conclusions **39**

Appendix 1. CFF Dataset Validation Plots, 3 pages

1 Introduction

This master's thesis is conducted in cooperation with Falconet Systems Oy, a Finland based software company developing aircraft performance management software. The company's interest is finding an optimal machine learning algorithm, that predicts the aircraft performance parameters as accurately as possible, using the data given in the Aircraft Flight Manual (AFM).

The possible savings from more efficient fuel management for a customer are in the range from 1 % up to 5 %. This incurs need for high accuracy in the performance modelling. The AFM is used as the basis for the performance parameters, as it is the Flight Safety Authority mandated publication that is required for the certification and operation of the aircraft. The selected performance parameters compiled in the AFM comprise the datasets used in the study.

The study is roughly divided in three parts. First part is to find suitable machine learning algorithms to be used in the study. During the few decades of machine learning history, a wide collection of algorithms has been developed for different applications. A literature research was conducted to obtain possible algorithms to test the datasets on. However, given the nature of machine learning algorithms, evaluation of the algorithms universally is difficult [1]. Secondly, the datasets are used as an input to the chosen algorithms. Certain algorithms require additional tuning parameters and the optimal setup is configured in this part. Lastly, the obtained results are evaluated using predetermined metrics. The most efficient algorithm is recommended for each dataset.

1.1 Problem Statement

The air transport industry is extremely competitive business. The aircraft operators are constantly trying to find new ways to reduce the costs of the operations and increase revenue. Currently, approximately 20% of the operating costs are caused by the fuel consumption during a flight, being as high as 35% in 2008 [2]. Depending on the mission, an airliner consumes several ten thousand kilograms of aviation fuel. Thus, a relatively small reduction percentage is converted to considerable reduction in actual consumed mass and consequently in fuel cost.

To obtain cost savings from improved fuel management, the aircraft operators

require enhanced flight path planning. As the aircraft consumes fuel, it becomes lighter. The reduction in weight raises the optimum flight altitude of the aircraft. Additionally, to calculate the actual fuel consumption, there are restrictions, for example air traffic control related limitations, that must be considered. This leads to a rather complicated optimization problem.

The optimization difficulty is further increased due to lack of extensive data to model the aircraft performance. The source of performance data available for aircraft operators is the AFM. The data contained there is usually provided in a tabular form, thus making simple transferring of the data to an optimization program inconvenient and not discrete enough. To be able to utilize the data in the AFM, it must be converted to a function form.

This thesis explores the prospect of using machine learning methods to model the data contained in the AFM. Three different algorithms are chosen to be used in converting the tabulated data into useful functions with which to predict the values of the performance parameters. Emphasis is given to the accuracy of the prediction due to small improvement margins in fuel consumption.

1.2 Structure of the Thesis

The thesis reflects the structure of the study. Firstly, in Section 2, we introduce the aircraft performance parameters, datasets used in the study and, according to the characteristics of the datasets, select three appropriate parameter modelling methods for the testing and explain requirement for data partitioning. Next, in Section 3, we concentrate on the three methods and describe them in detail, including mathematical background and their application, and define the algorithm performance metrics. The three methods are then used to model the aircraft parameters from the datasets. The results are presented in Section 4.

In Section 5, the different models obtained from the three machine learning methods are evaluated using common metrics determined in Subsection 2.5. Furthermore, the most accurate machine learning algorithm is proposed for each dataset. Section 6 consists of discussion on the case-specific characteristics in the thesis as well as limitations of the methods and results. Finally, Section 7 summarizes the thesis, including used dataset and methods, results of the modelling, and evaluation and discussion of the results.

2 Datasets and Methods

This chapter presents the datasets and methods used in the study. Firstly, we explain aircraft performance parameters in general. Then, we briefly describe the aircrafts that are analyzed, the parameters given and the parameters we are interested in modelling. Also, we identify the requirements for possible machine learning methods and introduce the three methods chosen for the analysis.

2.1 Aircraft Performance in General

Aircraft performance means the ability to which the aircraft meets the requirements set for various parts of the flight mission. For example, as dictated by European Aviation Safety Agency (EASA) airworthiness regulations, Certification Specification for transport category aircraft (CS-25), in a normal climb the aircraft shall achieve a climb gradient of 3,2 % with additional requirements to climb speed and engine power settings [3]. In addition to regulatory performance requirements, the designing and operating of an aircraft largely depend on the performance parameters.

To operate aircrafts with economical success, the operator utilizes the performance data provided by the aircraft manufacturer to perform mission planning as well as mass and balance calculations. The manufacturer uses the performance parameters to comply with authority regulations and to optimize the aircraft design. Thus, the performance parameters in design consist of a more wide selection of parameters describing the characteristics of the aircraft. Next, we present three parameters that are important in designing and also operating aircrafts.

One of the most basic parameters concerning aircrafts is the thrust, the forward propulsive force generated by the engines. In level flight, the thrust must equal to drag and, generally, it is presented as

$$T = D = \frac{1}{2}\rho V^2 C_D S, \tag{1}$$

where ρ is the density of the air, V is the airspeed and S is the projected wing area. In Equation 1, C_D is the drag coefficient that is usually presented in a polar form

$$C_D = C_{D_0} + KC_L^2, \tag{2}$$

where C_{D_0} is the zero-lift drag and C_L is the lift coefficient, that is dependent on e.g. angle of attack and usage of high-lift devices, such as flaps. The

coefficient K in Equation 2 is presented as

$$K = \frac{1}{A\pi e}, \tag{3}$$

where A is the aspect ratio of the wing and e is the Oswald's efficiency factor, a knock-down factor taking into account separation drag and the unidealities regarding lift distribution [4].

Another, a more complex, parameter is the cruise range with constant airspeed for jet engine aircraft. It is derivative parameter defined as

$$X\left(V\right) = 2\frac{VE_{max}}{c}\arctan\left(\frac{\left(W_i - W_f\right)V^2V_{R_i}^2}{V^4W_i + V_{R_i}^4W_f}\right), \tag{4}$$

where subscript i refers to the starting point of the inspection interval and subscript f refers to the ending point. Additionally, in Equation 4, the E_{max} is the maximum lift-to-drag ratio, c is the specific fuel consumption with dimension $1/s$ and W is the weight of the aircraft. The reference speed V_R is defined as

$$V_R = \sqrt{\frac{2W}{\rho S}}\sqrt[4]{\frac{K}{C_{D_0}}}, \tag{5}$$

which corresponds to the range when flying with maximum lift-to-drag ratio. The derivation of Equation 4 starts from generic momentary equilibrium equations [5].

The parameter of interest in this thesis is the fuel consumption of the aircraft. The Thrust Specific Fuel Consumption (TSFC) is defined as

$$TSFC = C_T = 8,435\frac{V}{\eta_{\mathrm{TOT}}H}, \tag{6}$$

where H is the heating value of the fuel and η_{TOT} is defined as

$$\eta_{\mathrm{TOT}} = \frac{TV}{\dot{m_F}H}, \tag{7}$$

where $\dot{m}_F$ is the mass flow of the fuel [6]. The second parameter considered in this thesis is the cruise fuel flow (CFF), which is $\dot{m}_F$ while in cruise phase of the flight. In Equation 6, the constant is a conversion constant when V is provided in m/s and H in kcal/kg, in which case the C_T yields fuel consumption in 1/h. This function for TSFC works really well on straight turbojets, engines where the entire airflow that travels through the engine travels through the ignition chamber, however, modern airliners are equipped with turbofan engines where only a part of the air travels through the ignition chamber, therefore affecting the the value of C_T more based on the airspeed compared to turbojets. [6]

2.2 Fuel Consumption Datasets

The aircraft types analyzed in the thesis are the Airbus A330-300 and Boeing B737-700. Both aircraft are twin-engine, however, the A330 has slightly longer range. Furthermore, both types are popular aircrafts with over 750 produced A330's as of 1992 and over 1000 B737's as of 1997. The data sets being analyzed in the study are composed of AFM data tables. The AFM is the official source for performance parameters for aircraft operators. It is a required document to obtain aircraft type certificate according to, for example, the airworthiness standards for transport category aircraft of EASA [3] in Europe and Federal Aviation Authority (FAA) [7] in the United States.

The data, and the presented format it is in the AFM, depend on the aircraft manufacturer. For Airbus A330 and Boeing B737 the relevant data used in the study is found in the Flight Crew Operating Manual (FCOM), that supplements the AFM with more in-depth performance data. The tables for cruise performance, climb performance and descent performance are converted into usable format for further data analysis. Moreover, the engine data for B737 has been complemented with data from the engine manufacturer. In this study, the datasets are used to model two variables utilized in the software to optimize the flight path and fuel consumption: CFF for A330 and TSFC for B737.

As described in Equation 6 and Equation 7, the TSFC and, therefore, the CFF are dependent on several variables

$$C_T = f(V, H, T, \dot{m}_F). \tag{8}$$

However, the data provided in the AFM does not include all of the parameters in Equation 8 and includes parameters not in the equation. The parameters incorporating the TSFC and CFF dataset are compiled in Table 1. In addition to consisting of different parameters for independent variables, the number of data points varies greatly between datasets, from 69 in the TSFC to 937 in the CFF. The datasets are also illustrated in Figure 1. Note, that the CFF dataset in Figure 1 (b) includes only data points where the deviation from standard atmosphere temperature $\Delta T_{ISA} = 0$ and the mass of the aircraft $m = 140000$kg, however, the distribution of the rest of the data points is similar.

Furthermore, Datasets are compiled in an Excel spreadsheet format for ease of import to Matlab and Python, which are used for the analysis stage of the thesis, as well as the analysis of the results. Each independent variable and

Table 1: Performance parameters comprising the fuel consumption data in AFM

variable	**description**	**unit**
TSFC dataset		
TSFC	Thrust Specific Fuel Consumption	g/kN
h	altitude	m
Ma	Mach number	-
CFF dataset		
CFF	Cruise Fuel Flow	kg/s
ΔT_{ISA}	Deviation from ISA temperature	$^{\circ}C$
m	mass of the aircraft	kg
h	altitude	m
V_{TAS}	True Airspeed	m/s

the dependent variable represent one column and each data point is a single row on the spread sheet.

2.3 Parameter Modelling Methods

The datasets discussed in Subsection 2.2 require processing to be useful in prediction of aircraft performance parameters in a software. One method to model a dataset of one dependent variable and one or more independent variables is to utilize regression analysis, a branch of statistical modeling. The models in regression analysis include parametric methods, such as multiple linear regression, nonlinear regression, and partial least squares regression, that are capable to model a wide variety of datasets with similar characteristics as TSFC and CFF dataset in this thesis.

The statistical regression models are usually parametric in nature, meaning that before applying such a model, certain amount of knowledge on the interactions between the variables is required. Since we have no knowledge of the exact interactions, the problem of optimizing the parameters in these models would be rather extensive. Therefore we are searching for nonparametric methods, that is, methods that utilize the dataset to define the interactions of the variables.

To obtain the most accurate model available, an optimization of the model created is required. Thus, we utilize machine learning algorithms to deduce interactions between variables in the data to construct a useable function for the output parameters. Machine learning incorporates mathematical principles of statistical modelling with computer science for the optimization of the training procedure and the interpretability of the model [1]. With applicable machine learning algorithm, we require little to no prior knowledge on the dataset

Different machine learning algorithms are used for different applications. They may be used to learn association rules from datasets, for example in customer behavior modelling. Unsupervised learning is used to form hidden structures in data, this includes methods such as clustering. Classification and regression methods are referred to as supervised methods. They both require a labeled dataset with a output variable and one or more input variables. The output in classification is the predicted class, often binary or Boolean, but may also include more than two classes. Regression output is in a continuous value format, much like in the datasets used in this thesis. Machine learning methods that adapt to new data observations are called reinforcement learning methods.

From dataset characteristics, we deduce that we require a regression algorithm. However, the problem with machine learning algorithms is, that it is difficult to predict which algorithm is suitable for a specific case [1]. Moreover, regarding the thesis, there are two main aspects of performance to evaluate between different algorithms: accuracy and execution time. Due to the nature of the problem in this thesis, we emphasize accuracy in model selection.

2.4 Machine Learning Algorithm Selection

Since the first computer learning program was written by Arthur Samuel in 1952, many machine learning algorithms have been developed [8]. Because the variables we are interested in are continuous, we are searching for a regression type algorithm as discussed in Subsection 2.3. Figure 2 gives insight, although rather limited, on how to choose an algorithm. The segment "regression" in the figure is the domain of our problem. However, we only select one of the three algorithms used in the thesis from Figure 2. In addition, we select two others not listed that are applicable.

The first machine learning method we selected is the Multivariate Adaptive Regression Splines (MARS). MARS is initially developed by Friedman [10]

and the term is copyrighted to Salford Systems. However, multiple open source implementations for MARS exist. These are often referred to as Enhanced Adaptive Regression Through Hinges (EARTH) methods. The term EARTH is used in this thesis as the algorithm applied to the datasets is an open source version. Nonetheless, the basic principle remains identical to MARS.

MARS utilizes the data to deduce the interactions between independent and dependent variables and produces a model that is continuous and has continuous derivates, therefore eliminating the need to make assumptions on the interactions and providing attractive output considering the prediction program. Additionally, the original introduction of MARS includes an expectation of good performance on datasets with 50 ... 1000 data points and 3 ... 20 independent variables. [10] Subsection 3.1 describes the EARTH method in detail.

Second method selected is the Random Forest Regression (RFR), also a copyrighted product of the Salford Systems. Breiman [11] introduced the Random Forests method, that implements a concept of independently distributed random vector predictors to grow each decision tree. RFR performs comparatively to many other powerful regression methods, like boosted trees, with easier training and model tuning.

According to the web source for Random Forest, the algorithm is the most accurate of current algorithms, a bold claim unsupported by absence of last update to the web page. Furthermore, the same source states that RFR should not overfit. [12] Based on these properties, RFR should be the optimal choice of algorithm and, therefore, is included in the thesis. The principle behind RFR is investigated in Subsection 3.2.

The third method, chosen from Figure 2, is the Kernel Ridge Regression (KRR). The KRR may be considered as simplified version of the popular Support Vector Regression (SVR) algorithm [13]. KRR applies the kernel trick to ridge regression, consequently enabling training of nonlinear function using linear regression. Basics on how to implement the kernel trick to a ridge regression method is presented in work by Murphy [14] and the full method described in Subsection 3.3.

The three methods chosen, EARTH, RFR and KRR, are adequately different to produce interesting comparison while being established methods known for their accuracy in modelling. However, as already stated, every data set analyzed using machine learning algorithms will achieve optimal solution with different algorithm that is discovered by testing different ones [1]. Especially, KRR is sensitive to algorithm tuning parameter and thus sufficient

concentration to selection is required.

As already mentioned, the platforms on which the machine learning algorithms are executed comprise of Matlab software by Mathworks and Python programming language. The RFR is readily incorporated into Matlab's Statistics and Machine Learning Toolbox [15]. The TreeBagger function, with right parameters, behaves as RFR. KRR is not included in the Toolbox as default. However, there is a KRR package available in the Mathworks community's File Exchange section, written by Joseph Santarcangelo [16]. The EARTH algorithm was executed using the pyEarth package [17] for Python. Description of the parameters affecting the algorithms is included in Subsection 3.4.

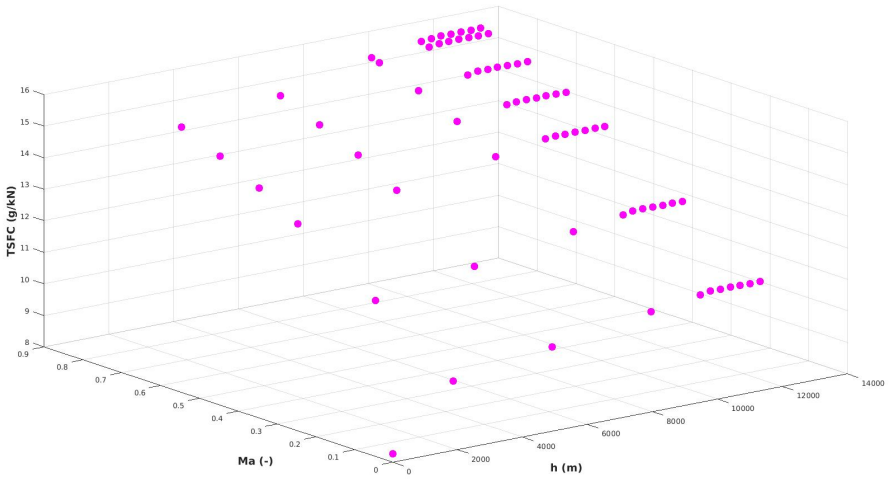
2.5 Data Partitioning

The three chosen algorithms have multiple parameters with which to fine tune the model. At first, we executed the algorithms with the default values, inputting only those parameters not having a default value. This, however, resulted in rather inaccurate models. Then, after successive tries, we settled with a set of parameters as described in Section 3. We evaluated the model by comparing the original output values to the output values gained from the machine learning algorithm models. Although, the difference is at best less than 1 % across the dataset, using the complete dataset for both modelling and testing may yield results with rather large bias.

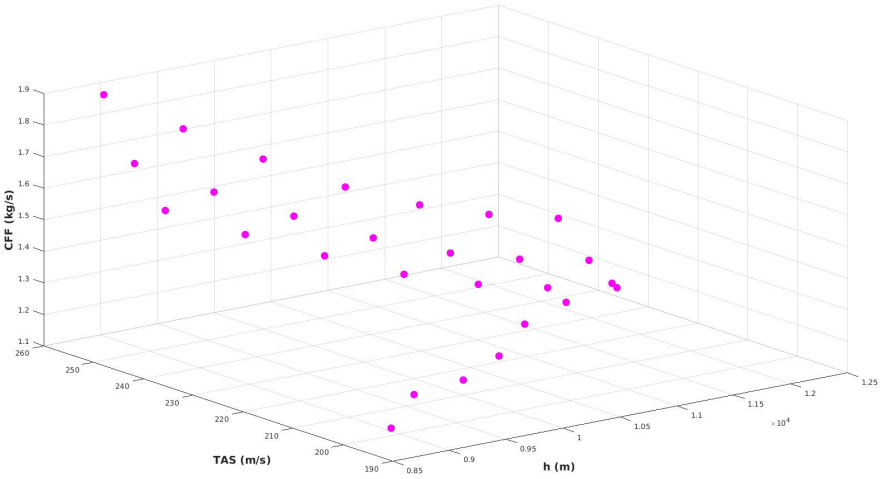
To reduce the effect of bias error, datasets are usually divided into two parts: training dataset and testing dataset. The training dataset is used to construct the model and the testing dataset is used to evaluate the performance of the model. Optimally, the dataset is large enough that a portion of the data may be divided to each subset. However, even the CFF dataset is too limited for good results, thus we need to use data partitioning methods to divide the data so that we have multiple training and testing sets. There are several different data partitioning methods, from which we have chosen the k-fold cross-validation to divide the data. K-fold cross-validation divides the data into k equally sized subsets. Each subset is used one after another as the testing dataset while the rest are used in training the model. A common value of $k = 5$ is used for the analysis of both datasets.

Consequently, using K-fold implies that we have 5 different models for each algorithm on which we must optimize the algorithm tuning parameters. To obtain the most accurate model, the tuning parameters of the algorithms are selected applying multiple combinations of the parameters for the

optimization. The parameters are presented in Subsection 3.4 and the combinations are chosen using insight gained during initial analysis. Finally, we choose the most accurate model from each algorithm for a comparison to decide on the optimal algorithm for a particular dataset.



(a) TSFC data



(b) CFF data limited to $\Delta T_{ISA} = 0$ °C and m = 140 000 kg

Figure 1: The data obtained from AFM fuel consumption data tables

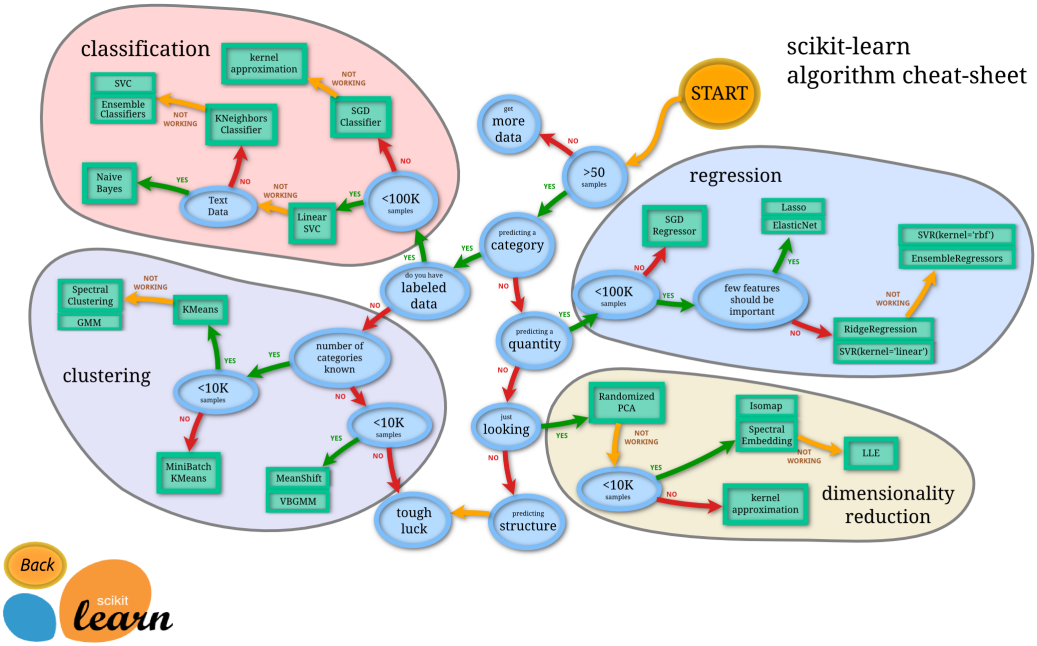


Figure 2: Machine learning method selection cheat-sheet from Scikit-learn [9]

3 Machine Learning Methods

This section describes the mathematical background of the machine learning methods used in the thesis. Furthermore, this section includes details of the algorithm applications for the chosen machine learning methods and the performance metrics used to evaluate the algorithms.

3.1 Enhanced Adaptive Regression Through Hinges

This subsection describes the EARTH method. EARTH regression differs from many other regression models, as it does not try to fit a single function, to model the dependencies between the variables in the data. Instead, the earth regression utilizes piecewise polynomial functions, or basis functions (BF), that are given between knots. Knots separate data regions in the dataset. One of the assets of EARTH regression is, that it solves independently the knots and, thus, requires no prior knowledge of the distribution of the data.

The BFs include elementary and complex functions. The complex functions enable interactions between variables in multivariate cases. An elementary BF comprise of a pair of equations of following form:

$$BF = MAX(0, x - t) \; or \; BF = MAX(0, t - x). \tag{9}$$

These functions are called hinge functions, where x is the independent variable and t is a knot where the split is made. Which of the BFs in Equation (9) is chosen, depends on the region of the data being analyzed, more specifically so that the BF is a positive number or 0. A complex BF is a product of more than one elementary BFs and the degree of complexity is one of the tuning parameters. The derivation of the BF is similar to linear regression, however, only the specific region of data is used.

The EARTH model is constructed in two phases: the forward pass and the pruning pass. In the forward pass, the algorithm iterates the optimal variable-knot combination that improves the model the most. The improvement of the model is measured in a decrease in mean squared error (MSE). The procedure of searching for the optimal variable-knot combination is repeated for a number of times until the limit of maximum number of BFs is reached or the increase in accuracy is below established threshold. [18] Both the maximum number of BFs and the threshold are tuning parameters in the algorithm. Each additional BF increases the model accuracy and includes

an additional constraint for the subsequent searches for the variable-knot combinations, therefore, increasing model complexity [18].

The pruning pass is an elimination procedure. The algorithm begins with the model with all BFs from the first phase. Then, the algorithm searches for the BF that has the least negative effect on the model if removed. The residual sum of squares (RSS) is used to measure the effect on the model. Next, the model is refitted and the procedure repeated until all BFs are removed. The sequence of removal produces a collection of possible models. [18]

From the collection of multiple models the most accurate is chosen using generalized cross-validation (GCV) as the metric. [17] GCV is defined as

$$GCV\left(\lambda\right) = \frac{\sum_{i=1}^{N}\left(y_i - \widehat{f_\lambda}\left(x_i\right)\right)^2}{\left(1 - M\left(\lambda\right)/N\right)^2}, \tag{10}$$

where λ is the number of terms, N is the size of the dataset, $\widehat{f_\lambda}(x_i)$ is the prediction in data point i and $M(\lambda)$ is the penalty term associated with model complexity. Hastie et al. [19] estimate the penalty using approximation $M(\lambda) = r + cK$, where r is the number of linearly independent BFs, K is the number of knots and c is the weight applied to knot selection. A value of $c = 3$ is used for pruned models [19].

3.2 Random Forest Regression

This subsection presents the RFR method applied in this thesis. Multiple tree based regression methods exist, all of them utilizing succession of logical nodes to grow a tree. A tree is a collection of logical nodes that are used to determine which end node, or leaf, is used as the predicted value based on the input variables from the dataset used. A simple regression tree is presented in Figure 3.

Tree methods usually have rather large variance which reduces their accuracy. Bagging of trees, meaning growing multiple trees and averaging over them, improves accuracy by reducing the variance induced error. The estimated variance of the average of n identically distributed variables is

$$Var\left(\overline{X}\right) = \rho\sigma^2 + \frac{1-\rho}{n}\sigma^2, \tag{11}$$

where ρ is the average pairwise correlation and σ^2 is the variance of the variables.

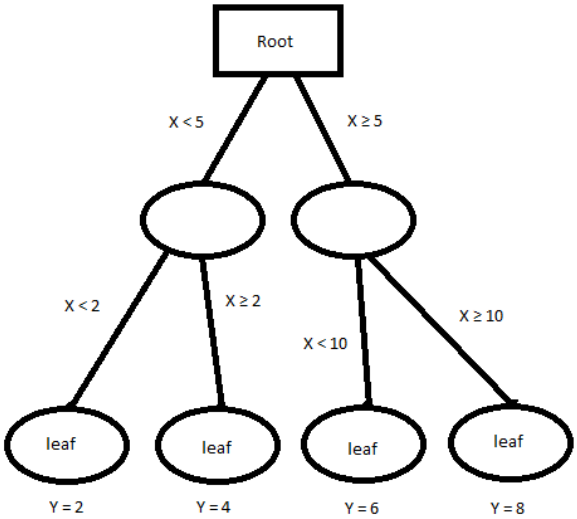


Figure 3: A simple regression tree

RFR, being a derivative of tree bagging, averages the regression of multiple trees with reduced correlation, to produce a model with reduced variance over standard bagging. This is achieved by growing each tree from bootstrapped sample, selecting a random subset of predictors in the original dataset for each node and choosing the best one to split it. By choosing a subset of predictors rather than all of them reduces the ρ in Equation 11. The trees are grown until the depth limited in the algorithm is reached, which is represented by the minimum number of training data points in a terminal node. [19]

The trees that have been grown, are collected for use in a prediction, which for a regression forest, is the average of predicted values of all the trees, calculated as

$$\widehat{f}_{rf}^{B}(x) = \frac{1}{B}\sum_{b=1}^{B} T(x;\Theta_b), \tag{12}$$

where B is the maximum number of trees, x is an new data point for which the prediction is made, $T(x;\Theta_b)$ is a trained regression tree from the random forest and Θ_b is the unique vector that defines the parameters used in growing the bth tree.

RFR accuracy can be improved with sufficiently large number of trees. When B is large, the second term vanishes and Equation (11) reduces to

$$Var\left(\overline{X}\right) = \rho\sigma^2. \tag{13}$$

Moreover, Equation (13) may be expanded to apply to the RFR model by choosing single target point x to consider and using the sampling correlation and sampling variance of the trees as the ρ and σ^2, respectively. The variance is then calculated as

$$Var\,\widehat{f}_{rf}^{B}(x) = \rho(x)\,\sigma^2(x), \tag{14}$$

where $\rho(x)$ is the sampling correlation between a pair of trees, $\sigma^2(x)$ is the sampling variance of a single tree and $\widehat{f}_{rf}^{B}(x)$ is presented in Equation (12). The dependence on x indicates that the correlation is dependent on the training set that is used to construct the RFR model.

3.3 Kernel Ridge Regression

This section describes the ridge regression as introduced by Hoerl and Kennard [20], utilizing the kernel trick [14]. Ridge regression is a linear least squares regression, with the Euclidean norm regularization. This is the same as with Support Vector Regression (SVR), however, the loss function used is the squared error loss, instead of ϵ-insensitive loss used in SVMs. [9] Furthermore, different kernels allow the calculation of non-linear interactions with linear regression, potentially reducing computation time and taking more complex interactions into account.

The ridge regression was developed to enhance the multiple regression models using ordinary least squares estimation of the form:

$$\hat{\beta} = \left(X^T X\right)^{-1} X^T Y, \tag{15}$$

where X is the matrix of training independent variables and Y is the matrix of the training dependent variables. The matrix $X^T X$ in Equation (15) is susceptible to error while it deviates greatly from a unit matrix. Therefore, an additional term is included in the ridge regression algorithm:

$$\hat{\beta} = \left(X^T X + kI\right)^{-1} X^T Y, \tag{16}$$

where k is a non-negative coefficient. Equation (16) is the form of the model used in the thesis.

The kernel trick is used to replace the inner products of an algorithm with a call to kernel function [14]. This trick modifies the Equation (16) accordingly

$$\hat{\beta} = \left(\kappa\left(X, X^T\right) + kI\right)^{-1} X^T Y, \tag{17}$$

where $\kappa(X, X')$ is the kernel function. Depending on the problem being analyzed, the optimal choice of kernel varies. In this thesis, we use the

Radial Basis Function (RBF) kernel. RBF kernel has the following form

$$\kappa\left(X, X'\right) = exp\left(-\frac{\|X - X^T\|^2}{2s^2}\right), \tag{18}$$

where s^2, often also denoted with σ^2, is the bandwidth of the kernel and $\|X - X^T\|$ is the Euclidean distance between X and X^T. Consequently, combining Equation (17) with the Equation (18) we obtain the equation used in the study. In addition to the type of the kernel, the algorithm requires input for the values of k and s, as the model is highly dependent on the chosen values.

3.4 Algorithm Applications

The machine learning algorithms used in the thesis are provided in packages as described in Subsection 2.3. The packages enable calls to predefined functions, included in the package, to construct machine learning models. In the simplest case, all that is needed from the user is to define the independent and dependent variable and to input the data into the function. Then, the constructed model may be used to predict the dependent variable from a new dataset.

Usually, however, the functions require tuning to some degree to perform well on a given dataset. First, in our analyses, the algorithms used require choices from the user before the function can be used. For example, the TreeBagger used in RFR analysis requires the number of trees to grow as an input without any default value. The chosen value effects the accuracy and execution time of the algorithm. Secondly, the default values for the parameters might not result in optimal models. Again, using the TreeBagger as an example, the default minimum number of data points per end node is 5 for a regression model, although, as the results in Section 4 dictate, that is not the optimal value for either of the datasets. The fact that this particular parameter should be fine-tuned is also described by Friedman et. al. [19].

The TreeBagger and EARTH functions accept multiple parameters to tune the model, whereas, the KRR requires firstly the kernel type, RBF in this case, and thereafter the parameters required are defined by the kernel. For KRR with RBF type, there are two parameters to input, already introduced in Subsection 3.3. For TreeBagger and EARTH, we have chosen three parameters for input, which are compiled, along with the KRR parameters, in Table 2. Also, listed are the default values for those parameters where defaults are provided. Two of those parameters have default values that

depend on the dataset characteristics, marked with *. If no default value is provided in the package, it is denoted with sign "-".

The Thresh parameter controls the forward pass of the EARTH algorithm by terminating the pass if improvement to the current model is under the Thresh value. For maximum accuracy, Thresh should be set as low as computationally feasible, as every new term increases the accuracy of the model. Max_terms limits the number of terms allowed in the model. As well as possibly limiting the accuracy of the model, this parameter also dictates the amount of memory reserved for the modelling. The default value for Max_terms is one of the two dataset dependent values. The default value is calculated with following equation

$$Max\_terms = \min\left(\frac{2n+m}{10}, 400\right), \tag{19}$$

where n is the number of independent variables and m is the number of data points. Max_degree sets the maximum degree of multiplicity allowed in the model. Higher degree allows the detection of more complicated interaction between variables but too high degree might expose the model to overfitting.

In the RFR, Trees is the only required input parameter. The number of trees grown affects the accuracy by moving the average variance towards the form of Equation (13), thus growing more trees should infinitely enhance the model. However, increase in number of trees also increases the execution time of the RFR algorithm, besides, the increase in accuracy levels off after certain number of trees. An experiment by Oshiro et.al. [21] suggests that, at least for their datasets, the optimal number of trees is between 64 and 128. MinLeafSize adjusts the depth to which the trees are grown by setting the minimum number of data points required in each end node. Setting a lower number increases the complexity and execution time but should produce more accurate model. NumPredictorsToSample is the second of the parameters having a dataset dependent default value. For regression, the default is calculated as follows

$$NumPredictorsToSample = \frac{n}{3}, \tag{20}$$

where n is the number of independent variables, and rounded up to nearest integer. NumPredictorsToSample represents the number of independent variables in a randomly selected subset from which the best one for a split is determined. In addition to any positive integer, up to and including n, the NumPredictorsToSample accepts "all" as valid input and is basically the same as selecting n as input value. Although, if n or "all" is the input, the TreeBagger function does not represent RFR as intended.

Table 2: Input parameters for the machine learning algorithms

ML algorithm	**Parameter**	**Default value**
EARTH	Thresh	10^{-3}
EARTH	Max_terms	* (TSFC: 7,3 & CFF: 94,5)
EARTH	Max_degree	1
RFR	Trees	-
RFR	MinLeafSize	5
RFR	NumPredictorsToSample	* (TSFC: 1 & CFF: 2)
KRR	ker	-
KRR	sigma	-
KRR	Regulation term	-

"*" denotes a dataset specific value and "-" denotes no default value.

For KRR algorithm, there are no default values for the input parameters chosen. The type of kernel, or ker in Table 2, was chosen first, fixing the remaining parameters. In this thesis, we chose the RBF kernel, although the algorithm used also supports linear, polynomial and Spectral Angle Mapper (SAM) kernels. Parameter s, short for sigma in Table 2, was introduced as the bandwidth of the RBF kernel. The magnitude of s affects the contribution of a single training data point has on the model. A lower s value limits the contribution on a smaller region resulting in a more discreet prediction but also exposes it to overfitting. The k_R value, or Regulation term, is the coefficient that the unit matrix in Equation (17) is multiplied by. As already mentioned in Subsection 3.3, while the matrix in Equation (16) deviates from unit matrix it inflicts additional error to the model. Therefore, Regulation term is used to scale the unit matrix to properly decrease or increase the values obtained from the kernel function.

3.5 Algorithm Performance Analysis

Since the knowledge on the effects of dataset characteristics have on the machine learning algorithms was limited at the beginning, an initial analysis was conducted to acquire basic knowledge on the behavior of the algorithm. For this step, the algorithms were analyzed starting from default values and variating a single parameter at a time to observe the algorithm behavior.

Additionally, the datasets used in training and testing were the same, which is highly discouraged, but adequate for the purpose. Nonetheless, the initial analysis was conducted using the entire dataset as training set and subsequently comparing the predicted values against the known values for the dependent variable. An average error for n data points was calculated

$$ERR_{\text{AVG}} = \frac{\sum_{i=1}^{n} ABS\left(\frac{Y_i}{\hat{Y}_i} - 1\right)}{n}, \tag{21}$$

where Y_i is a known value of the dependent variable and $\hat{Y}_i$ is the corresponding predicted value of the dependent value. In addition to average error, we were interested in the maximum error of the predicted values, calculated as follows

$$ERR_{\text{MAX}} = max\left(\left\{\frac{Y_i}{\hat{Y}_i} - 1\right\}_{i=1}^{n}\right). \tag{22}$$

For the initial analysis, average error and maximum error help us to determine the parameter range for the optimization.

To obtain the optimal parameter combination, we conducted a grid search meaning, that we decided on multiple values for the parameters in Table 2, excluding the ker parameter. Restricting factor for the size of the grid was execution time. Since the datasets have three parameters, or two in case of KRR, to optimize, adding of values multiplies the size of the grid. Moreover, as the search is conducted on the dataset divided to $k_R = 5$ folds, the grid is analyzed separately in five training and checking cycles. The grids are presented in Table 3 and Table 4, complemented with the number of possible combinations. Although, there are functions readily available to perform the grid search with, we decided on a in-house script to implement the error functions presented in Equation (21) and Equation (22).

Average error and maximum error indicate how well the algorithm performs on the dataset with a given combination. To decide on a set of parameters, a model was created for each parameter combination from Table 3 and Table 4, respectively for TSFC and CFF, and for the five folds. The best model was chosen based on the lowest ERR_{AVG} value. Additionally, the ERR_{MAX} was recorded for the best model as a simple indicator of deviation along with the total running time of the analysis. Given these results, the best algorithm was proposed from the three alternatives for both datasets.

Table 3: The grids for the search of optimal tuning parameters and number of combinations for the algorithm with TSFC dataset

ML algorithm	**Parameter**	**Grid values**
EARTH	Thresh	[10^{-7}, 10^{-9}]
EARTH	Max_terms	[25, 50, 100, 250, 400]
EARTH	Max_degree	[2, 3, 4]
EARTH combinations 30		
RFR	Trees	[100, 200, 300, 400]
RFR	MinLeafSize	[1, 2, 3, 4]
RFR	NumPredictorsToSample	[1, 2]
RFR combinations 32		
KRR	ker	RBF
KRR	sigma	[10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}, 10^{5}]
KRR	Regulation term	[10^{-8}, 10^{-7}, 10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}]
KRR combinations 70		

Table 4: The grids for the search of optimal tuning parameters and number of combinations for the algorithm with CFF dataset

ML algorithm	**Parameter**	**Grid values**
EARTH	Thresh	[10^{-4}, 10^{-5}]
EARTH	Max_terms	[25, 50, 100, 200]
EARTH	Max_degree	[1, 2, 3, 4]
EARTH combinations 32		
RFR	Trees	[25, 50, 100]
RFR	MinLeafSize	[1, 2, 3, 4]
RFR	NumPredictorsToSample	[1, 2, 3]
RFR combinations 36		
KRR	ker	RBF
KRR	sigma	[10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}]
KRR	Regulation term	[10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}]
KRR combinations 42		

4 Results

In this section, we present the results of the dataset analysis. First, we examine the results of the TSFC initial analysis for each machine learning algorithm and the optimized parameters and the corresponding errors. Secondly, we present the results of the initial analysis and parameter optimization for the CFF dataset. Finally, we summarize the results from all the analyses.

4.1 Thrust Specific Fuel Consumption

The TSFC dataset was analyzed with the three machine learning algorithms. The initial analysis was conducted using parameters that were chosen using "best guess" -method without extensive knowledge of the effect of the parameters to the model, utilizing default parameters where possible. After few test rounds, the first useable parameters were obtained. These parameters are compiled in Table 5.

Using the initial parameters in Table 5, the dataset for TSFC was analyzed. The training set and test set were identical and comprised of the entire dataset. Additionally, the parameters are chosen with the intent that nothing would limit the accuracy of the model. For example, the Thresh parameter for Earth algorithm is set at $1 * 10^{-9}$, resulting the forward pass to create as many terms as possible. The results of the analysis are also presented in Table 5.

The second stage was to perform a parameter optimization on the machine learning algorithms. The datasets were separated to training and testing subsets as described in Subsection 2.5. The grids for the search of optimum parameter combination are presented in Table 3 in Subsection 3.5. Using the in-house script, we created the machine learning models from training sets, predicted the values using test sets and evaluated the accuracy with average error from Equation (21). Performing this with the 5 folds, we could find the optimal solution for all three machine learning algorithms. The optimal parameter combinations and the error values are compiled in Table 6. Comparing the error values for the final results, the EARTH algorithm models the dataset most accurately.

Table 5: Results of the first iteration of TSFC dataset analysis

	EARTH	**RFR**	**KRR**
Parameter 1	Thresh $1 * 10^{-9}$	Trees 50	ker RBF
Parameter 2	Max_terms 400	MinLeafSize 1	sigma 0,5
Parameter 3	Max_degree 4	NumPredictorsToSample "all"	Regulation term 0,005
$\mathbf{ERR_{AVG}}$	0,00118	0,00335	0,00115
$\mathbf{ERR_{MAX}}$	0,00553	0,05256	0,00380

Table 6: Final results of the parameter optimization for TSFC dataset

	EARTH	**RFR**	**KRR**
Parameter 1	Thresh $1 * 10^{-9}$	Trees 200	ker RBF
Parameter 2	Max_terms 50	MinLeafSize 1	sigma 1
Parameter 3	Max_degree 3	NumPredictorsToSample 2	Regulation term $1 * 10^{-5}$
$\mathbf{ERR_{AVG}}$	$5,43 * 10^{-4}$	0,0031	$9,61 * 10^{-4}$
$\mathbf{ERR_{MAX}}$	$9,56 * 10^{-4}$	0,0232	0,0036
Time	278,93 s	88,98 s	0,36 s

4.2 Cruise Fuel Flow

The starting point for CFF dataset analysis was the parameters given in Table 5 for TSFC initial analysis. However, given the different structure of the dataset, some parameters were adjusted. The NumPredictorsToSample -parameter in the RFR algorithm and Max_terms -parameter in the EARTH algorithm were changed to the dataset default values. The change in RFR reverts the algorithm to accurately represent Friedman's RFR. Additionally, the EARTH parameter Thresh was changed to a larger value to reduce the execution time. The parameters for the initial analysis are listed in Table 7.

With the parameters from Table 7, we obtained the results also presented in Table 7. Similarly to the initial analysis of TSFC dataset, the training and testing datasets used in initial analysis of CFF are both comprised of the entire dataset and no parameter optimization was conducted, apart from the minor corrections due to dataset characteristics.

The parameters were optimized in the second stage of the analysis. Also, the CFF dataset was separated into training and testing subsets similarly to TSFC. Using the combinations from Table 4 in Subsection 3.5, we could obtain the optimal combinations and corresponding error values as presented in Table 8. The script was the same as used for TSFC in Subsection 4.1 as well as the equation for average error, Equation (21). Also, the results are similar compared to the TSFC dataset with the EARTH algorithm being the most accurate.

4.3 Summary

In this section, we presented the results of the dataset analyses. The initial analysis was conducted mainly to give insight on the algorithm mechanics. The results, presented in Table 5 and Table 7, of the initial analysis were promising, however, unreliable due to lack of separation of training and testing data.

The parameter optimization started from deciding on a parameter grid. The datasets were modelled using parameter combinations in the grid and on a dataset separated with the k-fold method. This procedure resulted in machine learning models, that include reduced bias and comparable error values to choose the best model with. Results of the final analyses are presented in Table 6 and Table 8, respectively for TSFC and CFF datasets.

Table 7: Results of the first iteration of CFF dataset analysis

	EARTH	**RFR**	**KRR**
Parameter 1	Thresh $1 * 10^{-4}$	Trees 50	ker RBF
Parameter 2	Max_terms 94,5	MinLeafSize 1	sigma 0,5
Parameter 3	Max_degree 3	NumPredictorsToSample 2	Regulation term $5 * 10^{-3}$
$\mathbf{ERR_{AVG}}$	0,00793	0,00614	0,00051
$\mathbf{ERR_{MAX}}$	0,04309	0,02919	0,00266

Table 8: Final results of the parameter optimization for CFF dataset

	EARTH	**RFR**	**KRR**
Parameter 1	Thresh $1 * 10^{-5}$	Trees 50	ker RBF
Parameter 2	Max_terms 25	MinLeafSize 1	sigma 100
Parameter 3	Max_degree 4	NumPredictorsToSample 3	Regulation term $1 * 10^{-8}$
$\mathbf{ERR_{AVG}}$	0,00386	0,0116	0,0040
$\mathbf{ERR_{MAX}}$	0,01081	0,0451	0,0366
Time	250,82 s	57,60 s	32,39 s

5 Evaluation

In this chapter, we evaluate the results obtained from the analyses. The primary objective is to model the aircraft parameters as accurately as possible. Therefore, as we evaluate the results, we emphasize the accuracy parameter and secondarily compare the computational time.

5.1 Initial Analysis

The initial analysis resulted in accurate models for both datasets. KRR performed well on both datasets and EARTH was also accurate on the TSFC dataset. RFR was outperformed by both dataset and EARTH exhibited the worst results in the CFF dataset. The computational time was not of concern in the initial analysis. The models were trained and tested in few seconds on all algorithms.

As seen from Table 5 in Subsection 4.1, EARTH and KRR models yielded similar results on the TSFC dataset, the average error around 0,001. Moreover, the maximum error on any given data point was 0,00553 and 0,00380, respectively for EARTH and KRR. Both were expected to perform well after the parameter optimization.

The RFR, however, performed considerably worse than the two others. While the average error was about thrice as large as the EARTH and KRR values, the maximum error on RFR was an order of magnitude larger with a value of 0,05256. The inferior performance and the fact, that the algorithm was not exactly in accordance with Friedman's RFR, did anticipate that the RFR was not optimal for this dataset. To properly represent RFR, the algorithm was tested with $NumPredictorsToSample = 1$ that resulted in decreased accuracy. Regardless, the parameter optimization was also conducted on the RFR algorithm.

After the initial analysis of the TSFC dataset, we analyzed the CFF. The results for CFF are presented in Table 7 in Subsection 4.2. With this dataset, the EARTH algorithm's performance was greatly reduced. The average and maximum error for EARTH model were 0,00793 and 0,04309, respectively, being the worst of the three algorithms. Furthermore, the RFR, now in accordance with Friedman's RFR, outperformed EARTH with errors of 0,00614 and 0,02919, respectively for average and maximum error.

However, the KRR algorithm had superior performance compared to the other two with average error of 0,00051 and maximum error of 0,00266.

Although, the KRR performed well, the error was almost constant at 0,0005 with only small variation. This invoked suspicion that the model overfitted the dataset. Therefore, KRR would require the data partition to confirm the algorithm performance and its optimal parameters.

5.2 Parameter Optimization

The parameter optimization highlighted the importance of the partition of data into training and testing subsets. The most accurate algorithm was different on both datasets and the values of the error were considerably different from the initial analyses. Furthermore, the computational time grew in importance as the algorithm was executed on a grid of parameter combinations and on the partitioned dataset.

5.2.1 Optimization for TSFC Dataset

The TSFC dataset was analyzed with data partition performed according to the k-fold method with $k = 5$. On a dataset with 69 data points that means approximately 14 data points are allocated to each subset. With such small sets the random partition might affect the results but iterating over five alternatives should produce the most accurate model. The results are compiled in Table 6.

The RFR was analyzed with 32 combinations. The RFR parameter NumPredictorsToSample was still optimal at "all", meaning that the model obtained from the parameter optimization did not actually function as proper Friedman's RFR. Moreover, the accuracy of the RFR was not remarkedly improved from the optimization. The ERR_{AVG} was almost identical with value of 0,0031 and the ERR_{MAX} was reduced to approximately half of the unoptimized model.

KRR algorithm produced the most accurate model for TSFC in the initial analysis. Therefore, KRR was expected to perform well with the parameter optimization. The regulation term was reduced two orders of magnitude, with optimal value of 10^{-5} compared to the 0,005 of the initial analysis. The sigma parameter was largely unaffected, changing from 0,5 to 1 due to the grid. The KRR performance was greatly increased from the parameter optimization as the average error value reduced to $9,61{*}10^{-4}$. The maximum error remained rather constant.

The EARTH algorithm performed also well on the initial analysis, resulting in almost the same average error as KRR and only slightly higher maximum

error. The Max_degree parameter was reduced from 4 to 3 with the parameter optimization and the Max_terms was reduced from 400 to 50, while the Thresh parameter was unaffected. With these parameters, the EARTH algorithm produced the most accurate model with values $ERR_{\text{AVG}} = 5,43 * 10^{-4}$ and $ERR_{\text{MAX}} = 9,56 * 10^{-4}$.

With the parameter optimization, the execution time was also of interest. According to Table 6, the algorithms had very different Time values. KRR was the quickest to perform in 0,34 seconds, RFR the second in 88,98 seconds and Earth algorithm took 278,93 seconds to execute. While the Earth was the slowest to train, by large margin, it did produce the most accurate model for the TSFC dataset and thus would be the optimum choice of machine learning algorithm for the dataset.

Finally, to validate the results of the parameter optimization, a plot was constructed to visually represent the models for all three algorithms. Actually, the KRR algorithm did not produce any usable figures with the RBF kernel. The two others were adequately visualized, confirming the usability of the models. The plots are presented in Figure 4 for EARTH model and Figure 5 for RFR model.

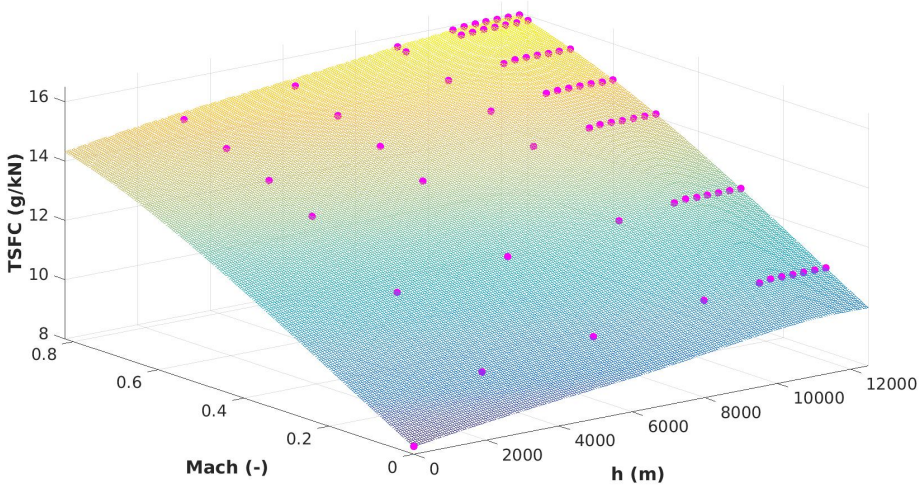


Figure 4: EARTH algorithm model with the optimized parameters

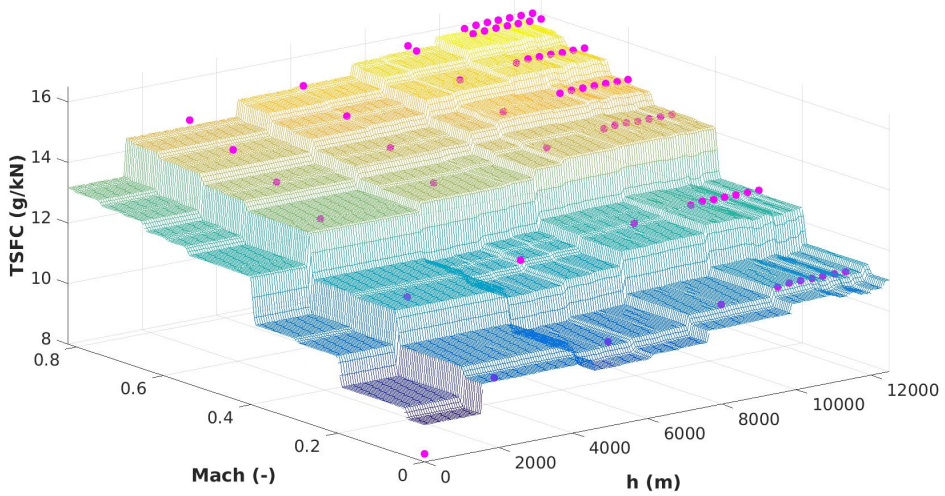


Figure 5: RFR algorithm model with the optimized parameters

5.2.2 Optimization for CFF Dataset

CFF consists of 937 data points with 4 independent variables. Therefore, the separation of the dataset should not induce any problems regarding the random subsets. The testing set includes on average 187 data points that intuitively would represent a descent sampling of the entire dataset. The results of the parameter optimization for CFF dataset are compiled in Table 8.

RFR algorithm optimization did not alter the parameters much, the only change being the $NumPredictorsToSample$ optimum of 3 compared to the 2 of initial analysis. However, the separated dataset revealed that the bias in initial analysis was a major factor and the result from the optimization reduced the performance. The average error changed from 0,00614 to 0,0116, an increase of almost 100 %, and the maximum error was approximately 50 % higher compared to the initial analysis.

The results of initial analysis for the KRR was very accurate, an order of magnitude of better than EARTH or RFR. After parameter optimization, the sigma increased to 100 and Regulation term decreased greatly to 10^{-8} from $5 * 10^{-3}$. As expected, the result from initial analysis was too optimistic and the performance of KRR decreased, although KRR still produced a descent result $ERR_{\text{AVG}} = 0,0040$, comparable to the result from EARTH algorithm.

The EARTH algorithm produced the least accurate model in the initial analysis. The altered parameters from the optimization include the Thresh becoming 10^{-5}, Max_terms reducing to 25 and Max_degree increasing from 3 to 4. The performance improved with the optimized parameters so that average error reduced from 0,00793 to 0,00386 and maximum error is now 0,01081. The results indicate that the EARTH model is the only model that improved in performance with the parameter optimization and is the most accurate with the CFF dataset.

Similarly as with the TSFC dataset, the execution time was recorded. The order of the execution times is the same as with TSFC dataset, but the KRR Time increased substantially to 32,39 seconds and RFR decreased to 57,60 seconds, while the EARTH algorithm took 250,82 seconds to complete. Again, although the EARTH was clearly the slowest to analyze, it produced the most accurate model and thus being the optimum choice for CFF dataset.

The results of the CFF parameter optimization required same validation as TSFC dataset. However, the CFF dataset includes 4 independent variables, meaning that a plot of the data must be divided to multiple plots. We have created 3 plots per algorithm, keeping weight and ΔT_{ISA} constant for the plots and including the data points with only the specific weight and ΔT_{ISA} combination. The plots are similar to the ones in Figure 4 and Figure 5 and are presented in Appendix 1.

For CFF dataset, a plot for the KRR model was constructed but, from the plot, it is obvious that the model is not usable to accurately represent the dataset. The problem with KRR model is depicted in Figure 6. While the model does provide accurate prediction of the training and testing data points, prediction on new observations further away from the data points reduce towards value 0. Looking at the data used to plot the TSFC model, same effect is present there only more emphasized and thus not producing a viable plot. Later, in Section 6, we examine the KRR algorithm with another kernel to model the datasets.

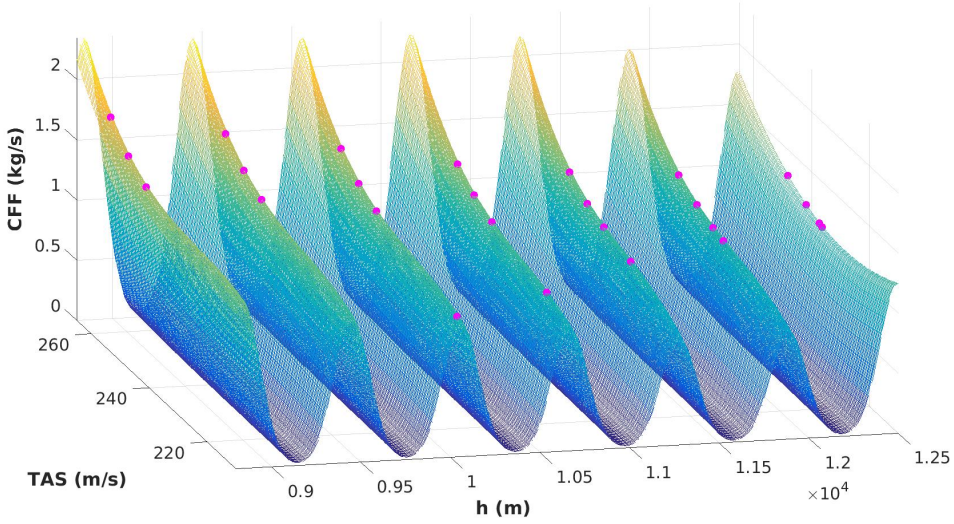


Figure 6: Validation plot of KRR algorithm showing the problem with the model.

6 Discussion

This chapter discusses the various characteristics of the thesis. Firstly, we discuss the datasets used. Secondly, we discuss the machine learning algorithms utilized to construct the models. And thirdly, we discuss the results of the analyses, including the evaluation of the results.

6.1 Datasets

The datasets in this thesis were obtained from the AFM's for the specific aircraft models. Thus, the dataset was complete, something not always expected in data analysis, and implied that the data should be rather noiseless, reducing the effect of overfitting. In theory, there would be an exact formula with which to model the dependencies between the input and output variable. However, even if known, the formula is not described in any accessible publication, therefore creating the need for a regression analysis model. Regression type machine learning algorithms were chosen for the modelling from the requirement of accurate models.

The two datasets used, TSFC and CFF, were chosen for having different characteristics. TSFC is a considerably smaller dataset with 69 data points and only two features, while the CFF is a larger dataset, having 937 data points and four features, as described in Subsection 2.2. Having two datasets with varied sizes gives insight how the algorithms perform on different datasets. Granted, the optimal algorithm might be different for similarly sized datasets that have different dependencies between the variables.

6.2 Machine Learning Algorithms

This thesis analyzed the performance of three machine learning algorithms. As already mentioned in Subsection 2.3, the algorithms were chosen to model regression problems, to be adequately different to obtain comparable results and to find the dependencies between variables for accurate models. The decision to include the algorithms, EARTH, RFR and KRR, was based on the on the fact that all of them were well established methods and usually resulted in quite accurate models. However, due to the nature of machine learning problems, a comprehensive study to evaluate the general performance of different algorithms is difficult to conduct and the evaluations are always case-specific.

From interpretability standpoint, the EARTH and RFR models are quite easy to asses and their implementation to the Falconet software would be rather straightforward. Moreover, the RFR model is a decision tree, meaning that only discrete values are produced in prediction. The stepwise behavior of output value impacts the prediction accuracy, especially outside of the data points as seen from Figure 5. With Earth and KRR models this should be not an issue as the models are constructed from BF's that may have continuous values on the entire dimension defined by the dataset.

In the end, however, as stated in Subsubsection 5.2.1, KRR did not produce usable models with the parameters provided. Figure 6 in Subsubsection 5.2.2 presents the problem with KRR algorithm with the parameters used in modelling. The "ridging" in the plot was suspected to result from the choice of kernel, thus, a quick analysis was conducted on another kernel to verify the assumption. This is presented in Subsection 6.4

6.3 Results and Evaluation

The initial analysis of the machine learning algorithms provided good results. Based on those results the KRR would have been the best choice for both datasets. For the smaller dataset, the EARTH was almost as good model, but on the larger dataset its performance decreased greatly. On contrary, RFR was clearly the inferior choice on TSFC dataset and witnessed improved performance on the CFF dataset. This is probably a consequence of difference in the number of features. The TSFC dataset has only two features and thus the RFR algorithm has fewer alternatives to split the data.

Additionally, during the analysis for TSFC, the algorithm for RFR used in the Matlab package includes an option to set the $NumPredictorsToSample$ parameter to equal the independent variables in the dataset. Doing this means that the algorithm does not represent Breiman's RFR as described in his work [11]. It was included in the parameter optimization and the optimal parameter combination does have $NumPredictorsToSample = 2$, as evident from Table 6.

As the intention was to use proper Breiman's RFR, we analyzed the RFR with restriction of maximum value of 1 for the NumPredictorsToSample. According to the optimization, the Trees reduced to 25 and performance reduced slightly to $ERR_{AVG} = 0,0058$, while maximum error improved, reducing to $ERR_{MAX} = 0,0139$. Validation plot for the restricted RFR analysis is shown in Figure 7. The improvement in maximum error is evident in the front corner of the plot and reduced average error performance from

the data points being further under the plotted plane.

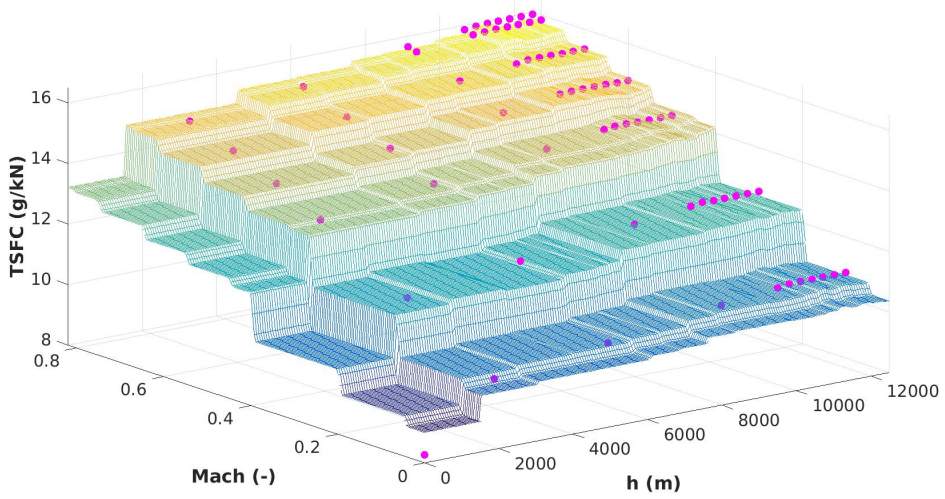


Figure 7: Validation plot of RFR algorithm with restriction to $NumPredictorsToSample = 1$.

Looking at both plots for TSFC modelled with RFR, the general appearance depicts the limitation of a decision tree model. The predicted values obtained from the model change in intervals that are occasionally rather steep. This might increase the true average error if additional "real" data would be used to validate the model. In this instance, the state of RFR makes little difference since the EARTH algorithm outperformed RFR, but the smaller average error outweighs the status of proper RFR. However, in future analyses, it might be beneficial to use a third subset of the data for validation and choose the optimized model with performance based on the validation set. In this thesis, the CFF dataset might have been large enough for it, but TSFC dataset definetly was not.

6.4 KRR Algorithm with Polynomial Kernel

After founding the RBF kernel to be unsuitable to model the datasets, an alternative kernel was analyzed to determine whether the algorithm itself was suitable to model the datasets. The kernel chosen for the alternative analysis was polynomial kernel, or "POLY" for the Matlab application. Polynomial

Table 9: The grids for the search of optimal tuning parameters and number of combinations for KRR with POLY kernel

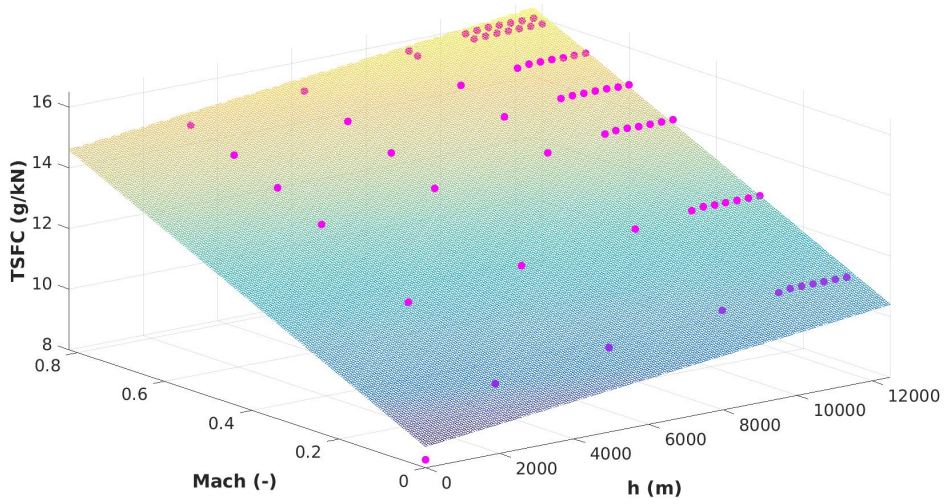
Dataset	**Parameter**	**Grid values**
TSFC	Degree	[1, 2, 3, 4, 5]
TSFC	Bias	[10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}]
TSFC	Regulation term	[10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]
TSFC combinations 150		
CFF	Degree	[1, 2, 3, 4, 5]
CFF	Bias	[10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}]
CFF	Regulation term	[10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}]
CFF combinations 150		

kernel replaces the Equation (18) with

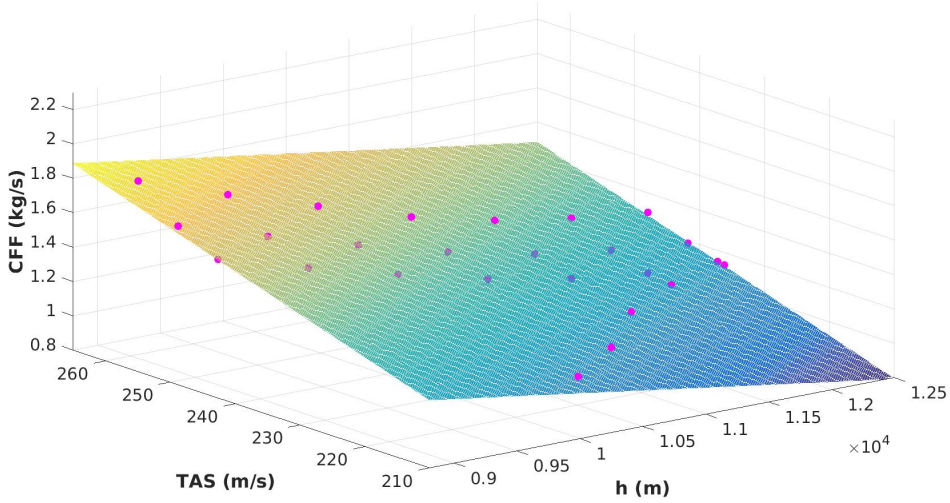
$$\kappa\left(X, X'\right) = \left(X'X + b\right)^{d}, \tag{23}$$

where b is the bias and d is the degree of the model. Otherwise the description of the KRR algorithm in Subsection 3.3 is valid, combining Equation (17) with Equation (23) instead.

In addition to the parameters b and d, the KRR with POLY kernel also requires the Regulation term as input parameter. The parameter grid for the analysis is presented in Table 9 and results in Table 10. The optimum parameter combination for both datasets includes polynomial degree of 1 meaning that the modelled plot is a plane defined by straight lines. Unsurprisingly, the performance is decreased compared to the RBF analysis, however, the KRR algorithm seems to still be able to model the dataset. The validation plots for polynomial KRR analysis are presented in Figure 8.



(a) Validation plot for TSFC



(b) Validation plot for CFF with $\Delta T_{ISA} = 0$ and $m = 140000$kg

Figure 8: Validation plot of KRR algorithm with polynomial kernel

Table 10: Final results of the parameter optimization for KRR with POLY kernel

	TSFC	**CFF**
Degree	1	1
Bias	0,1	100
Regulation term	10^{-4}	10^{-1}
$\mathbf{ERR_{AVG}}$	0,0105	0,0211
$\mathbf{ERR_{MAX}}$	0,0219	0,1444
Time	1,75 s	101,93, s

7 Conclusions

This thesis describes the study conducted to obtain an optimal machine learning algorithm to model aircraft performance parameters. The analyzed datasets were TSFC for Boeing B737-700 and CFF for Airbus A330-300.

Three machine learning algorithms were chosen as potential candidates for optimal performance in predicting the interaction of independent variables have on the dependent variable. The function of the three algorithms, EARTH, RFR and KRR, were explained and subsequently an analysis was conducted to obtain initial information of the tuning parameters. Next, chosen parameters were included in a optimization including data separation to different training and testing sets.

According to the parameter optimization analysis in Subsection 5.2, the EARTH algorithm appears to be the optimum choice for both TSFC and CFF datasets. During the thesis work, the RBF kernel seemed to model the dataset well, however, the validation plot revealed that good accuracy did not persist outside the range of observed values. Additionally, while the RFR algorithm produced decent models, the optimum parameters, however, did not portray Breiman's RFR which was the intention.

Regarding computational time required by the algorithms, EARTH was clearly the slowest to train and evaluate, although that was in turn countered by superior performance. Thus, I would recommend using the EARTH algorithm to model the aircraft parameters used in the thesis. Furthermore, given that EARTH was the optimum algorithm for two completely different datasets, it would be a great starting algorithm to apply to new datasets, even if different in characteristics. Considering new datasets, suggested approach would include the following steps:

- Separate the data to training set and test set (and validation set)
- Choose an algorithm according to the problem to solve
- Define the parameter grid, choosing a large spread of values
- Train the model and, according to predetermined metric(s), evaluate the performance
- If the optimum parameters are in the extreme values of the grid, redefine the grid and train again
- Use validation dataset or plots to verify suitability

- Repeat the process for at least one other algorithm suitable for the problem

The steps above consist roughly the procedure utilized in the thesis, excluding the use of validation dataset.

References

[1] E. Alpaydin, *Introduction to Machine Learning*. Adaptive computation and machine learning, MIT Press, 2014.

[2] IATA, "Fact sheet - fuel," 2016. https://www.iata.org/pressroom/facts_figures/fact_sheets/Documents/fact-sheet-fuel.pdf. Accessed 23.1.2017.

[3] EASA, "Certification specifications for large aeroplanes cs-25," 2003. https://www.easa.europa.eu/system/files/dfu/decision_ED_2003_02_RM.pdf. Accessed 8.5.2017.

[4] D. Raymer, A. I. of Aeronautics, and Astronautics, *Aircraft design: a conceptual approach*. Educ Series, American Institute of Aeronautics and Astronautics, 1989.

[5] Aalto University School of Engineering, "Lentokoneen suoritusarvot [class handout]."

[6] E. Torenbeek, *Synthesis of Subsonic Airplane Design*. Springer Netherlands, 2013.

[7] "Part 25 - airworthiness standards: Transport category airplanes." 14 C.F.R. § 25.1581, 1990. https://www.ecfr.gov/cgi-bin/text-idx?SID=d193685f4585778f481bbf1428ba0fec&mc=true&node=pt14.1.25&rgn=div5. Accessed 8.5.2017.

[8] B. Marr, "A short history of machine learning – every manager should read," 2016. https://www.forbes.com/sites/bernardmarr/2016/02/19/a-short-history-of-machine-learning-every-manager-should-read/#5c7d920615e7. Accessed 17.5.2017.

[9] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[10] J. H. Friedman, "Multivariate adaptive regression splines," *The Annals of Statistics*, vol. 19, no. 1, pp. 1–67, 1991.

[11] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[12] L. Breiman and A. Cutler, "Random forests." https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm. Accessed 21.5.2017.

[13] V. Vovk, *Kernel Ridge Regression*, pp. 105–116. Springer Berlin Heidelberg, 2013.

[14] K. Murphy, *Adaptive Computation and Machine Learning*. The MIT Press, 2012.

[15] "MATLAB and statistics toolbox release 2016a." The MathWorks Inc., Natick, Massachusetts, United States.

[16] J. Santarcangelo, "Kernel ridge regression in Matlab," 2015. https://se.mathworks.com/matlabcentral/fileexchange/49989-kernel-ridge-regression-in-matlab. Accessed 15.9.2016.

[17] J. Rudy, "py-earth," 2013. https://github.com/scikit-learn-contrib/py-earth. Accessed 1.9.2016.

[18] K.-M. Osei-Bryson and O. Ngwenyama, *Advances in Research Methods for Information Systems Research*. Springer, 2013.

[19] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, Springer New York, 2013.

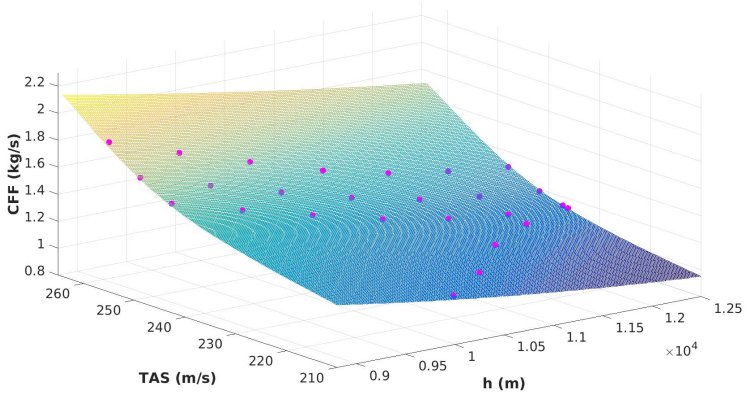
[20] A. E. Hoerl and R. W. Kennard, "Ridge regression: Biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[21] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?," in *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, pp. 154–168, Springer, 2012.

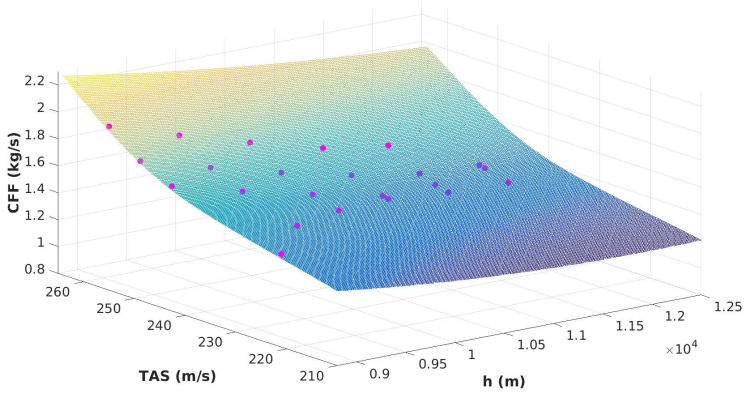
Appendix 1. CFF Dataset Validation Plots

This appendix incorporates the validation plots for the CFF dataset. The CFF dataset includes 4 independent variables. For 3D plots, 2 of the variables are required to be set constant and the rest 2 with the dependent variable are used to plot the model and compare the observation from the dataset to validate the model. The ΔT_{ISA} and mass variables are chosen to be the constant values due to having less distinctive values in the dataset.

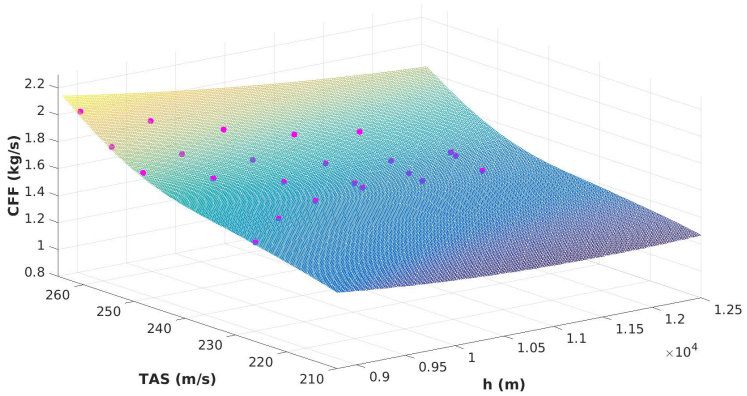
In the first plot, Figure 9 (a) and Figure 10 (a), we have set ΔT_{ISA} to 0°C and mass to 140000 kg. The second plot in Figure 9 (b) and Figure 10 (b) the parameters are set to 0°C and 180000 kg for ΔT_{ISA} and mass, respectively. Finally, the ΔT_{ISA} is set to 10°C and mass to 180000 kg in the third plot in Figure 9 (c) and Figure 10 (c). The data points in the figures include only data points that have the corresponding ΔT_{ISA} and mass.



(a) $\Delta T_{ISA} = 0$ °C, m = 140 000 kg

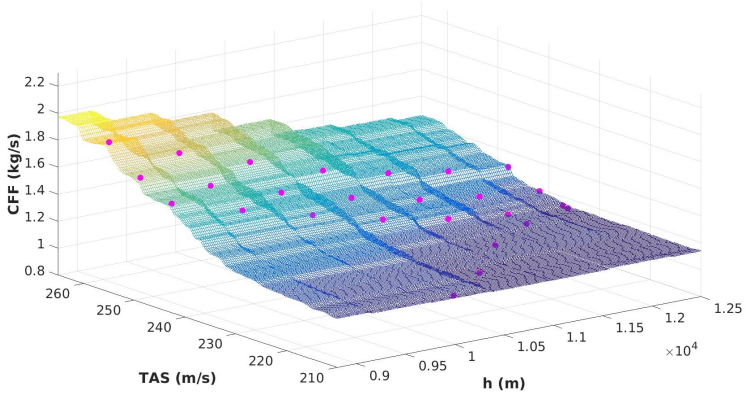


(b) $\Delta T_{ISA} = 0$ °C, m = 180 000 kg

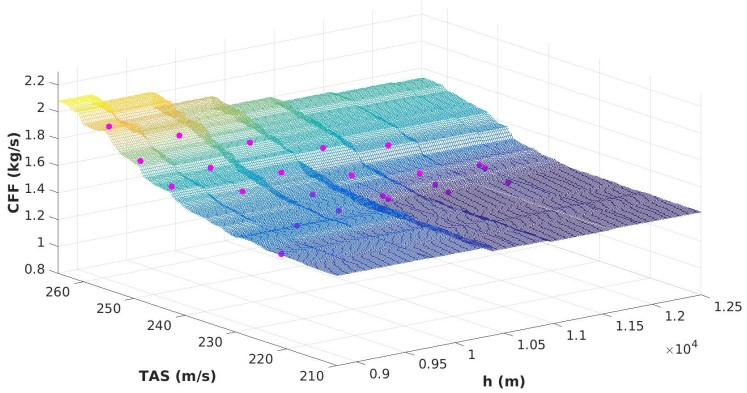


(c) $\Delta T_{ISA} = 10$ °C, m = 180 000 kg

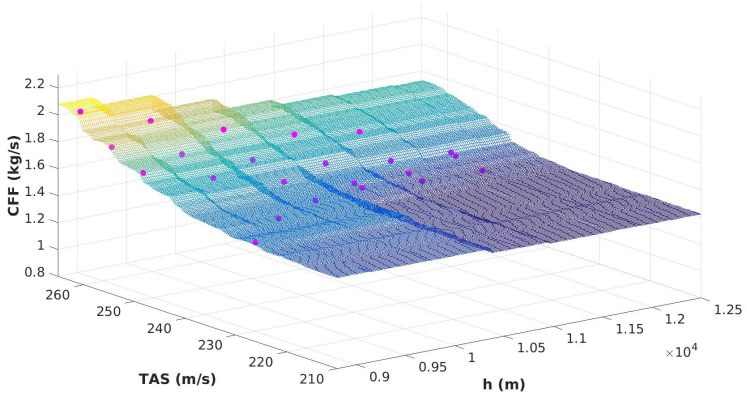
Figure 9: Validation plots for the optimum Earth algorithm with three parameter combinations



(a) $\Delta T_{ISA} = 0$ °C, m = 140 000 kg



(b) $\Delta T_{ISA} = 0$ °C, m = 180 000 kg



(c) $\Delta T_{ISA} = 10$ °C, m = 180 000 kg

Figure 10: Validation plots for the optimum RFR algorithm with three parameter combinations