Linköping University | Department of Electrical Engineering
Master’s thesis, 30 ECTS | Diagnostics
22 | LITH-ISY-EX--22/5528--SE

Adaptive detection of anomalies in fuel system of Saab 39 Gripen using machine learning

– Investigating methods to improve anomaly detection of selected signals in the fuel system of Gripen E.

Modellbaserad adaptiv anomalidetektion i Saab 39 Gripens bränslesystem

Olof Ahlgren Bergström

Supervisor : Olov Holmer
Examiner : Erik Frisk

External supervisor : Ylva Nilsson and Olof Bengtsson

Linköpings universitet
SE–581 83 Linköping
+46 13 28 10 00 , www.liu.se

Abstract

The process of flying fighter jets naturally comes with tough environments and manoeuvres where temperatures, pressures and forces all have a large impact on the aircraft. Part degeneration and general wear and tear greatly affects functionalities of the aircraft, and it is of importance to carefully monitor the well being of an aircraft in order to avoid catastrophic accidents. Therefore, this project aims to investigate various ways to improve anomaly detection of selected signals in the Gripen E fuel system. The methodology in this project was to compare collected flight data with generated data of a simulation model. The method was conducted for three selected signals with different properties, namely the transfer pump outlet pressure and flow, as well as the fuel mass in tank 2. A neural network was trained to generate predictions of the residual between measured and simulated flight data, together with a RandomForestRegressor to create a confidence interval of said signal. This made it possible to detect signal abnormalities when the gathered flight data heavily deviated from the generated machine learning algorithm predictions, thus alarming for anomalies.

Investigated methods to improve anomaly detection includes feature selection, adding artificial signals to facilitate machine learning algorithm training and filtering. A large part was also to see how an improved simulation model, and thus more accurate simulation data would affect the anomaly detection. A lot of effort was put into improving the simulation model, and investigating this area. In addition to this, the data balancing and features to balance the data on was revised. A significant challenge to tackle in this project was to map the modelling difficulties due to differences in signal properties. A by-product of improving the anomaly detection was that a general method was obtained to create a anomaly detection model of an arbitrarily chosen signal in the fuel system, regardless of the signal properties.

Results show that the anomaly detection model was improved, with the main improvement area shown to be the choice of features. Improving the simulation model did not improve the anomaly detection in the transfer pump outlet pressure and flow, but it did however slightly facilitate anomaly detection of the fuel mass in tank 2 signal. It is also concluded that the signal properties can greatly affect the anomaly detection models, as accumulated effects in a signal can complicate anomaly detection. Remaining improvement areas such as filtering and addition of artificial signals can be helpful but needs to be looked into for each signal. It was also concluded that a stochastic behaviour was seen in the data balancing process, that could skew results if not handled properly. Over all the three selected signals, only one flight was misclassified as an anomaly, which can be seen as great results.

Acknowledgments

I would like to thank Saab, and more specifically the TDGT section at Saab for giving me the opportunity to explore the advanced technology of Gripen E. A huge thanks to my supervisors and Saab engineers Ylva Nilsson and Olof Bengtsson for all the time and support you have given me during this project. The project surely reached higher levels with your expertise and knowledge. I also want to thank Dan Louthander for his very helpful *git* expertise and always being willing to help. In addition to this, I will forever be grateful for the very interesting *fika* discussions and good company everyone at the TDGT section at Saab gave me.

I would also like to thank my project inspector and professor Erik Frisk for the support during this project, together with supervisor Olov Holmér for his assistance and input. Thank you!

Contents

1 Introduction

1.1 Background and Purpose

This section presents the background and purpose of the project, revolving around anomaly detection and improvements in the fuel system model of Saab 39 Gripen. Here, information is presented motivating why the project work is relevant and interesting to study. Additionally, the work done is put into a more general context as well as giving the reader a more extensive background.

Purpose

Every year thousands of flight hours are performed with the Gripen aircraft. There is a constant need to keep the aircraft in good health to protect the countries, or at Saab to further develop aircraft technology. When flying, the aircraft is exposed to tough environments, great forces, and a variety of temperatures. This creates a great interest to gain insight into an aircraft's "well-being" in order to prevent system failures. Information regarding component wear and tear as well as recognizing component degradation is crucial, making it easier to plan repairs and service. This type of information also facilitates mechanical diagnosis and can be of great use for troubleshooting and recognizing system deviations.

It is easy to motivate why there is a need to further develop the ability to detect anomalies in flight data so that faulty parts can be replaced in time and suspicious aircraft behavior can be investigated. The main goal of improving the ability to accurately detect anomalies in flight data can not only help reveal alarming data patterns but also point out interesting information that facilitates further fuel system model development. In addition to this, a by-product of efforts put into anomaly detection investigation is an improved simulation model, that can be useful in other aspects. The knowledge gained together with simulation model improvements can then be of great use at Saab for further flight data analysis. It can also be used when developing and redesigning current fuel system models in order to evaluate functionality as the technology moves forward. But the main argument these factors motivate is the need for anomaly detection, and improving the simulation model is one area of investigation that can result in a desirable by-product. These factors motivate the goal of the thesis project: to investigate the fuel system model in order to achieve improved anomaly detection.

Background

It is no doubt that it is beneficial to accurately detect anomalies in flight data signals of Gripen. However, model-based anomaly detection comes with difficulties as the degrees of freedom in an aircraft are vast. The fact that every aircraft is hand-built, its uniqueness in parts, and thousands of flying settings, environments, and properties are all problems that need to be handled, to mention a few. Degradation of items and general wear and tear also alters plane characteristics. In addition to this, the problem of deciding which types of data patterns should be classified as anomalies poses another question to be answered. In some situations deviating data might be expected, meanwhile, the same data patterns in other situations could be considered alarming. This shows the need for a dynamic anomaly detection threshold and motivates the idea to let machine learning algorithms decide what is normal in an aircraft and not. Thus, algorithms can point out abnormal patterns in the data which would otherwise be a cumbersome process.

This thesis presents a method where a simulation model implemented in Dymola simulates signals that are also measured in Gripen aircraft. By looking at the residual between these signals, it is possible to measure the deviation between model output and measured data. In regards to deciding what is abnormal and not, machine learning algorithms are trained to predict the residual. Being trained to find the general residual behavior allows for alarming patterns in the measured data to be detected. However, the area of improving anomaly detection is vast and there are a lot of different ways to tackle the problem. One suspected area thought to facilitate anomaly detection is an improved simulation model. In particular, to know whether there is something to gain in regards to anomaly detection by shrinking the residual as a result of a more accurate simulation signal is of interest. Being able to accurately simulate and predict flight data from a complete fuel system model can be of great use in many aspects, and not only for anomaly detection. For example, an accurate simulation model reduces the need to constantly gather new flight data. To obtain a good fuel system model knowledge is needed about which areas need improvement and what type of structure and signal behaviors make a good simulation model. Information such as how to properly model different sub-parts of the fuel mass model and which signals are important to accurately model will be of great interest. It is desirable to obtain more information about the fuel system model, and to what extent the interesting signals needs to be improved. This also creates an urge to know the trade-off in what type of normal data patterns can be covered by ML algorithms and where the fuel system model needs to be improved. Additional areas that facilitate the machine learning algorithm accuracy and thus anomaly detection such as data processing, different types of filtering, and data balancing are also of interest. The benefits of using model-based anomaly detection are something that became the foundation of a previous master thesis [20]. This was done by investigating several different ML-methods on $r(t)$ to predict $\hat{r}(t)$ as well as a confidence interval for said signal, $I(t)$. However, the ML model used for now was implemented at a later stage. The methods used for anomaly detection of a signal in the fuel system model were created in the previous master thesis project.

For this type of analysis setup, an anomaly can be defined as when measured flight data of the chosen signal deviate from the prediction interval created with ML algorithms during an extended period. A more precise description is when $r(t)$ is not within the prediction interval of the trained ML method, $I(t)$ during a longer period. A lot of the work done in the previous master thesis project contains signal pre-processing, feature selection, and evaluation of the ML methods. It also investigated what kind of model deviations the ML methods can cover. The work to be done in this project will take a new direction, focusing more on gaining crucial information about the simulation model/ML algorithm anomaly detection trade-off, and also investigating new signals with new properties.

1.2 Uniqueness of this master thesis

A previous master thesis work done by *C. Tysk and J. Sundell* [20] developed an anomaly detection method by analyzing the residual between simulated and measured data of the fuel mass in a tank by using ML methods. This process included data cleansing and investigation of different ML algorithms to see which algorithms produced the most accurate predictions. A look into which features to select when training ML algorithms and a data balancing process to achieve a more evenly distributed dataset were also done. It was later shown that the data balancing procedure contained faulty code which skewed the balancing, and this issue was later resolved by Saab. In addition to this, a method to generate a prediction interval $I(t)$ making it possible to detect anomalies was also developed. Thus, an anomaly detection alarm was implemented to detect large deviations in measured flight data. Additional areas were also looked into such as leakages in the target signal and offsets. The thesis resulted in an MDM (which in this thesis is renamed to RM), *Model Deviation Model* developed in *Python* in *Jupyter notebooks*, where methods to perform said procedures were implemented.

This master thesis aims to improve anomaly detection of aircraft data, and a central part of the procedure of doing so is improving the Dymola simulation model by analyzing the behavior of three chosen signals with different properties. Another central thesis goal is to find a suitable level in the trade-off between improving the DM simulating data of the given signal and ML algorithm performance. By comparing simulated and measured data, efforts will be made to see how the DM can be improved, and the consequences of improving the DM in ML algorithm prediction accuracy are of great interest. The previous master thesis did not include any work in the DM, which extends this thesis into previously unexplored subjects.

Although some work by previous master thesis writers was put into similar areas that are looked into in this thesis, such as feature selection and data balancing, the main focus of this thesis is to improve anomaly detection. The areas of feature selection and data balancing will be further investigated in this thesis, but with an aim to gain necessary information about the fuel system model and what ML algorithms can cover.

1.3 List of Terms

A list of commonly occurring terms describing the different models and signals that are central to the thesis project is shown below. To facilitate discussion and presentation of the results, the terms listed will be used to simplify understanding and text.

List of Signals

Below, signals associated with the system model can be seen. Signals include measured signals from flight data of the aircraft, simulated signals from the Dymola fuel system model, and signals created from combining mentioned signals in combination with ML algorithm predictions. The list presents notations for an arbitrarily chosen signal of the fuel system, which then will be denoted with the actual signal name. The signals that will be investigated in this project mainly consist of three target signals with different properties. The following signals will be investigated:

- Transfer Pump Outlet Pressure denoted pr.
- Transfer Pump Outlet Flow denoted fl.
- Fuel Mass in Tank 2, denoted fm.

To differentiate each of the target signals from each other, and represent measured and simulated signals the following signal notations are introduced.

- $y_{meas}(t)$ - An arbitrarily chosen signal from measured flight data in a Gripen aircraft, for example, the fuel mass or the transfer pump outlet pressure. The signal of choice will be denoted, e.g. $y_{meas,PR}(t)$ for the measured transfer pump outlet pressure.
- $y_{sim}(t)$ - The Dymola model simulated output signal, given a set of input signals. In similar fashion as $y_{meas}(t)$, the signal of choice will be denoted, e.g. $y_{sim,FL}(t)$ for the measured transfer pump outlet flow.
- $r(t)$ - The residual between $y_{sim}(t)$ and $y_{meas}(t)$.
- $\hat{r}(t)$ - The predicted output signal from an ML-model that has been trained to predict $r(t)$ given a set of input signals.
- $I(t)$ - Prediction interval for $\hat{r}(t)$. Calculated by training a RandomForestRegressor ML-method to predict $(\hat{r}(t) - r(t))^2$. The RandomForestRegressor predictions are then used together with $\hat{r}_{RM}(t)$ to create a model prediction interval for $r(t)$. If $r(t)$ deviates from $I(t)$ for an extended period of time the data of the chosen signal will be classified to contain an anomaly.

List of Models and Terms

There are two different model areas that are essential in this project. Firstly, an extensive fuel system model implemented in Dymola contains models to produce different simulation signals, listed as $y_{sim}(t)$ above. Secondly, the models used to analyze $r(t)$ are implemented in Jupyter notebooks and Pycharm containing ML methods, data pre-processing code, and implementation of the anomaly detection algorithms.

- ML - Short for machine learning.
- Features - Which signals in the data set that is included when training ML algorithms.
- Target Feature/Signal - The signal of which the ML algorithm is trained to predict.
- DM - Dymola Model: A system model for the fuel system as a whole, which is implemented in Dymola. Here it is used with a set of input data taken from sensors and control signals of Gripen to produce the signal $y_{sim}(t)$. The notations *pr*, *fm* and *fl* will be used to specify a certain submodel of the DM, for example DM_{fm} for the submodel simulating data of the fuel mass.
- FD - Flight Data: Registered signals in the aircraft during flight representing flight information such as pressures, velocities, temperatures, etc. This will be used to train ML models, and as true data to evaluate simulated data with.
- RM - Residual Model: A model used to handle the signals in *List of Signals* to generate predictions of the target signal. There is a complete method containing procedures that result in an RM, that loads and pre-processes $y_{meas}(t)$ and $y_{sim}(t)$ to generate the signal $r(t)$. The model consists of ML algorithms to generate predictions of $r(t)$, namely $\hat{r}(t)$. Additionally, the model also contains ML algorithms to create the prediction interval $I(t)$, by generating predictions of $(\hat{r}(t) - r(t))^2$. Included in the method to create the RM are also procedures to handle filtering, anomaly detection, and data balancing of the signal looked at. A separate RM is developed for each of the target signals to handle different signal properties, thus creating an RM_{fm} for the fuel mass, RM_{fl} and RM_{pr} for the transfer pump outlet flow and pressure respectively. The methods to develop each RM mostly contain similar procedures, with a few exceptions to tackle the unique behaviors that come with each signal.

1.4 Problem Formulation and Goals

To obtain a clear vision of what is to be achieved during this project, a thorough description of the goals is required. Additionally, to reach project goals a good understanding of the problems and challenges of the subject is crucial. The project problems and goals are thoroughly presented below.

Description of problems

The main problem of this master thesis is the question of how to acquire a better anomaly detection model. To gain the necessary knowledge needed to improve anomaly detection, a few central problem areas are stated. Initially, it is of great interest to see how an improved DM, generating a more accurate simulation signal, affects the ML algorithm performance, and thus anomaly detection. A central question is how different signal properties affect anomaly detection, such as the difference in model performance between accumulated signals such as the fuel mass, and non-accumulated signals such as the transfer pump outlet pressure. In addition to this, a problem lies in enhancing the conditions for the ML algorithms to acquire optimal RM performance. This includes all types of procedures done to the training data set and processes done before training the ML algorithms. For example, the importance of data balancing, filtering, and feature selection are areas that can pose problems if not looked into properly. An important problem to look into is the balance between improving the DM and pinpointing the model flaws that can be accurately covered by ML algorithms. In some cases, it might be sufficient to improve the model to a certain extent and to let the ML method cover any deviations. The main problems of the thesis are stated in the following list.

- To see if improved anomaly detection can be gained by improving the DM generating $y_{sim}(t)$, thus shrinking the residual $r(t)$. This is done by identifying systematic modeling faults in the DM, analyzing why these faults occur, and then locating which parts of the DM need to be improved.
- Developing RMs for the transfer pump outlet pressure and outlet flow. Thus, solving the problem of creating a good testing environment in Jupyter notebooks, where large-scale model and signal evaluation is easy.
- To map the different signal properties in the fuel mass, outlet pressure, and outlet flow and how these properties affect model performance.
- Solving the problem of choosing the right feature for each signal, and what kind of data pre-processing that needs to be done in each RM. Which procedures are of greater importance to the ML algorithm performance, and which processes can be neglected?
- Finding a good balance between improving the DM and RM. How much impact does each model area have on $\hat{r}(t)$ and the prediction interval $I(t)$, and where lies the trade-off between improving these two models?

Goals

Ideally, the main goal of this master thesis is to acquire significantly improved anomaly detection models for the three chosen target signals. A by-product of the improved anomaly detection is an improved Dymola simulation model, and what's to gain from an improved simulation model. As this knowledge, for now, is unknown, the goal is rather to explore this area and the improved simulation model can be seen as a by-product when gathering this information. Additionally, a goal is to see how to best facilitate ML algorithm training to handle different signal properties. It is also desired to extract wisdom about what kind of deviations the RMs can accurately cover and not, and also find a good balance between improving the parts of the DM and the RMs. Some deviations might be hard to accurately model in Dymola, but slight DM improvements might be enough for the ML methods to successfully cover those specific situations. Mapping situations and signals involved when anomaly detection struggles are also of great interest, and a large part of the project will aim to find this knowledge. The overall main goal of the project is however improved anomaly detection, and fulfilling the goals listed below are all thought to help achieve the main goal.

- Main goal: To acquire a significantly overall improved anomaly detection, in regards to the signals investigated and their corresponding different properties. A central part of this is to see what is to gain from improving the Dymola simulation model, thus shrinking the residual signal.
- To acquire a good knowledge base of what is a reasonable simulation model level to sufficiently detect anomalies, and to improve the DM in needed areas.

To successfully achieve the main goal the following sub-goals are set.

- To create a structured Python code environment to make data analysis and model evaluation easier. This will also simplify the understanding of how model improvements of the DM should be implemented.
- Finding a reasonable trade-off and extracting information about what kind of model deviation can be accurately covered by the RMs, and where the RMs fails to cover the DM errors.
- To create more accurate simulation models in Dymola of the three target signals, and to gain the necessary information about how these improvements affect anomaly detection.
- To create a numerical measurement to track the accuracy of $\hat{r}(t)$ and $I(t)$ which can be used for the analysis of changes made in the DM and RMs.
- To gain knowledge about how to properly train ML algorithms to handle certain signal properties. This includes goals to find important features, how to balance the data, and general procedures that facilitate ML algorithm training.

1.5 Technical Information

The work done will be coded in *Python* in online browser tool *Jupyter Notebooks*, where code and informative text can be combined. Additional functions, also written in Python from the previous master thesis will be imported from *PyCharm*. The simulation tool used to build the fuel system model is implemented in *Dymola* and contains an extensive model of the Gripen fuel system. To simulate and preprocess data, *PyCharm* and *MATLAB* will be used.

1.6 Delimitations

Military aircraft like Gripen consists of many subsystems that could be of interest to look at when considering anomaly detection in flight data. This thesis is limited to only looking at the fuel system, and more specifically the Dymola simulation model of the fuel system. The fuel system is made up of hundreds of interesting signals that could be used to analyze flight data and detect anomalies, but this thesis focuses only on the transfer pump outlet pressure and flow, together with the fuel mass of tank 2. The properties of these signals differ vastly, making it possible to gain additional insight into challenges with the anomaly detection method. The Dymola model will only be improved to simulate a more accurate version of said signals. In addition to this, work will also be put to facilitate training and predictions of ML algorithms. However, comparing different ML algorithms is out of scope for this thesis, although processes like feature selection are included. Going in-depth into the data balancing process implemented in the previous thesis is out of the scope of this work.

This thesis will only handle data for one aircraft, and one transfer pump. Parts of the aircraft can be changed, which in turn can cause deviations in the general signal level of the chosen signals. To handle this behavior is out of the scope of this thesis.

2 Theory & Related Research

This section presents the relevant theory for the thesis area of investigation. It also presents the necessary information required to follow the anomaly detection method.

2.1 Previous Master Thesis

The most relevant work to this master thesis project is the previous master thesis done by *C. Tysk and J. Sundell* [20] developing the data pre-processing steps, anomaly detection algorithms, and ML models. The project concluded that ML methods can be successfully used to predict deviations between simulated and measured fuel mass in Gripen E and to detect fuel mass anomalies. Various settings and methods were tested, but in regards to anomaly detection, results showed that a linear regression model in combination with a RandomForestRegressor performed best for the fuel mass to accurately predict anomalies. However, this was further improved by Saab at a later stage to instead use a neural network in combination with a RandomForestRegressor, which performed even better. Results also pointed toward the importance of pre-processing the data, including procedures such as feature scaling. In addition to this, the complexity of feature selection was also highlighted. Another important source of information for the previous master thesis as well as this thesis is the book by *A. Géron* [4] which gives a good foundation of how to sort and handle data within the area of Machine Learning as well as valuable information about different models of Machine Learning. The previous master thesis took a lot of inspiration from [4], so it is believed that using the information present there will greatly facilitate the work of this thesis.

2.2 Saab 39 Gripen and the Dataset

Below, a short introduction to the fuel system, transfer pump, and the corresponding signals that are introduced with the said system is presented.

Signals

Every aircraft has a set of probes and measuring equipment that collects information and data during a flight. These measurements include signals such as altitude and measured fuel mass in all the tanks to information about when a specific valve was opened or closed during flight. A big portion of these signals is binary in the fashion that they represent when a valve is turned on or off, or in the case of a control signal that works in a similar manner. Other signals that are numerical are available and can contain information about pressures, velocities, and so on. The other part of the data set is generated from the Dymola simulation model described in Section 2.3 which generates simulated representations of the measured signals.

Fuel System

A simple layout of the tanks in the fuel system can be seen in Figure 2.1. These are the different tanks the fuel system connects, consisting of six different tanks including the right/left wing tank. At the heart of this fuel system sits the transfer pump which is solely used to supply the engine feed with fuel by draining fuel from tanks in a specific order. A simple illustration of the transfer pump can be seen in Figure 2.1. In addition to this, it is also possible to hook up to three drop tanks to the aircraft, one centrally and one for each wing. However, these can not be seen in Figure 2.1.

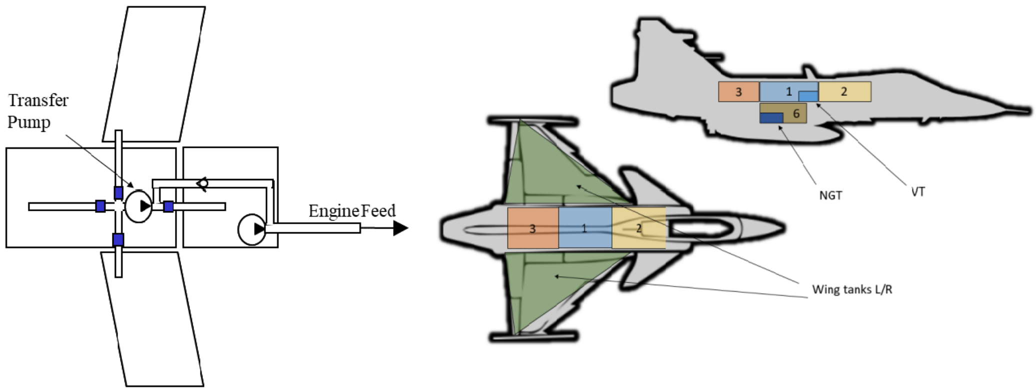

Figure 2.1: An overview of the aircraft fuel system, showing all six tanks and the transfer pump structure.

2.3 Dymola Model & Modelica

Dymola is a modeling and simulation environment that builds on the object-oriented Modelica modeling language, where it is possible to properly model large and complex systems using a component-oriented structure. A general modeling environment is developed to easily model systems including electrical, thermal, and fluid parts, and to connect these systems. Saab started the transition of moving models implemented in other simulation tools to Dymola more than ten years ago, which is discussed more in detail by *I. Lind & H. Andersson* in [11]. In addition to this, research has been done in the area of modeling various aircraft properties and signals in Dymola. *Oehler et. Al* presents in [16] the challenges of modeling thermal and fluid effects in an aircraft fuselage. Three areas of improvement are suggested including mostly user interactive actions. However, the power and potential to easily model complex systems by using Dymola are also highlighted, showing the great aspects of the tool.

The model used in this project is previously developed by Saab and is a simple large-scale model of the whole fuel system in Gripen 39-9 E. It contains sub-models generating simulations of various interesting signals in the aircraft, such as different tank masses, pressures, and flows. The overall model is quite simple and mainly used as something to test software and functionalities against. It is not built on physical formulas, but rather on simple logic to be able to test general behavior. This also explains why the simulation data of the three target signals initially is not very accurate. Simulation is then done by providing Dymola with an input file, representing control and measurement signals that are needed to generate the simulated signals. This input file contains time vectors to define the period of flight, binary values for control signals, and initial conditions as illustrated in Figure 2.2. The input files needed for simulation are mostly created by taking measured data from actual flights. Sub-models are then created by using either mathematical equations or by building manually with blocks using logical gates, integrators, de-limiters, and other available building blocks. There are vast options when building the models manually, which makes large-scale modeling simple.

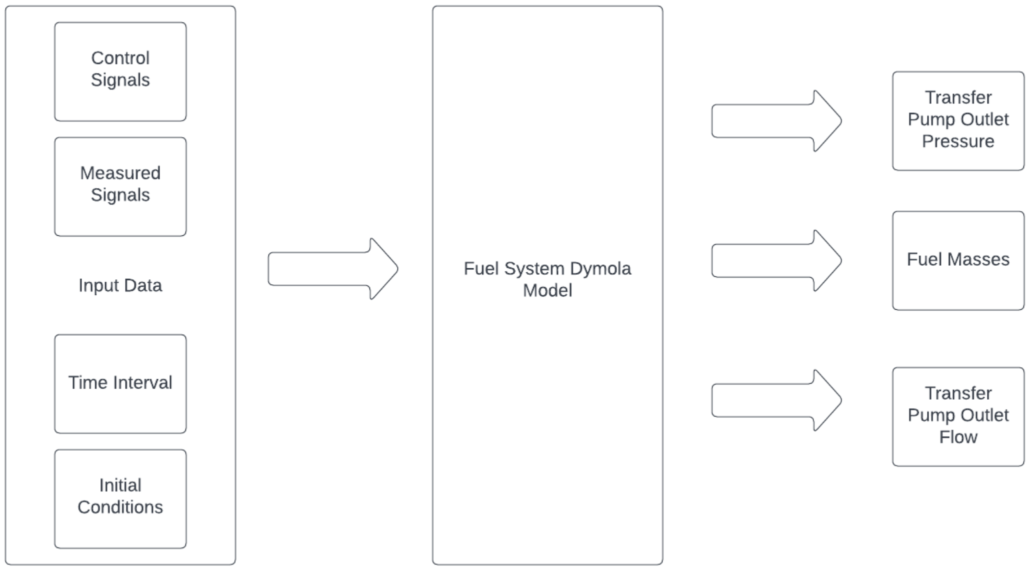

Figure 2.2: Overview of the input/output structure in Dymola.

2.4 Machine Learning Theory

As a central part of this thesis revolves around gaining relevant information about the difference between modeled and registered fuel properties via machine learning algorithms, a brief presentation of the most relevant machine learning algorithms and aspects are conducted. The two main algorithms used are the *neural network* and the *RandomForestRegressor*. Two additional machine learning topics are feature selection and data balancing, which have a great impact on the machine learning algorithm output. These two areas are also put into context in this section. This section presents a brief overview of the algorithms and subjects together with a few sources of related research.

Neural Networks

Neural networks are a powerful group of machine learning algorithms. Inspired by the human brain, a neural network creates a network of *neurons* in divided layers. Each layer can consist of one or more neurons, which are connected to other neuron layers via weights and biases. Generally, three different types of layers are used with the first one being the input layer. Secondly, one or more layers making up the hidden layers are introduced. Lastly comes the output layer which represents the neural network output. Depending on the type of neural network, algorithms can produce output that is either of the classifier types, producing an output that selects one class out of several pre-defined classes. On the opposite, neural networks can also be of the regressor type, used to generate an output value instead of choosing between predefined classes. What is interesting in regards to this thesis is the type of neural network that is used, and how it is used with the data set. Simply put, there are three different types of neural networks. Multi-layer Perceptrons (MLPs) are rather simple networks, using a feed-forward principle, as can be seen in Figure 2.3. MLPs are the most basic neural networks and can be used with great results on simpler tasks. Secondly, the Convolution Neural Networks (CNNs) are a group of neural networks widely used in image and video processing. CNN's are very similar to MLPs but also add a type of filtering in the process that is suitable when handling matrices representing images. Lastly, there is the group of Recurrent Neural Networks (RNNs). These types of neural networks consider the effect of past data, basically storing information about previous time data instances to reuse when making new predictions. Thus, the RNNs are of great use when analyzing time-series data such as the stock market [19], or accumulated signals of a system. A lot of research and work has been done in the area of RNNs, and the ability to counter time-dependent data sets. *M. Canizo* presents results in [2] that show the power of combining an RNN and CNN to tackle the problem of multi-time series anomaly detection with great results.

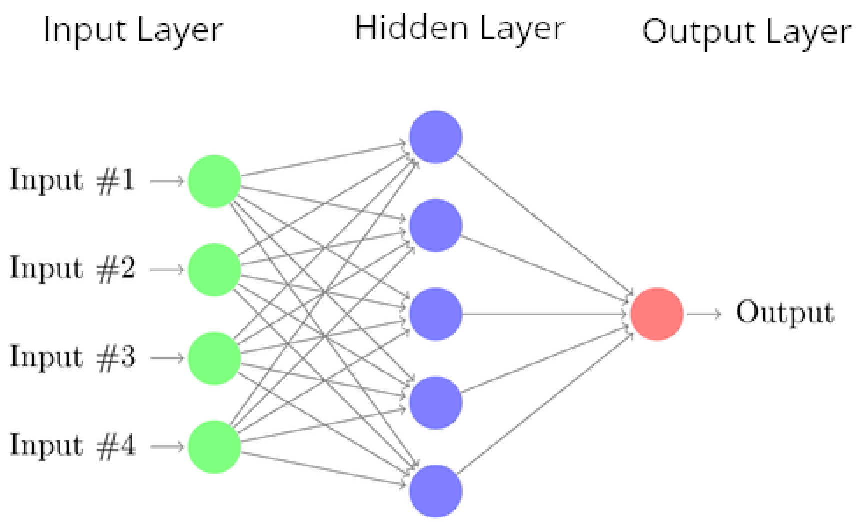

Figure 2.3: A simple overview of the basic structure in a neural network.

The basic math structure behind a neural network can be seen in Figure 2.4. Every neuron of a layer is connected to the neurons of nearby layers, via weights denoted w, as can be seen in Figure 2.4. The output of a neuron is then calculated by taking the sum of all the neurons connected to it, where each neuron's output is multiplied by its corresponding weight. Additionally, a bias b is added to the sum. Lastly, the total sum and bias output of a neuron is taken as input to an activation function, which introduces the ability to handle non-linearity in the network. The introduction of the activation function makes it possible for the network to learn, something that is achieved by pre-training the network. There are several different activation functions, including the *Linear Function*, *Sigmoid Function* and *Tangent Hyperbolic Function* .

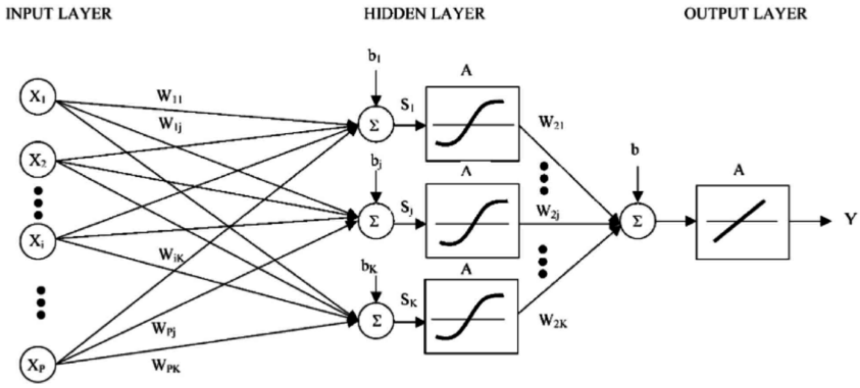

Figure 2.4: An overview of the mathematical structure behind a neural network.

As neural networks belong to the supervised machine learning algorithms family, training of the neural network is needed. To adjust the weights and biases of the network, a method called *backpropagation* is used. Basically, an error term is introduced in the output layer punishing faulty outputs. Errors are then back-traced, adjusting the weights and biases in the neurons for each layer. This is done by reserving a large portion of the input data set to *train* the neural network, basically connecting input signals to the correct output, making the network learn which types of input correspond to the right output. As the performance of neural networks heavily depends on the training phase, it is important to good training set containing a lot of data with great variety.

RandomForestRegressor

The RandomForestRegressor is a machine learning algorithm that utilizes ensemble learning to combine the output of several decision trees to boost performance. To illustrate, decision trees can be described as an algorithm method that compares a data point X with set limits $A, B, and\ C$, as can be seen in Figure 2.5. In a similar fashion to a neural network or any other ML algorithm, a decision tree can be of regression or classification type. In the classification type of decision trees, the output is one particular class out of a pre-defined set of classes, and the algorithm is called a RandomForestClassifier. However, this thesis focus will be on the RandomForestRegressor. The regression type of decision tree outputs something that can be considered a real number, e.g. the pricing of a house or the outlet pressure of a transfer pump. The RandomForestRegressor is just like the neural network part of the supervised ML algorithm family. Thus, it requires a training phase to adjust decision tree limits. Training the RandomForestRegressor includes re-training all the decision trees until the most accurate results are acquired. This means that the limits $A, B, and\ C$ can be adjusted until the most accurate result according to training data output is achieved.

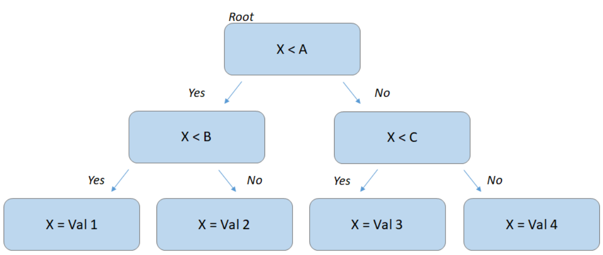

Figure 2.5: An illustration of the Decision Tree algorithm structure.

The method of ensemble learning is used in a way that takes the average output of all decision trees to make predictions more accurate, where a majority vote of all the decision trees in the RandomForestRegressor is conducted. In the regression type of RandomForestRegressor, the average output value of all decision trees is used, as can be seen in Figure 2.6. To go a bit further into the general functionality of a RandomForestRegressor, each decision tree starts with a root, as can be seen in Figure 2.5. To decide each limit (seen as *A, B, and C* in the example, a penalty function is introduced. Oftentimes, the penalty function *sum of squared residuals* is used. By trying all possible ways to divide the data by adjusting *A*, a penalty score is obtained that shows, on average, how accurate the output is for each value of the limit *A*. The limit that obtains the lowest score is then chosen. This process is then iterated, where a minimum of data points per *leaf* can be set to decide when to stop iterating. Repeating this process then yields the limit values of *A, B, and C* that give the best predictions. In the case where there are several input variables, which is the standard situation, the penalty function score is simply compared between the input variables to again chose the limit with the lowest score across all input variables. When all the decision trees have been trained, the average output of all decision trees is chosen as the main output of the RandomForestRegressor, as can be seen in Figure 2.6.

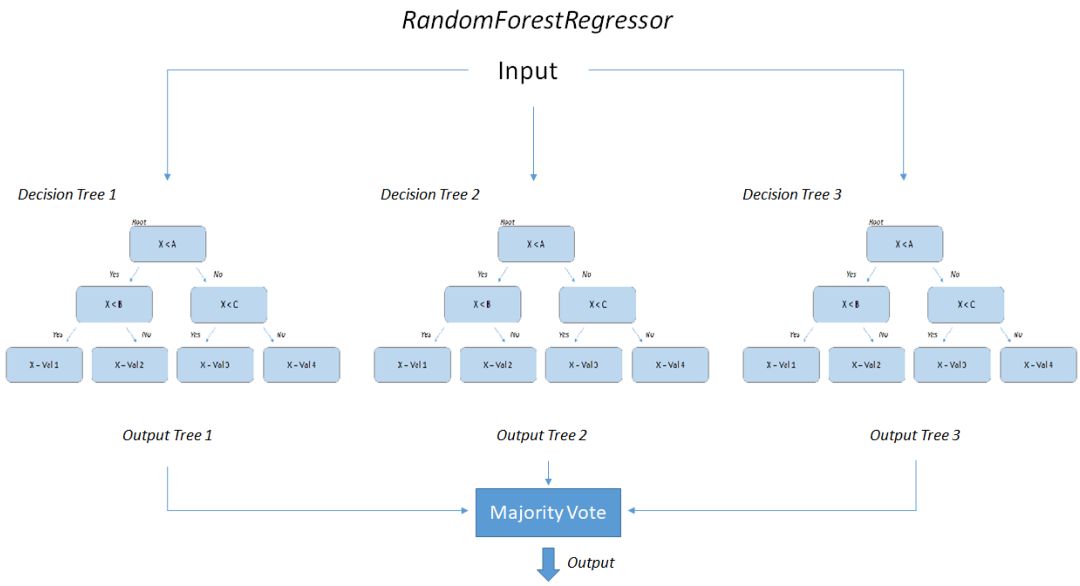

Figure 2.6: An illustration of the RandomForestRegressor algorithm structure.

Feature Selection

A large part of ML algorithm performance lies in selecting the right features that contain relevant information in regard to the target signal. The importance of doing so is widely known and discussed by *V. Kumar et Al.* in [7]. This is especially important if the data set contains hundreds of features to choose from. Many features contain information of a similar type, and some features may even contain information that is completely irrelevant to the ML algorithm. This is discussed by *P.E. Pintelas* in [18] where different qualities of features are discussed, and furthermore how these features are categorized, selected, and evaluated. The methods of selection can prove to be of big importance when selecting features as they can affect computational time, correlation, and general effectiveness. The previous master thesis report [20] also discussed the difficulties when trying to choose the most relevant features, as many signals are closely correlated to each other. Using two features that are highly correlated to each other might be needless. In addition to this, a majority of features can have a low correlation to the target signal, making it harder to properly choose the right features. This suggests that looking more into physical and reasonable connections between features and target signals can be the way to go. The topic of feature selection is also examined by *I. Guyon and A. Elisseeff* in [5], showing results that using two features that are highly correlated with each other gives no more information to the ML method. However, using features that are anti-correlated to each other can help model performance. Additionally, it is also important to remember that choosing the right subset of features is more important than finding single relevant features. A feature that can seem irrelevant alone might be of great use when combined with other features.

New ways to tackle the problem of feature selection and thus reduce the dimensions of the training set is to use ML algorithms to select the proper features. *A. Verikas et Al.* present results in [21] that shows improved accuracy in predictions when choosing features by using a feedforward neural network. In addition to this, *R. Weber et Al.* present in [13] methods that utilize the Support Vector Machine types of ML algorithms to find the most optimal features. Even though the implementation of this type of approach might be out of scope for this thesis, it presents a possible area for future work.

Balancing Data

Research shows that ML algorithms perform poorly when being trained on imbalanced data sets [14]. A lot of collected data generally contains information from similar environments and positions, which can boost the performance of these conditions. On the other hand, the algorithm's predictive power is significantly reduced when presented with minority data. Training algorithms on data sets that contain very few instances of certain types of data points usually produces biased ML algorithms that have a higher predictive accuracy over the majority of data, but poorer predictive accuracy over the minority types of data.

An important step that can boost overall ML algorithm performance is data balancing. When an ML algorithm is trained on imbalanced data sets it is easy to overfit the model since data is not always naturally distributed evenly among the different features. When using 10-cross validation to evaluate an imbalanced dataset versus a balanced dataset *Y. Yao, et al.* [22] found that the balanced dataset exhibits a 42% improvement over the imbalanced data. The model trained on the balanced training set also performed much better using a leave-one-out validation when discovering new parameters, (in this case predicting the abilities of new alloys). The earlier master thesis project [20] presented results suggesting that balancing the data would not improve prediction results. However, their results did not improve due to a faulty implemented balancing. This was later fixed by Saab, as an improved balancing algorithm was developed which boosted ML algorithm performance.

2.5 Anomaly Detection

The area of finding worrying and unusual patterns in flight data with the help of ML algorithms is something that can be used with good results to increase security [10]. Research has been conducted which reveals that patterns can be found in flight data that would otherwise go under the radar, highlighting the potential of the subject [9] [15]. To describe a central topic of this thesis of finding anomalies in flight data, a thorough description of anomalies and anomaly detection is first presented here. As described shortly above in the background, Section 1.1, an anomaly can be defined as when flight data deviates from the prediction interval for an extended time. That is, temporary spikes in the flight data which deviate from the prediction interval for a short period are not considered anomalies. The previous master thesis implemented this effect by introducing an accumulated sum, which alarms for an anomaly if this accumulated sum exceeds a manually set threshold. This allows for anomaly alarms only if the *area* between the prediction interval and deviating data is significant. The predicted values from the ML algorithms can differ from the real values due to many reasons. A poor training set for the ML algorithms, faulty data balancing, or differences due to uniqueness in aircraft parts. As each component is hand-built the overall fuel system properties can change drastically from aircraft to aircraft. This is where the use of a trained residual model comes into play. The use of a residual model is great for diagnostics, and by training it with ML algorithms and adding a prediction interval it can handle model inadequacy as well.

By looking at the residual in Figure 2.7 there is a difference of a little more than 20% at around 1500 seconds, which at first thought might be an anomaly. This is not the case however as we can see by the trained model shown in the right-hand plot of Figure 2.7. The trained model along with its prediction interval shows that this difference in the residual is expected at that certain working point, which means that it is in fact not an anomaly. Again, it is when the residual is outside of the prediction for an extended period of time and not coinciding with the trained model that we can start to suspect an anomaly. In addition to this, there can be several reasons if FD were to be classified as an anomaly that needs to be taken into consideration. Thus, anomaly detection should be regarded as an alarm that indicates *something* unusual, rather than the direct classification of an anomaly. The alarm could also be triggered by e.g. poor ML algorithm training, or something faulty in the RM.

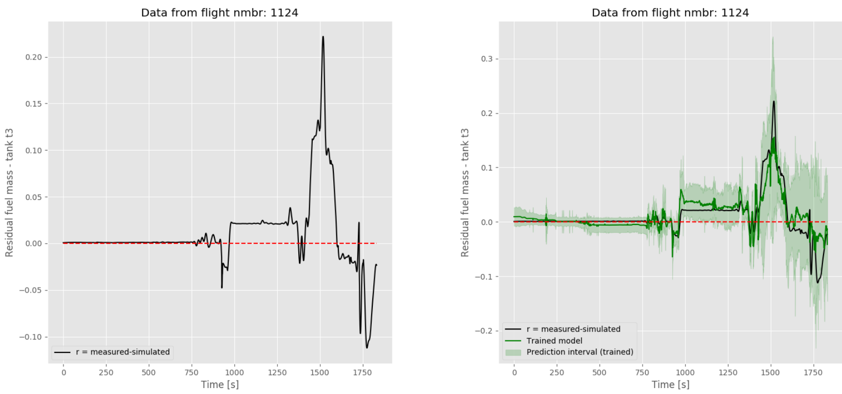

Figure 2.7: Left figure showing the measured flight data residual of the fuel mass. More specifically, the difference between measured and simulated fuel mass. The figure to the right shows the added prediction interval and predicted residual, generated by the neural network and RandomForestRegressor.

3 Method

The main objective of this master thesis is to explore the possibilities of improving anomaly detection in the fuel system of Gripen E. This is done by gaining new knowledge about the three chosen target signals, flight data, and the ML algorithms of the RMs to improve the DM in needed areas. Hopes are that improving the DM to generate more accurate simulation data can facilitate anomaly detection, and is a central part of this thesis. The thought is to make it easy to develop a new anomaly detection model of an arbitrarily chosen signal of the Gripen E by following the steps presented in this chapter.

The method procedure does not only aim to gain more information about the fuel system behavior, but it is also of interest to acquire a better DM as well as data processing procedure to favor and make more accurate ML-algorithm predictions. Whether improved simulation data favors ML algorithm predictions is a central topic of the thesis. In addition to this, another problem of interest is the balance between improving the DM and ML algorithm conditions.

The thesis work carried out is divided into two parts with the first part managing the RMs being done in *Jupyter Notebooks* where methods, ML algorithms, and data processing are done in *Python* using open-source packages such as *Pandas, Numpy*, *Scikit-learn* and *Keras/Tensorflow*. Secondly, the DM which contains a simulation model of the fuel system as a whole is used to generate simulated data of the interesting flight data signals.

3.1 General Method: Training the RMs to predict $\hat{r}(t)$ and $I(t)$

The main work of the project to gain knowledge about how to improve anomaly detection is done in *Jupyter notebooks* to investigate $\hat{r}(t)$ and $I(t)$. The three signals of investigation are the fuel mass in tank 2 $y_{meas,fm}(t)$, transfer pump outlet pressure $y_{meas,pr}(t)$ and flow $y_{meas,fl}(t)$, and are chosen such that information about different properties in the signals can be investigated. It is believed that the outlet pressure possesses vastly different behavior than the flow, and the fuel mass possesses accumulated effects that can reveal modeling flaws. This is an important aspect to consider if the same procedure is done with new signals and to also use the information from already investigated signals. To handle each specific target signal, a corresponding RM for each signal is implemented. The RMs consist of one *Jupyter Notebook* each and handle all the data pre-processing, filtering, ML algorithm training, and model evaluation for each corresponding signal. More specifically, a **neural network** is used in each RM to generate predictions of $r(t)$ using a specified subset of flight data, chosen by the set of signal features. Additionally, a *RandomForestRegressor* is used to generate a prediction interval estimation $I(t)$. Looking at systematic deviations between simulated and measured data are then used to improve the Dymola submodels generating each of the three target signals, to acquire more accurate simulation data.

An overview of all the procedures done to handle data can be seen in Figure 3.1. Most procedures to find improvement areas of the DM are also done in the notebooks, where signal investigation exposes situations and environments where the models are faulty. Flaws in the DM, in this case, can be thought of as situations where the simulated signal is very inaccurate. Corresponding RM flaws is shown where $\hat{r}(t)$ heavily deviates from the measured residual $r(t)$. A majority of the code in the notebooks is aimed at facilitating the training of ML algorithms to make the prediction of the residuals more accurate, but also to see where ML algorithms are inaccurate and where the Dymola model needs improvement.

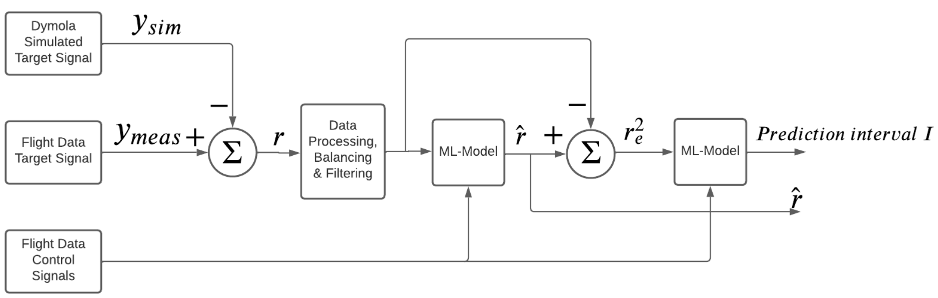

Figure 3.1: Overview of the method to develop a model generating $\hat{r}(t)$ and $I(t)$.

Training & Evaluation data

The first step in the process is to split the data set into a training set and an evaluation set. The training set is chosen to contain flights with data of great variety, consisting of data from different environments and settings such as altitude, velocities, and flying angles. This is to facilitate and prepare ML algorithms for a variety of flights and is thought to reduce the risk of having flights misclassified as anomalies when being presented with more unique situations. It is of great importance to only include flights with no anomalies in the training data, as ML algorithms otherwise might learn to predict the anomalies making large deviations in the data go under the radar. Secondly, eleven test flights should be chosen to evaluate model settings. The test flights are chosen such that the initial five flights were *nicer*, meaning that the pilot flew controlled and calm resulting in flight data that is commonly seen in the training data. The following five flights are messier, which results in more uncommon sequences of flight data. In addition to this, two flights that contain clear anomalies are also chosen to verify that anomalies in the flight data still are classified as so. If possible, it is preferable to have additional flights containing anomalies for verification but as anomalies in flight data are sparse, this might not always be possible. Having a variety in the evaluation set of flights is important as it is preferred to verify that ML algorithms learn all the corner cases, and so that unique flights where the pilot flies tougher do not get misclassified as anomalies.

Filtering & Data cleaning

Before the data is used to train ML algorithms, a few steps need to be taken to pre-process the data and add specific signals not present in the data set. The processing steps that are performed in this project work and steps that also need to be considered when creating a model for an arbitrarily chosen signal from the fuel system of Gripen are listed as follows.

- Filling in missing data points of the dataset by either removing the data row or by artificially creating valid data points. Several methods to do this are discussed in [4], but the method used here is simply to remove the data row.
- Lowpass filtering of the measured and simulated signal before the residual is created. It is important to adjust the Butterworth LP filter parameters to suit the signal of choice so that a good representation of the signal is acquired.
- Add residual signals between measured and simulated target signals to the dataset.
- Removing non-interesting parts of the data. For example, the first seconds of a flight where the aircraft stands still might contain noisy, generally misleading information in the data. This data can be left out to not skew the data set. In this project, the first 60 seconds of all flights are removed.

Note that these steps need to be revised when developing a model in *Jupyter Notebooks* for a new target signal, as signal properties differ a lot. Some signals might require unique filtering due to stochastic processes which can skew data and not give an accurate representation of the signal. This is important since it highlights the uniqueness of each signal and the properties that come with each signal, and shows that each signal needs to have its own model with a thought-through signal processing procedure.

Feature Selection

The data set containing information from test flights consists of approximately 120 features (signals) which consist of both binary and numerical values. Training the ML algorithms on all available signals would be a cumbersome and time-consuming process. To properly train the ML algorithms there is a need for a reduced set of features. The features should be chosen such that ML algorithms can find connections between flight data of the reduced feature set and the target signal. This is something that needs to be reconsidered for each target signal as every signal possesses different behaviors. Each target signal chosen in this project is unique and has properties that differ from one another. To gain a suggestion of which kind of feature to choose, a linear correlation between features in the data set and the target signal can be done. This is done in this project by using the *Pandas* function *.corr()*, which generates a linear correlation matrix for all the signals listed in the data set. However, most thought to choose a suitable set of features is done by thinking about reasonable connections between signals in the data set and the target signal. Note that this process is done in an iterative way, where trial and error reveals a lot of what the ML algorithms learn from different sets of features. The procedure consists of adding or removing certain features in order to re-train ML algorithms to see how it affects $\hat{r}(t)$ and $I(t)$. The process can also help exploit weaknesses and strengths of the RMs, and a lot of information is gained by the choice of features.

Adding Artificial Signals

The addition of artificial signals is one important step to facilitate ML algorithm training. Adding a self-built signal can contribute to information that is lost due to filtering, or add new information by highlighting patterns in the data that are known. Building a helpful artificial signal is tricky and needs to be well thought out. For example, the previous thesis project added the aircraft's vertical and horizontal angles θ and ϕ which were calculated from the measured accelerations in the aircraft. Oftentimes there is a pattern in data that can be exposed to ML algorithms by adding the artificial signal. There is no direct guide on how to build a well-working artificial signal as each signal has unique properties, but it is a possibility that needs to be considered when working with a new model. When an artificial signal is built, it is simply added to the data set when training ML algorithms. The signal needs to be added to the test flight data as well.

Data balancing

To counteract the uneven distribution of data points in the training data the data set needs to undergo balancing. What is meant by this is that data points from certain flight environments and settings are overrepresented, resulting in the ML algorithms focusing too much on commonly occurring types of data and not on sparse data. This will result in poor predictions when being presented with test data that is rare in the data set. To obtain a more balanced data set *stratified sampling* can be used, a technique used to modify unequal distributions to create balanced data sets. If the quantity of rare data is insufficient, the method tries to increment the size of rare samples to balance the data set by randomly copying rare data points. If some type of data is overrepresented in the data set the same method can be used to randomly select data points to remove. The procedure as a whole is called stratified sampling and combines the usage of oversampling and undersampling to balance the data set. Although stratified sampling mostly contributes to a better balance to improve ML algorithm performance, it is important to also consider the loss of information that comes with removing data points, something that can be costly when trying to train ML algorithms.

The method of stratified sampling was developed in the previous thesis project, and refined by Saab to accurately balance the data set according to a set of chosen balancing features. The balancing is done with three features in this project. The stratified sampling method uses both oversampling and undersampling to balance out the dataset by dividing the samples into bins. Bins represent different ranges of values, and the bin discretization values are user chosen. The same number of samples are randomly picked from each bin. A sample may be picked more than once. By re-choosing the same samples several times from the same bin the number of rare data points can be increased. All the data points chosen from each bin are then combined into one set, representing the balanced data set. Three parameters decide the balancing. Firstly the features chosen to balance around, secondly the bin discretization values, and lastly, the number of samples to be chosen from each bin. To avoid the risk of having a single *data outlier* alone in one bin being re-chosen a maximum limit of how many times a single data point can be chosen is set. This could otherwise skew the data set if a faulty data point is chosen. When choosing the set of features to balance around, it is important to only consider signals which represent the unequal distribution of data. If the signal chosen already is equally distributed across the relevant levels, there is nothing to gain. By setting the maximum bin size and maximum times a sample can be re-chosen together with the choice of features a more equal balanced dataset can be acquired.

Generation of Target Signal Residuals and Prediction Interval

The target signals used are generated by taking the difference between the measured and simulated signal of choice, creating the target signal residual.

$$r(t) = y_{meas}(t) - y_{sim}(t) \tag{3.1}$$

A *neural network* is used to obtain the predicted target signal $\hat{r}(t)$, trained with a set of carefully chosen features and r(t) as the target signal. A *RandomForestRegressor* is used to generate the predictions of $r_e(t)$ seen in Equation 3.2, where $\hat{r}_e(t)$ is the predicted output from the algorithm. This is used as a prediction interval for the measured residual signal r(t). To train the *RandomForestRegressor*, the same set of features are used as when training the *neural network* but together with $r_e(t)$ as a target signal. The *RandomForestRegressor* is used to exploit the fact that a positive target signal is used, removing the difficulties that come with extrapolation.

$$I(t) = \hat{r}_e(t), \quad r_e(t) = (\hat{r}(t) - r(t))^2 \tag{3.2}$$

Average Prediction Interval

In order to see how all the listed areas of improvement affect the performance of models, and more specifically on ML algorithm predictions it is possible to analyze the resulting changes of the prediction interval $I(t)$ and the predicted residual $\hat{r}(t)$. This is also a method to expose weaknesses and strengths in the model, as faulty anomaly classification and large prediction intervals can reveal in what types of situations the model struggle. On the other hand, very accurate predictions and a small prediction interval point toward a very accurate model, where the strengths of the model can be extracted. To get a numerical value to use for the comparison of different flight settings, three numerical measurements were implemented to map a score to each model setting. The first one measures the average prediction interval per data point.

$$I_{avg} = \frac{1}{N} \sum_{F_X} |I_{upper}(t) - I_{lower}(t)| \tag{3.3}$$

Where N is the total number of data points in-flight data F denoted with flight number X, and I_{upper} and I_{lower} are the upper and lower prediction interval limits respectively. The sum runs over all the data points in the data set F_X. Note that the score I_{avg} only can be used to compare different settings of the same flight, and the score cannot be used to compare results from different flights with each other.

Average Predicted - Measured Residual Score

In addition to the average prediction interval score, an average predicted-measured residual score is also calculated, which in short is the average difference between the predicted residual $\hat{r}(t)$ and the measured residual $r(t)$.

$$r_{avg} = \frac{1}{N} \sum_{F_X} |\hat{r}(t) - r(t)| \tag{3.4}$$

Where $\hat{r}(t)$ is the predicted residual output from the neural network, and $r(t)$ is the measured residual between flight and simulation data. This score gives a sense of on average, how accurate the predictions are compared to the measured values.

Average Simulated - Measured Signal Score

Lastly, a score showing how accurate the simulated data is compared to the measured data is used and is in a similar fashion as I_{avg} and r_{avg} an average difference between the simulated and measured target signal.

$$y_{avg} = \frac{1}{N} \sum_{F_X} |y_{meas}(t) - y_{sim}(t)| \tag{3.5}$$

The scores are used as an indication of whether changes made, features selected, filtering, etc improved the model accuracy and can as earlier mentioned only be used to compare data from the same flight. In this case, a smaller value of the scores would indicate a greater accuracy. The ideal score would be zero, as the predictions then perfectly would follow the measured data. Important to note is that the scores need to be considered as one part of the puzzle and that other aspects of the results also need to be considered. For example, the results would be bad if a majority of flights were misclassified even though an average smaller prediction interval could be seen. It is beneficial to see trends in the results of several flights, but one needs to keep in mind that single flights can reveal a lot of information about model weaknesses and strengths.

Anomaly Detection & Analysis of Results

As the central goal of this thesis project is to improve anomaly detection, here is the part where it is made sure that the anomaly detection model actually is improved. To decide what should be classified as an anomaly and not, a threshold has to be set in each RM that decides the maximum area of deviation the measured residual can accumulate. This needs to be overlooked for each signal, as each signal greatly differs in magnitude. A rule of thumb is to not let temporary spikes be classified as anomalies (as they are in fact, not anomalies), but extended periods of deviation should be alarmed for. This threshold is manually set in each RM after a reasonable value is decided. Note that all the previously discussed subjects contribute to improved anomaly detection in a way that the model becomes more accurate.

Additionally, it is desirable to expose flaws and highlight well-performing areas of the DM, where analysis of the results plays an important role in the preparation of good quality discussion and conclusions. To improve the DM, each of the mentioned steps above is worked with in an iterative way, taking necessary actions to remedy exposed flaws. It is of great importance to look at several flights containing a great variety in flight data, so that corner cases can be covered. Work is done in a way that analyzes results both by looking at trends of all the test flights, but also by looking at individual flights to expose certain situations where the model fails. It has to be assured that a very small fraction of the flights are misclassified as anomalies, and accurately predicted as non-anomaly-containing flights. Only then do better predictions and a smaller prediction interval make the model better. It can be of great use to look into the data of when the model predictions are very accurate with a small prediction interval as this shows the strengths. On the opposite, regions where $\hat{r}(t)$ heavily deviates from $r(t)$ and where $I(t)$ is large can indicate where the models need improvement.

3.2 General Method: Improving Dymola Model

To go into detail about a general method to find improvement areas in the DM, and implement such areas in Dymola, this section covers the method used to specifically find and build more accurate simulation submodels in Dymola. A complete model of the fuel system in Gripen is implemented in Dymola, where existing submodels of the system generate simulation data of the transfer pump outlet pressure, transfer pump outlet flow, and the fuel masses of different tanks (Only the fuel mass in tank 2 is used in this project). These signals are only some of the outputs of the fuel system DM. This improvement stage aims to find situations where the simulated signal deviates from the measured and the reason behind this deviation. The end goal is to reduced the magnitude in $r(t)$ as a result of smaller deviations between $y_{sim}(t)$ and $y_{meas}(t)$. If the right control signals in the aircraft can be linked to behaviors in the measured target signal, it is easy to reproduce the signal in DM as the control signals are easily accessed there. A great way of finding the links between target and control signals is to plot the signals over each other and compare the binary values of the control signals to stabilization levels in the target signal. Linear correlation is also used to suggest the most relevant control signals. Ideally, one wants to find direct connections between binary control signals in the data set such as tank valve signals and the target signal, where direct connection to the target signal is easy to determine. It can also be beneficial to analyze the connections of measured signals such as altitude and fuel consumption to see correlations, but the direct correlation between the target signal and these types of measured signals is harder to detect visually. Below, the iterative working method is visualized, as can be seen in Figure 3.2.

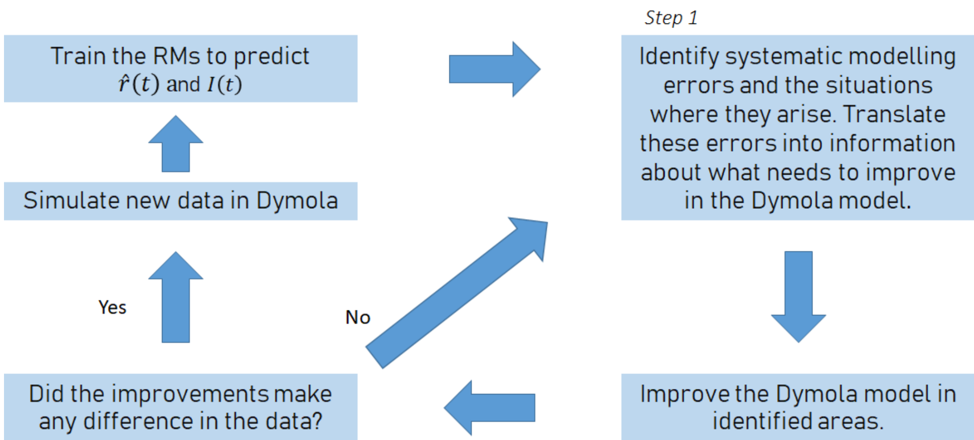

Figure 3.2: Working method used to develop and refine the Dymola model generating simulated target signals.

The procedure of finding connections between measured data of the target signal and control signals in the data set can be tricky and heavily relies on an extensive investigation phase where signals of interest are plotted and analyzed. By using logical gates the patterns seen in the data can be modeled to create an improved Dymola model. Figure 3.3 shows an overview of the steps taken to investigate and improve the DM.

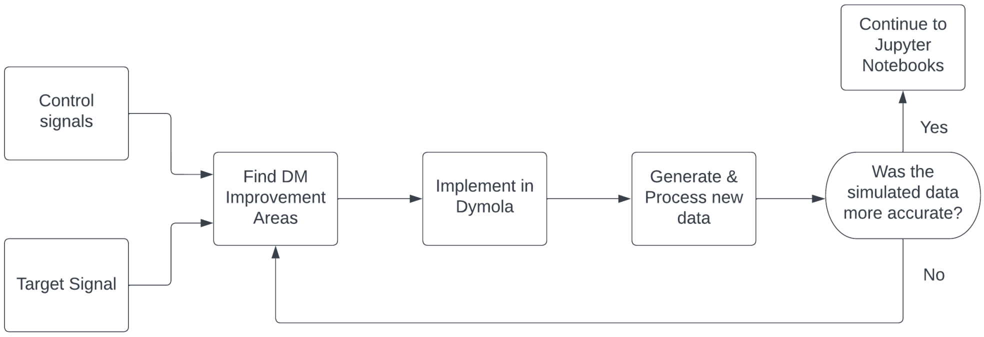

Figure 3.3: Process of improving the Dymola model generating the simulated signals of the transfer pump outlet pressure, transfer pump outlet flow, and fuel masses.

4 Results

This chapter presents the results. Initially, a brief overview of the results is presented together with the investigated and improved areas. General results regarding the fuel system as a whole are also presented here. Following this comes the presentation of in-detail results of model improvements done in the transfer pump outlet pressure RM, transfer pump outlet flow RM, and fuel mass of tank 2 RM. This includes work done to improve conditions for ML algorithms, signal and data processing, an improved DM corresponding to each target signal, and a black box analysis for each of the target signals to better understand how the ML is trained. Figures include illustrations of how the improved DMs simulating the given target signal perform, and the accuracy of ML predictions when compared to the measured signal residual.

4.1 General Fuel System Results

This section presents the general results and improvement areas of the fuel system models, which gives an overview of what areas have been investigated and improved. Results consist of three improved Dymola models generating simulated signals of the transfer pump outlet pressure $y_{sim,pr}(t)$, the transfer pump outlet flow $y_{sim,fl}(t)$, and the fuel mass in tank 2 $y_{sim,fm}(t)$. A lot of focus is put into signal processing and ML algorithms, where one individual notebook is developed to analyze each signal residual. This results in three different *Residual Models*. Each notebook investigates properties and behavior in the residual $r(t) = y_{sim}(t) - y_{meas}(t)$ of each target signal to find improvement areas in the DM, and detect anomalies in measured data. The three models are developed in *Jupyter notebooks* in *Python*. Each model contains methods to filter, pre-process and balance the data set. Code to investigate interesting signals of the data set and train ML algorithms is also present in the models. An overview of the improvement areas can be seen in Figure 4.1.

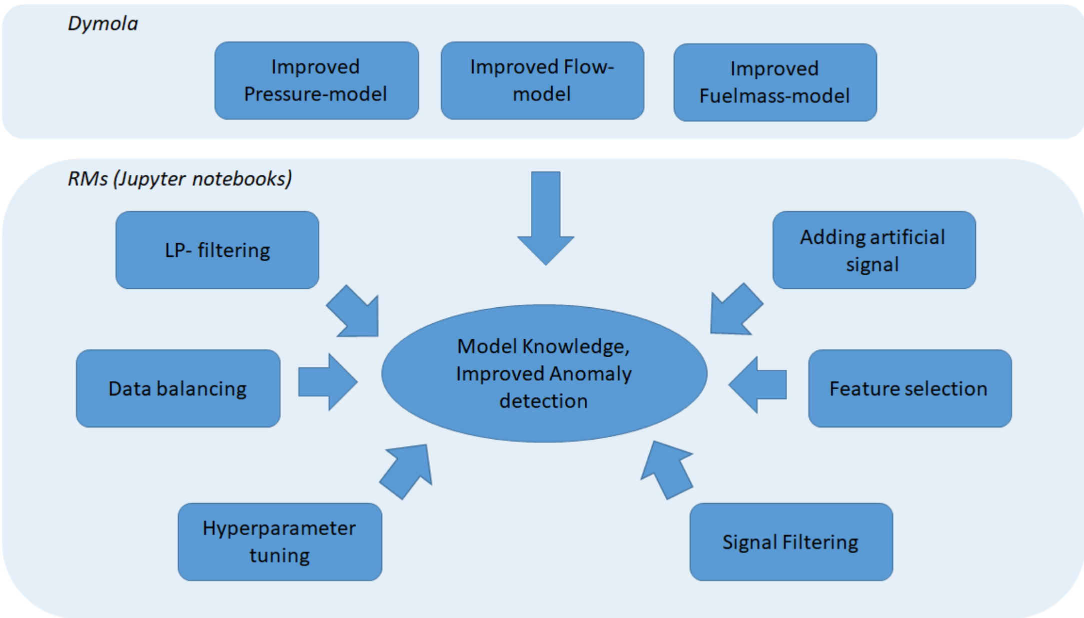

Figure 4.1: An overview of improvements made in Dymola and the Residual Models implemented in Jupyter notebooks.

Data Balancing

Two different sets of features are used when balancing the data set in order to best represent the data before training ML algorithms. When balancing the data set in the outlet pressure and flow RMs, the transfer pump valve command signal is used as a balancing feature instead of the fuel mass in tank 2, which is used when balancing in the fuel mass RM. This is motivated by the result that ML algorithms of transfer pump outlet pressure and flow are heavily dependent on the transfer pump valve command when predicting their corresponding target signal. The features used when balancing the data set of each model are listed below, together with the bin limit resolutions.

List of Features Data Balancing: Fuel Mass Tank 2

- Vertical flying angle θ [Degrees] - *Bin size of 5 ranging in between [-25, 25]*

- Static surrounding pressure [kPa] - *Bin size of 20 ranging in between [20, 120]*
- Measured fuel mass in tank 2 [Kg] - *Bin size of 50 ranging in between [50, 600]*

List of Features Data Balancing: Transfer Pump Outlet Pressure & Flow

- Vertical flying angle θ [Degrees] - *Bin size of 5 ranging in between [-25, 25]*
- Static surrounding pressure [kPa] - *Bin size of 20 ranging in between [20, 120]*
- Transfer pump valve command signal [4 different levels] - *Bin size of 1.5 ranging in between [1, 6]*

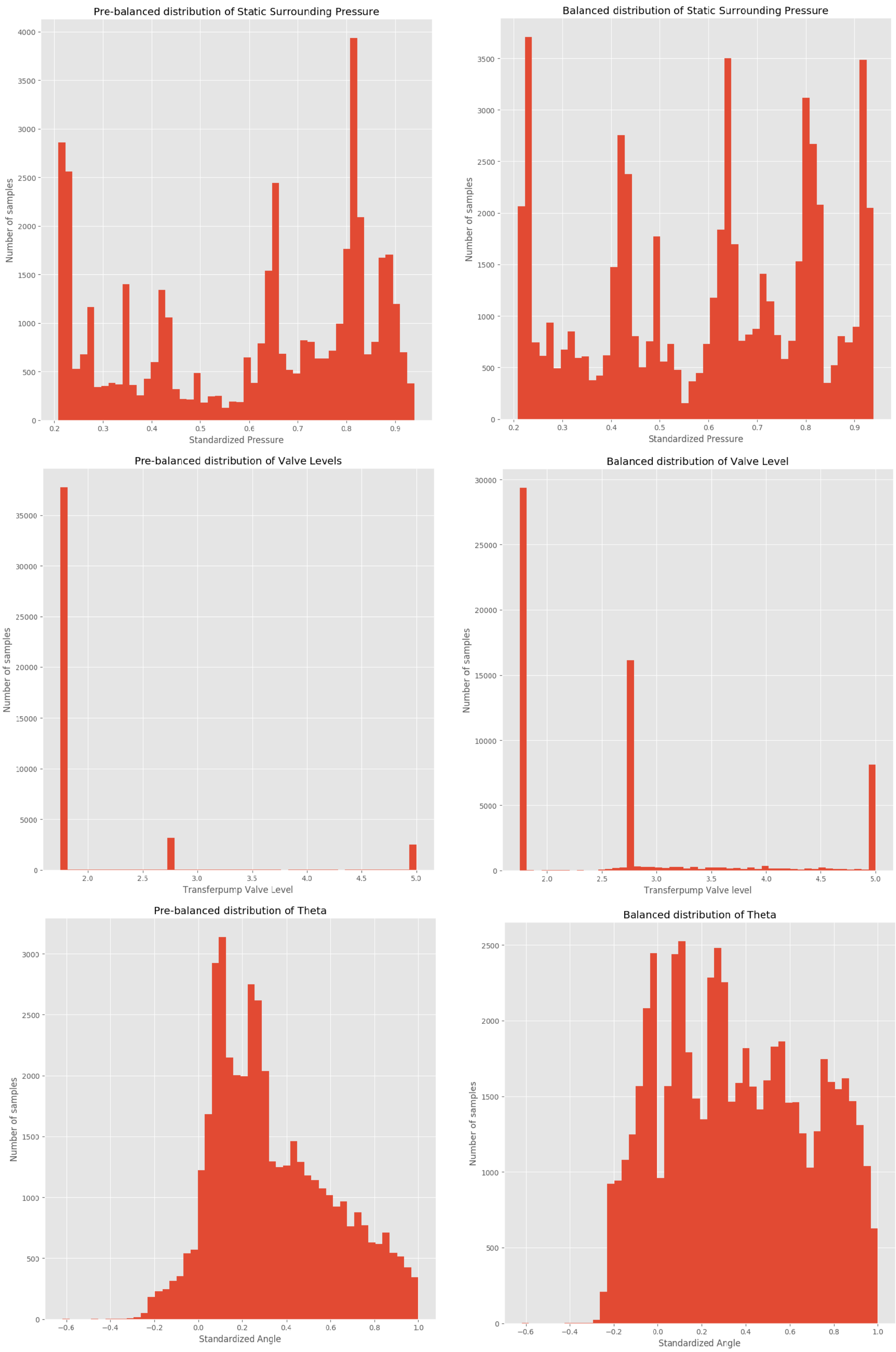

Figure 4.2: Pre-balanced distribution to the left and post-balanced distribution in data of chosen features used in the transfer pump outlet pressure and flow RMs when balancing the data set.

Signal Variation due to Wear & Component Degeneration

An important finding is the variation of the inspected transfer pump outlet pressure signal. As every part in the aircraft is unique the performance is affected should one part be replaced, either as a consequence of the uniqueness of the part or due to component degeneration. This is illustrated in Figure 4.3 where the pressure for a few tanks in the aircraft is presented before and after the exchange of a new transfer pump. The general level of pressure in the tanks increases significantly after the old transfer pump is replaced, which is one aspect that needs to be taken into consideration when modeling. This can be seen as a weakness of the system that can cause problems both for machine learning algorithms and when trying to model the signal. It can be cumbersome for ML algorithms to learn data patterns if level differences up to 20 kPa in the target signal can be seen for the same type of control signals, which most likely will result in poorer predictions and a larger prediction interval. This is troublesome for ML algorithm training as the same type of input signals in the training data can be connected to an output signal which greatly varies. In regards to modeling the signal, this phenomenon makes it almost impossible to create a static model in Dymola that simulates an accurate simulation signal, unless this degeneration effect can be implemented into the model. This is due to the structure of the DM, and the control signals of the DM that build the simulated output.

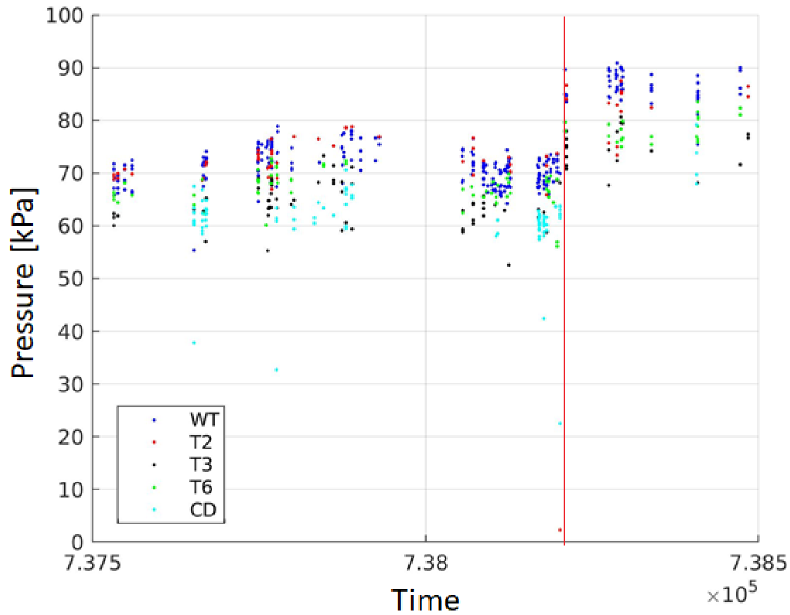

Figure 4.3: Illustration of the general level deviation in the transfer pump outlet pressure. Clusters show the general level of pressure for each tank in an aircraft. The red line marks a point in time when the transfer pump was replaced, and it is possible to see a level difference in outlet pressure between the old and new transfer pumps.

ML Algorithms and Hyper parameters

The machine learning algorithm used in the RMs is a neural network for predicting the target signal residual $\hat{r}(t)$, and a random forest regressor to predict the variance of the neural network prediction error, $(\hat{r}(t) - r(t))^2$. The neural network uses two hidden layers, with the first hidden layer having a size of 8 artificial neurons and the second layer size of 4 artificial neurons. The neural network is implemented with the hyperbolic tan function as an activation function for the hidden layers, and the random forest regressor uses the mean squared error as a penalty function.

A small grid search was done to see whether more suitable hyperparameters of the neural network could be found. This included trying the *logistic sigmoid function* and the *rectified linear unit function* as activation functions. Additionally, these changes were combined with different amounts and sizes of the hidden layers. Three layer settings were tested, first to three hidden layers with the corresponding sizes of [40, 40, 40], and then to two and one hidden layers with sizes [80, 40] and [50] respectively. Each of the parameters is set in the function call. However, none of the investigated hyperparameter settings showed no improved results. Thus, the previous settings of using the hyperbolic tan function and hidden layers of 8 and 4 are used. The process of in-depth investigation of ML algorithm hyperparameters is considered out of scope for this thesis, which explains the not so throughout hyperparameter search.

Anomaly Detection

A central aspect when improving the model is the ability to correctly detect anomalies in measured flight data. A lot of focus has been put into making ML algorithms more accurate and shrinking the prediction interval $I(t)$ for given test flights, but this effect would have no meaning if normal flights are incorrectly classified as anomalies. Thus, after each model change in both the RMs and DM for each target signal the ability to accurately detect anomalies is tested in 11 selected flights, where sets of flights are representing a variety in flight conditions. Test flights include two flights with numbers 1175 and 1201 that contains anomalies in the outlet pressure, and the same two flights plus an additional flight with number 1081 that contains anomalies for the outlet flow and fuel mass in tank 2. Temporary spikes of data outside the prediction interval are thus tolerated. Overall results show that only one flight is misclassified as an anomaly for the outlet flow and fuel mass, flight number 1091. The remaining flights are all correctly classified both in regards to anomaly detection and not detecting any anomalies when none were present. Table 4.1 shows all the tested flights for each target signal with the corresponding result with all the total improvements done in each RM and DM. For each flight, whether the flight data contains an anomaly is first answered by *Yes/No*. Secondly, *Yes/No* shows if the data is predicted to contain an anomaly or not.

To accurately represent flight test data in a variety of conditions, the flights selected for testing ML algorithms and model performance are chosen such that some are easier and some are tougher. The first five flights listed in Table 4.1 can be considered easier, where the pilot has flown relatively easily, not pushing the aircraft to do anything extraordinary. In terms of data, this means that the transfer pump and aircraft are not under any stress and most of the data in test flights are common in the training data. Flights 1081, 1086, 1088, and 1091 contain data from where the aircraft is flown in a tougher manner, having a higher fuel consumption, and general data which is rarer in the training set. Flights 1175 and 1201 contain data that classify as anomalies, in the sense that something broke or heavily deviated to unnatural measurements. This is also true for flight 1081 for the outlet flow and fuel mass.

Anomaly Predictions Outcome, Flight 39-9			
Flight Nbr	TP Outlet Pressure	TP Outlet Flow	Fuel Mass T2
1044	No/No	No/No	No/No
1047	No/No	No/No	No/No
1048	No/No	No/No	No/No
1052	No/No	No/No	No/No
1055	No/No	No/No	No/No
1081	No/No	Yes/Yes	Yes/Yes
1086	No/No	No/No	No/No
1088	No/No	No/No	No/No
1091	No/No	No/No	No/Yes
1175	Yes/Yes	Yes/Yes	Yes/Yes
1201	Yes/Yes	Yes/Yes	Yes/Yes

Table 4.1: Results of the anomaly detection algorithm for the three target signals investigated. Whether the flight data contains an anomaly is first answered by *Yes/No*. Secondly, *Yes/No* shows if the data is predicted to contain an anomaly or not. The only wrong prediction is highlighted in red.

General List of Improvement Areas

Presented below is a general list of the procedures followed in each notebook. When finding ways to improve the RMs, and lastly also the DM, these were the main steps that were re-occurring in each notebook. The steps listed are generally followed in each RM, with a few exceptions. Each procedure exception from the list together with additional steps are presented for each RM in the sub-chapters below. Generally, the RM settings mostly differ in the selection of features, data filtering, and the addition of artificial signals. The signal analysis and implementation of the improved DM is also a unique process as each target signal possesses different properties.

1. Low pass filtering of $y_{sim}(t)$ and $y_{meas}(t)$.
2. Feature selection for the training of ML methods - Choosing suitable and relevant features.
3. Feature selection for data balancing in preparation to train Neural Network and RandomForestRegressor - Choosing features that are of importance and have a good representation of the uneven distribution in data.
4. Creation of an artificial signal - Adding self-built signals to help Neural Network and RandomForestRegressor predictions.
5. Data balancing - Editing of the data balancing to get a more evenly distributed data set.
6. Implementation of improved DM - To acquire more accurate simulation data.
7. Analysis of result scores - Model score to compare different settings and improvements of DM as well as changes made in RM.
8. Analysis of target signal, RM, and DM results to exploit weaknesses and strengths of said models.

Results show that it is of great importance to redo the following steps for each new target signal, as the signals differ a lot in behavior and each setting needs to be revised. Especially the last step, as different patterns in data can reveal the best way to model each signal in Dymola. Additional steps that are introduced mainly come from the signal investigation, and are implemented due to certain effects that come with certain signals.

4.2 Results: Transfer Pump Outlet Pressure

This section presents results in the analysis of the transfer pump outlet pressure when looking at the outlet pressure residual $r_{pr}(t)$ of the transfer pump as the target signal. The section shows results from both an improved simulation model implemented in Dymola, as well as improvements that are done in the pressure RM (*implemented in Jupyter notebooks)*. The resulting RM includes processes such as signal pre-processing, filtering, data balancing, and the addition of an artificially built signal to help ML methods. In addition to this, features used for both ML methods as well as their impact on ML predictions are shown.

Improvement Areas of the Transfer Pump Outlet Pressure

The RM of the outlet pressure follows the general steps listed in Section 4.1, with the addition of two steps which are listed below. The steps are done in preparation for training of the neural network as well as the RandomForestRegressor to get the most accurate predictions of the outlet pressure residual $\hat{r}_{pr}(t)$ and a reasonable small prediction interval $I_{pr}(t)$. Features of relevance are chosen such that the general behavior of $r_{pr}(t)$ is accurately mapped by the ML algorithms, as well as picking out certain features to cover corner cases in flights. An artificial signal is also added to the data set and the data set is balanced to obtain a more evenly distributed data set. At last, the target signal along with carefully chosen control signals are studied which laid the ground for the improved Dymola model. Lastly, the two steps listed below are added between steps 5 and 6 in the *List of Improvement Areas*, as can be seen in Section 4.1.

- Pre-processing of data set - Removing data points when transfer pump is not active.
- Pre-processing of the outlet pressure $y_{meas,pr}(t)$ - Filtering out *outliers*.

These additional steps are motivated by the fact that the outlet pressure is not relevant unless the transfer pump is active. This is decided by a signal that represents the transfer pump valve command level, which has four set levels. The valve command can be 0, 1.75, 2.75, and 5, and is representative of the level of flow that is provided by the transfer pump. Thus, the data set which contains data from where the valve command is 0 is cut out, resulting in a data set that only contains data from the other valve command levels. In addition to this, a stochastic signal behavior was seen in the measured outlet pressure just at the beginning and end of each active region (where the valve level is not zero). It is believed that this stochastic behavior comes naturally when the transfer pump turns on and off, introducing random measurements of the outlet pressure as the signal lags behind the transfer pump valve position. This caused spikes in $r_{pr}(t)$ which motivates the second additional step, which removes these data points of stochastic properties. If not removed, the stochastic behavior would have been caught by the ML algorithms, being trained to model the wrong patterns.

List of ML Features

Below is a list of the features used to train ML algorithms to predict the outlet pressure of the transfer pump, $r_{pr}(t)$. The features displayed are carefully chosen to facilitate the training of the ML algorithms. Linear correlation analysis of the features available together with the outlet pressure was also conducted and helped when choosing the listed features. The linear correlation analysis was merely used as something to suggest interesting features. The features listed below make up the set used when training ML algorithms in the pressure RM, and are displayed with a short motivation of why they were chosen.

- **Target signal - residual of outlet pressure** $r_{pr}(t)$ [kPa] - Self explanatory. Included to train ML prediction output.

- **Static surrounding pressure** [kPa] - As the aircraft can fly on vastly different altitudes, the surrounding pressure can differ a lot. This is believed to have some sort of impact on the internal transfer pump pressure.
- **Pressure of fuel system before transfer pump** [kPa] - It is reasonable to think that changes in the pressure before the transfer pump can affect the outlet pressure of the transfer pump.
- **Temperature of fuel to engine feed** [C] - It is known that temperature and pressure are closely linked. A higher temperature results in a higher pressure if the volume is kept constant.
- **Vertical flying angle** [Degrees] - Different angles of the aircraft could expose geometric variations which could affect the outlet pressure.
- **Vertical acceleration** [m/s^2] - Vertical acceleration is often time linked to high fuel combustion in the engine due to certain pilot maneuvers. If the transfer pump is very active, data on the vertical acceleration could be connected to the transfer pump outlet pressure.
- **Transfer pump valve command signal** [4 different levels] - Crucial signal as the transfer pump outlet pressure is believed to be closely linked to how the transfer pump behaves in general.
- **Fuel consumption** [Kg/s] - A higher fuel consumption results in a more active transfer pump, something that affects the outlet pressure.
- **Aircraft velocity** [Mach] - Higher aircraft velocities results in a more active transfer pump.
- **Fuel mass in central drop tank** [Kg] - It could be seen that fuel drainage of the drop tank had an effect on the outlet pressure, and including this feature could cover corner cases otherwise classified as anomalies. This was due to drainage of fuel in the drop tank when there was little to no fuel left, causing great drops in the transfer pump outlet pressure.
- **Tank Valve command signal for tanks 1,2,3,6, Right/Left-wing and central drop tank** [On/off] - Each tank has a different geometry, and due to differences in parts and interconnections between tanks, changes in volume these binary signals are included.
- **Artificial signal** describing the dynamic behavior in the transfer pump - described in-depth in section *Effect of filtering and Addition of Artificial Signal*

To illustrate the importance of choosing a suitable combination of features the predictions of the outlet pressure residual for two different sets of features are displayed below in Figure 4.4. The right figure contains all the listed features, and the left figure shows predictions when signals for *tank valve command signal for tanks 1,2,3,6, Right/Left wing, central drop tank* and *tranfer pump valve command signal* are not included. This also shows that ML algorithms learn a lot by having information about different tank valve commands together with the transfer pump valve command signal. The results seem to show that by adding the binary signals indicating when fuel is taken from certain tanks, the prediction accuracy of the neural network increases. In addition to this, the included features also seem to reduce the prediction interval to a reasonable level while still not misclassifying any anomalies.

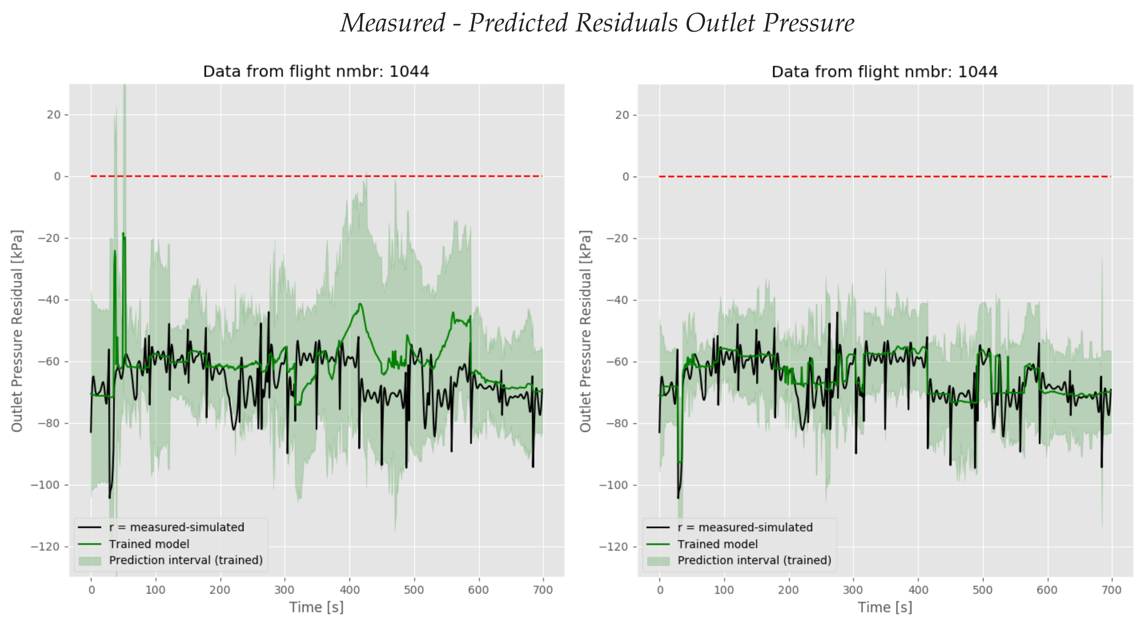

Figure 4.4: Old features (left) and updated features (right). The predicted residual $\hat{r}_{pr}(t)$ in green, measured residual $r_{pr}(t)$ in black and the prediction interval $I_{pr}(t)$ in light green.

To show the effect numerically by adding the *Tank Valve command signal for tanks 1, 2, 3, 6, Right/Left wing* and *Tranferpump valve command signal* signals as features, nine test flights are chosen and their corresponding *Average Predicted - Measured Residual* and *Average Prediction Interval* are presented below in Table 4.2. This is to see how the resulting predictions and prediction intervals are affected by the choice of features. As can be seen, the prediction interval for each flight heavily decreased indicating that ML algorithms benefit from the listed signals as features. The average difference between the predicted outlet pressure residual and the measured residual also heavily decreased, meaning that the neural network was more accurate in predictions with the new feature set. Note that the flights presented here do not contain any anomalies and are also predicted as so.

Transfer Pump Outlet Pressure Data, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	Original Features	Updated Features	Original Features	Updated Features
1044	15.71	3.53	42.81	29.41
1047	14.10	3.53	78.53	26.49
1048	14.23	3.27	51.47	26.16
1052	13.69	3.55	69.49	25.90
1055	22.18	3.72	71.97	26.47
1081	17.73	5.95	73.72	32.76
1086	9.55	3.68	83.46	29.66
1088	14.63	5.35	59.30	30.95
1091	16.76	4.27	69.50	30.73

Table 4.2: Columns showing the residual and prediction interval scores when adding tank valve command signal for tanks 1, 2, 3, 6, Right/Left wing, drop tank and transfer pump valve command signal in the set of features, compared to using the original feature set.

What can be seen in Table 4.2 is a large reduction in the average difference between the measured and predicted residual signals. In addition to this, the prediction interval is more than halved for a majority of the flights tested. This points toward more accurate predictions of

$r_{pr}(t)$ and more precise ML algorithms as the prediction interval is reduced. Similar improvements can be seen across all the flights, so the updated feature set seems to facilitate predictions for all the different test flights, even though different flying data are present in the data.

Effect of Signal filtering and Addition of Artificial Signal

This section presents the effect of adding an artificial signal to the data set and filtering out *early outliers* of the target signal. The artificial signal is built to help ML algorithms predict the outlet pressure residual by representing the dynamic behavior of the transfer pump. This property is lost when the data set is cleared of data from when the transfer pump valve level is zero. When the zero regions are removed, information about when the pump is active and not is lost. The signal is built by concatenating a series of impulse responses of a second-order LTI system whenever the pump is turned on. The coefficients of the LTI system were chosen arbitrarily in a way that made the impulse response decay reasonably fast. More focus was not put into the choice of LTI system, as it was sufficient enough to introduce a dynamic behavior in the artificial signal. This is illustrated in Figure 4.5, where the original *transfer pump valve level* signal is shown together with the artificial signal. The right-hand figure shows the signal after the zero region is removed, and how the artificial signal indicates when the transfer pump is turned on.

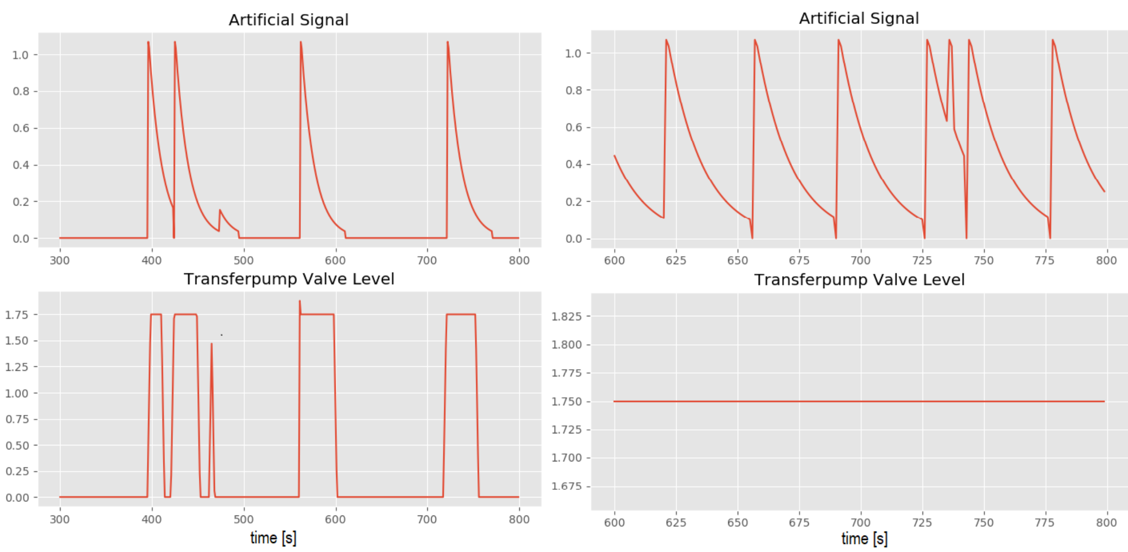

Figure 4.5: Illustration of the transfer pump valve level and the artificial signal before and after regions where the transfer pump is turned off was removed. As can be seen to the right, information is lost as the transfer pump valve level mostly is constant.

Another part of the signal pre-processing is to remove *outliers* of the measured transfer pump outlet pressure. The measured data outliers resulted from removing the zero regions of the data, as the pressure signal suffered from a stochastic behavior just as the pump turned on. This resulted in a set of data points that heavily deviated from the rest of the data just at the start of each active region of the transfer pump. An active region is referred to in this case as the region when the transfer pump valve level is non-zero. As these data points had a stochastic behavior a decision was made to remove these, resulting in the term filtering of *outliers*. The filtering is made in a way that removed the first and last three data points right as the pump is turned on, and the difference between the non-filtered and filtered outlet pressure signal is shown in Figure 4.6.

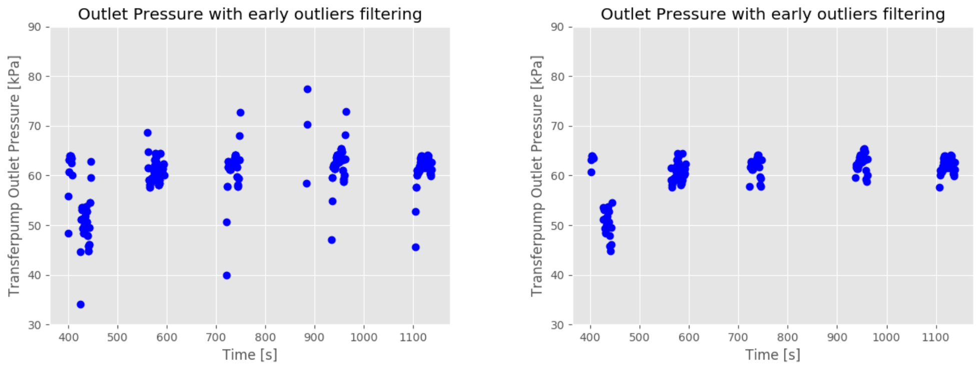

Figure 4.6: Transfer pump outlet pressure before (left) and after filtering of outliers. Clusters of the signal can be seen due to only considering data points when the transfer pump is active, and the regions between the clusters are from when the transfer pump is not active.

The effect of adding the artificial signal and the filtering of outliers made it easier for ML algorithms to predict the outlet pressure residual, and the resulting impact can be seen in Figure 4.7. The original setting predictions can be seen to the left, and the results of adding the artificial signal together with early outliers filtering can be seen to the right. Note that all the features listed in *List of ML features* were used when generating these predictions, that is the updated feature set. What can be seen in Figure 4.7 is that the spikes in the measured residual are removed, due to the filtering of outliers. In addition to this, $\hat{r}_{pr}(t)$ more accurately follows $r_{pr}(t)$ and the prediction is overall reduced meaning that ML algorithm predictions are more precise. This can also be seen in the numerical values of Table 4.3.

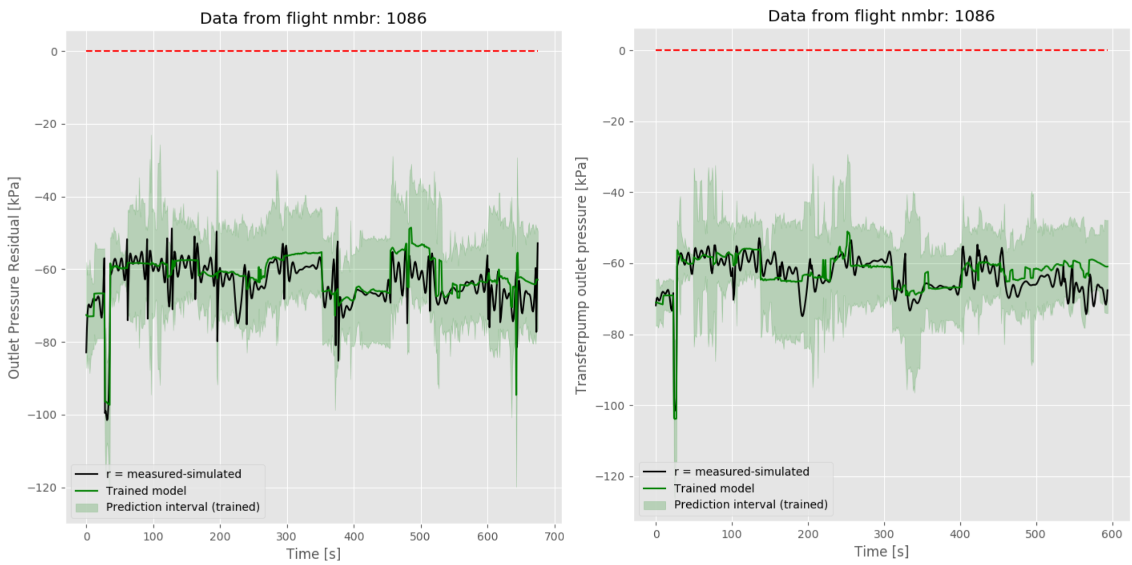

Figure 4.7: Predictions of the transfer pump outlet pressure residual with (right figure) and without (left figure) the addition of the artificial signal and filtering of early outliers. In black, $r_{pr}(t)$ and $\hat{r}_{pr}(t)$ in green. $I_{pr}(t)$ is seen in light green.

Note that both the effect of filtering outliers and adding the artificial signal generally improved the scores of both the predicted residual and the prediction interval. The effect seen from each procedure seems to contribute equal much, but only the total improvement is presented here. Thus, it is beneficial to include both procedures in the RM. In the same fashion as for the updated features, the same nine test flights are investigated and their corresponding *Average Predicted & Measured Residual* and *Average Prediction Interval* is again analyzed to evaluate how the resulting predictions and prediction intervals are affected by the changes made. The features used when generating this data are all the features listed in *List of ML Features*, presented earlier in this subsection. The flights presented here do not contain any anomalies and are accurately predicted as so. When looking at the numerical scores presented in Table 4.3, both the average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$ and the average $I_{pr}(t)$ are reduced in general. However, this mostly holds for flights that are of the kinder type. The last five flights seen in Table 4.3 where the pilot flew in a tougher manner are of mixed results, where flights 1081, 1086, and 1091 show a larger prediction interval. Flight 1091 also recorded larger deviations in the predicted residual. This shows vulnerabilities in the model when being presented with more sparse data, which could be a result of e.g. bad data balancing. This could also be improved by gathering more flight data of said situations. However, the scores seem to generally have increased when looking across all the flights, which can be seen as a model improvement.

Transfer Pump Outlet Pressure Data, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	No filt, No AR-sig	Filt, AR sig	No filt, No AR-sig	Filt, AR sig
1044	3.53	2.97	29.41	19.51
1047	3.53	2.75	26.49	18.67
1048	3.27	2.97	26.16	19.20
1052	3.55	3.16	25.90	20.69
1055	3.72	2.80	26.47	19.22
1081	5.95	5.88	32.76	36.79
1086	3.68	2.61	29.66	31.37
1088	5.35	4.04	30.95	25.01
1091	4.27	6.53	30.73	33.14

Table 4.3: Columns illustrating the measurement scores of the test flights when filtering out outliers and adding the artificial signal.

Implementation and Effect of Improved Dymola Model

This section presents the results of when the improved Dymola model is used to simulate the transfer pump outlet pressure and the effect that the improvements have on the ML algorithm predictions. The thought behind improving the DM is to facilitate ML predictions by creating a more smooth measured residual signal. As can be seen in the left plot of Figure 4.8, the simulated signal (in blue) is not very representative of the measured data (red). One factor that is believed to improve the anomaly detection and accuracy in the RM was to gain a more representative simulation signal.

Original Dymola Model: Transfer Pump Outlet Pressure

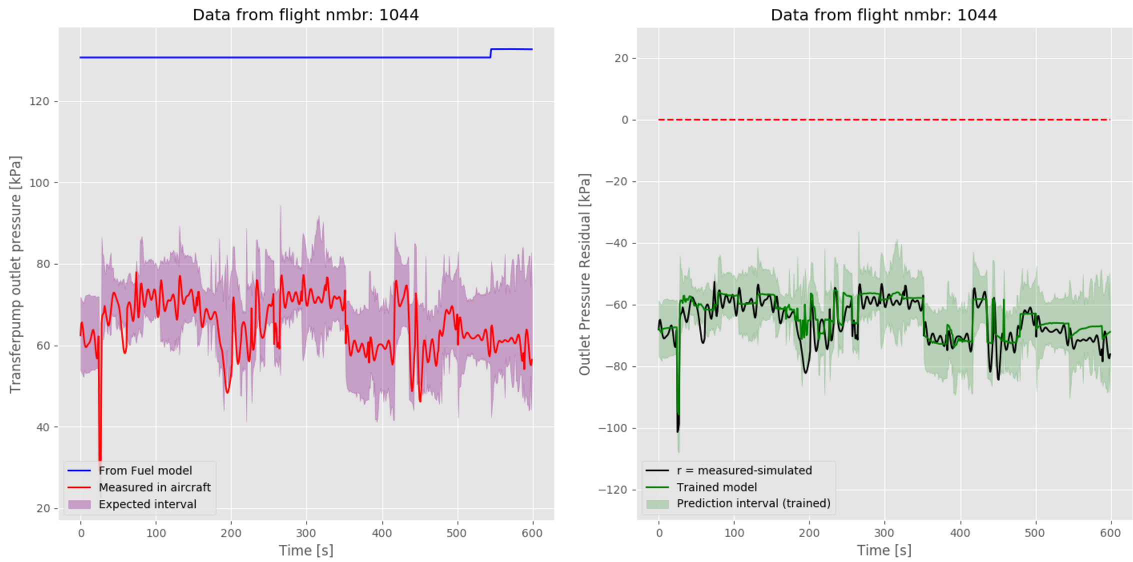

Figure 4.8: Illustration of the simulated data from the original DM to the left. The blue line showing the simulated data heavily deviates from the measured data in red. However, the ML algorithm predictions seen to the right in green are still accurate.

By achieving a more accurate simulation signal larger deviations in the residual can be avoided, resulting in easier signal behavior of the target signal outlet pressure residual. The process of improving the pressure sub-model of the DM is done by comparing the outlet pressure of features in the training set. Signal analysis between the outlet pressure and all the features used to select the training set revealed connections between stabilization levels in the outlet pressure and the tank valve commandos of different tanks. Generally, (4.1) describes the resulting output pressure.

$$f(x_1(t), x_2(t), x_3(t), x_4(t)) = y_{sim,pr}(t) \tag{4.1}$$

Where $x_1(t), x_2(t), x_3(t)$ and $x_4(t)$ represent control signals in the aircraft, in this case, the binary tank valve command levels of tanks 1,2 and 3, and the transfer pump valve command level. In Figure 4.9, two different stabilization levels in the pressure are seen at 300-400s depending on whether the tank valve command of tank 2 or tank 3 is active (binary value 1). In addition to this, a connection between the transfer pump valve command and the outlet pressure was found. It could be seen that a more active transfer pump, where the valve command level generally is higher results in higher outlet pressure. The transfer pump valve command signal together with the tank valve command levels of tanks 1,2 & 3 is then used to implement the improved DM. A basic structure of the resulting DM simulating the outlet pressure can be seen below.

- If the transfer pump valve command signal < 2.5, set output pressure to 65 kPa.
- If the transfer pump valve command signal > 2.5, set output pressure to 100 kPa.
- If tank 1 valve command is zero, decrease output pressure by 20 kPa.
- If tank 2 valve command is one, increase output pressure by 5 kPa.
- If tank 3 valve command is one, decrease output pressure by 5 kPa.

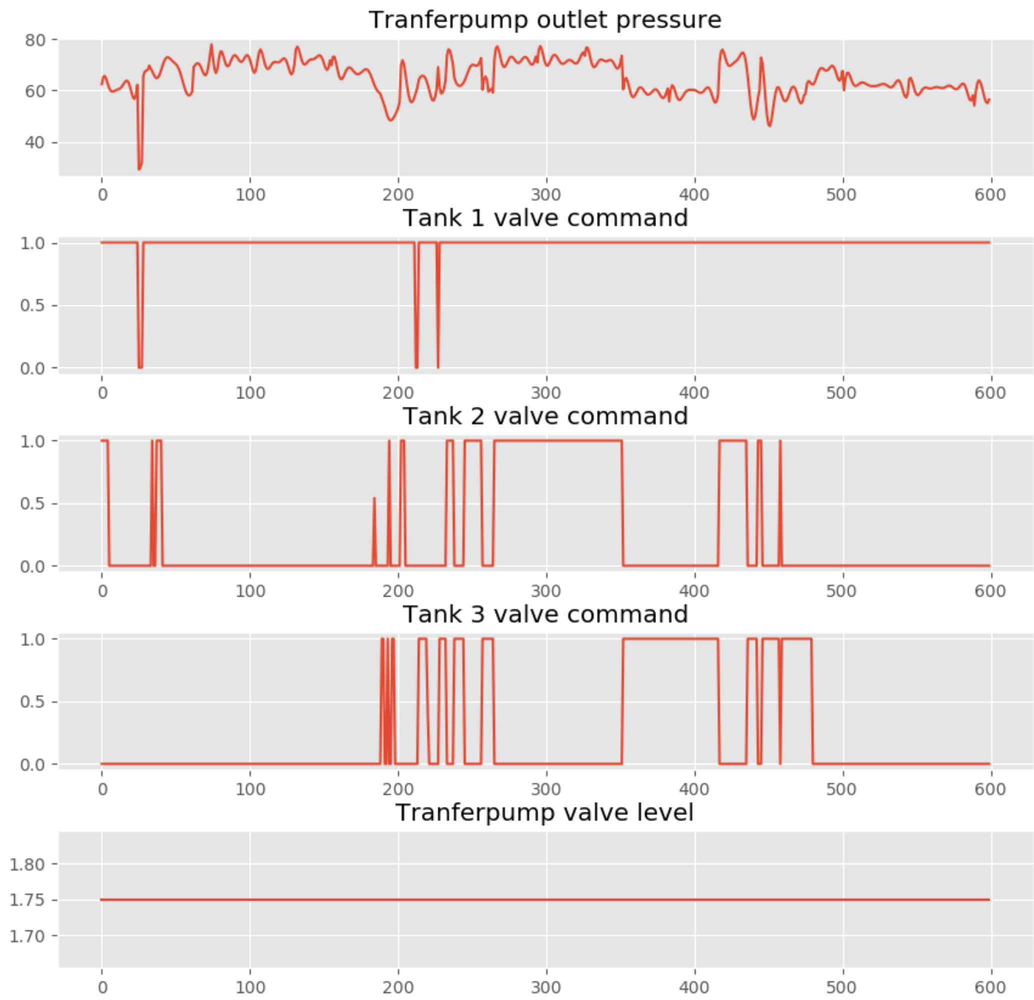

Figure 4.9: Illustration of how the transfer pump outlet pressure depends on which tank is being active. Note the different levels of pressure depending on whether tank 2 or tank 3 is being used, from 300-400s.

Figures showing data from the improved DM are seen below together with how the improvement affected ML algorithms. The features listed in *List of ML Features* presented earlier in this subchapter are used together with the added artificial signal. Filtering of outliers is also applied here. It can be seen that the ML algorithm predictions are accurate for both the improved and original DM settings and that ML algorithms perform well even when the simulation data does not accurately reflect measured flight data. This can be seen by comparing the right-hand figures between Figure 4.8 and Figure 4.10.

Improved Dymola Model: Transfer Pump Outlet Pressure

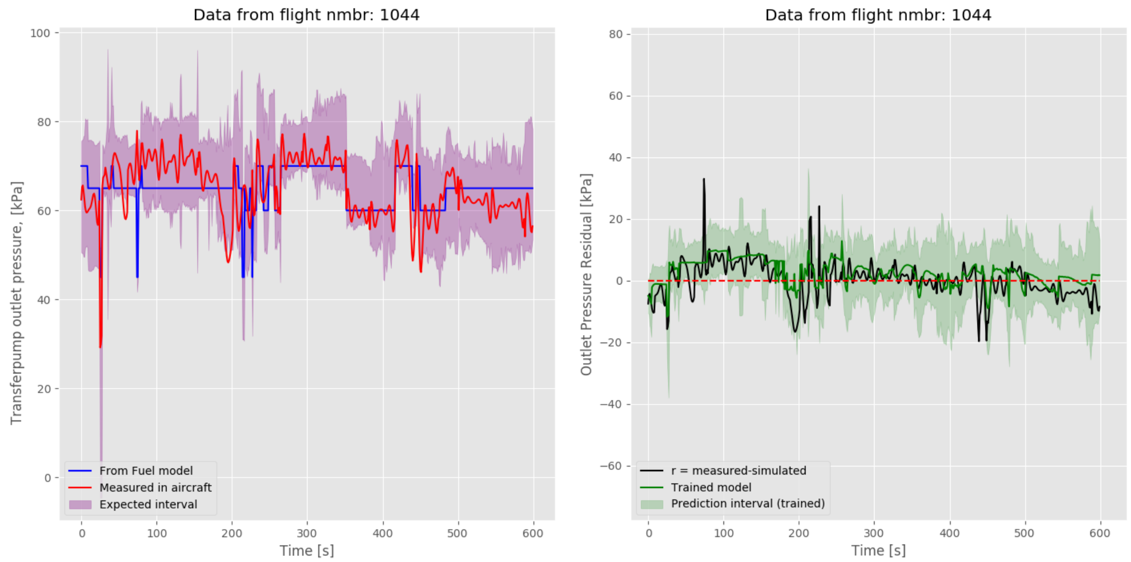

Figure 4.10: Data from the improved Dymola model. $y_{sim,pr}(t)$ (blue) and $y_{meas,pr}(t)$ (red) to the left. The measured outlet pressure residual $r_{pr}(t)$ (Black) together with the predicted outlet pressure residual $\hat{r}_{pr}(t)$ (Green). The prediction interval can be seen in light green.

This effect is also seen in the tougher flights, e.g flight 1091 as can be seen in Figure 4.12 and Figure 4.11. This suggests that emphasis, in this case, should be focused on improving ML algorithm conditions such as feature selection, filtering, and addition of artificial signals rather than improving the DM. Improving the DM does not facilitate anomaly detection either, as the ML algorithms heavily rely on the ML algorithm's performance.

Original Dymola Model: Transfer Pump Outlet Pressure

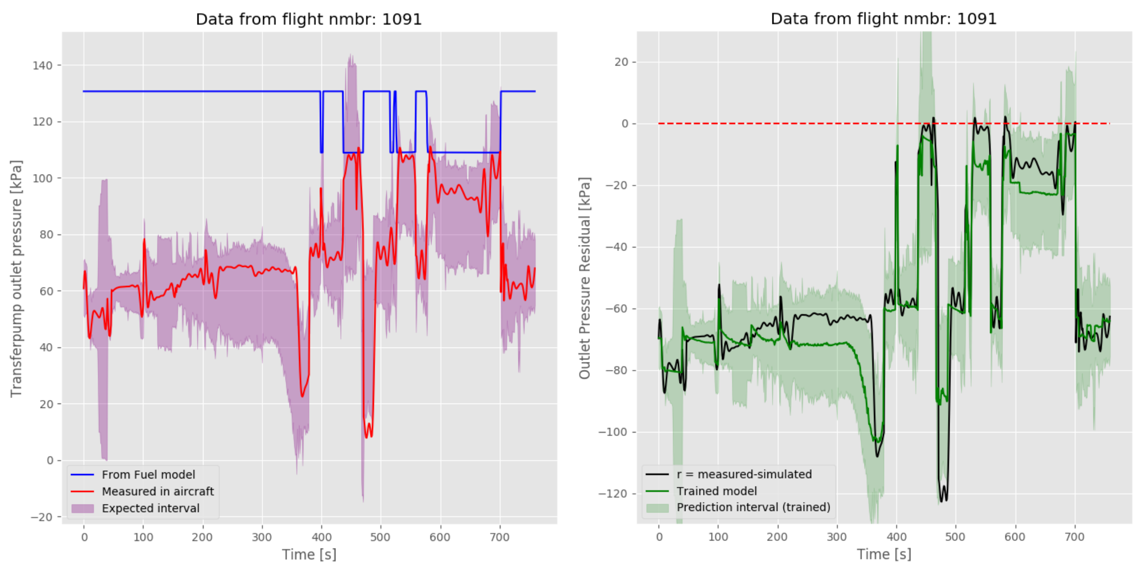

Figure 4.11: Data from the original Dymola model. $y_{sim,pr}(t)$ (blue) and $y_{meas,pr}(t)$ (red) to the left. The measured outlet pressure residual $r_{pr}(t)$ (Black) together with the predicted outlet pressure residual $\hat{r}_{pr}(t)$ (Green). The prediction interval can be seen in light green.

Improved Dymola Model: Transfer Pump Outlet Pressure

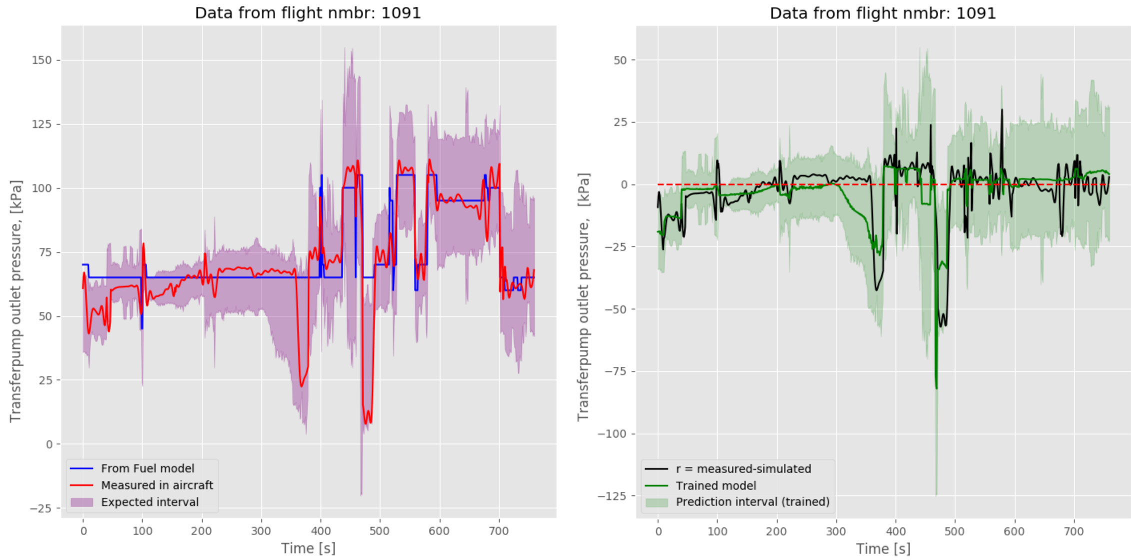

Figure 4.12: Data from the improved Dymola model. $y_{sim,pr}(t)$ (blue) and $y_{meas,pr}(t)$ (red) to the left, where the improved simulation signal can be seen. The measured outlet pressure residual $r_{pr}(t)$ (Black) together with the predicted outlet pressure residual $\hat{r}_{pr}(t)$ (Green). The prediction interval can be seen in light green.

Below the results are seen for the original and the improved Dymola model, with the features listed in *List of ML features* together with the artificial signal and filtering of early outliers. Results show that improving the Dymola model does not decrease the average prediction interval. Improving the Dymola model rather makes the average prediction interval slightly larger. The same trend is also seen for the average difference between the predicted and measured residuals, where the results indicate that the neural network has a harder time predicting the outlet pressure after the Dymola model is improved. Lastly, a table showing the difference between measured and simulated outlet pressure for both the original and improved Dymola model is presented, giving numerical values for the changes made.

Data of Original & Improved DM: Transfer Pump Outlet Pressure, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	Original DM	Improved DM	Original DM	Updated DM
1044	2.97	3.96	19.51	21.41
1047	2.75	3.48	18.67	20.78
1048	2.97	3.33	19.20	22.18
1052	3.16	3.73	20.69	21.55
1055	2.80	3.47	19.22	20.08
1081	5.88	5.47	36.79	34.07
1086	2.61	3.67	31.37	24.25
1088	4.04	4.34	25.01	26.07
1091	6.53	5.89	33.14	31.83

Table 4.4: Columns illustrating the measurement scores of the test flights when comparing data from the original and improved DMs.

Average $y_{meas,pr}(t)$ - $y_{sim,pr}(t)$: Tank 2, Flight 39-9		
Flight Nbr	Original Dymola Model	Improved Dymola Model
1044	66.32	4.16
1047	65.01	3.05
1048	65.91	3.62
1052	65.54	3.78
1055	63.71	3.63
1081	55.46	11.95
1086	63.85	4.10
1088	52.35	9.68
1091	54.93	6.95

Table 4.5: The average difference between simulated and measured flight data of the outlet pressure. The right columns gives a numerical value for the improved Dymola model. Figures of the improved dymola model in flights 1044 and 1091 can be seen in Figures 4.10 and Figure 4.12.

Anomaly Detection of Outlet Pressure

Here the results are presented showing how anomalies are detected in flights. To accurately detect anomalies, an algorithm is implemented to detect when the measured outlet pressure significantly deviated outside the prediction interval for an extended period of time. The algorithm uses an accumulated sum described more in detail in Section 2.5, and alarms for an anomaly when this accumulated deviation sum exceeds a set threshold. The algorithm correctly predicts the first nine test flights listed in Table 4.1 as non-anomalies, and the remaining two flights that contain anomalies are also accurately predicted as anomaly-containing flights. Note that thanks to the accumulated sum, temporary spikes in the measured data are allowed, not triggering the anomaly detection alarm. Figures of anomaly containing flights 1201 and 1175 are seen in Figure 4.13 and Figure 4.15 where the red area marks anomaly regions. The model settings used here are from the improved DM, with added artificial signal and the filtering of early outliers.

Anomaly Flight 1201: Transfer Pump Outlet Pressure

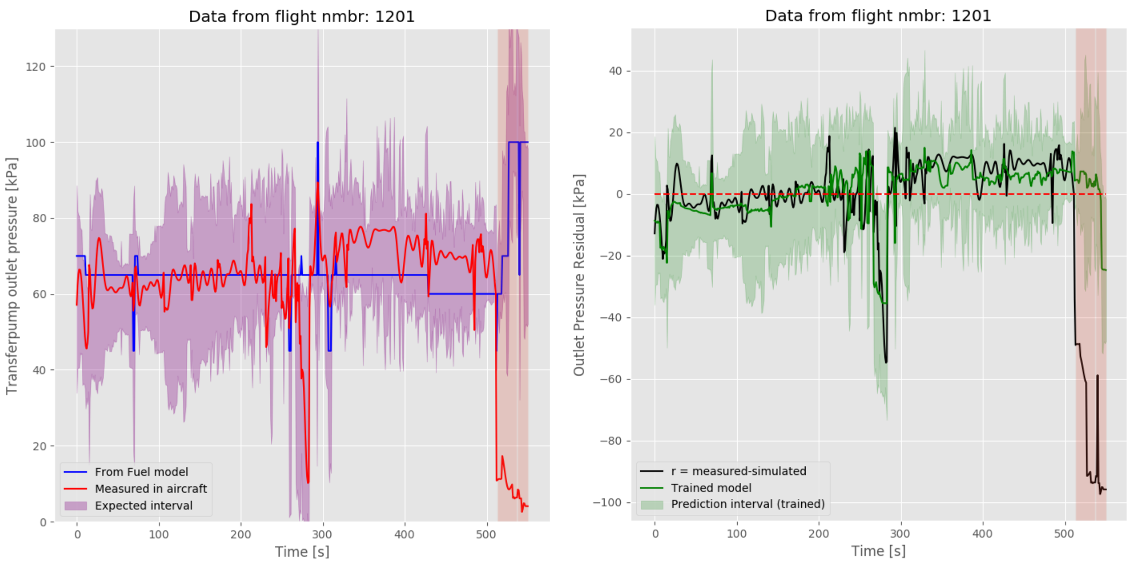

Figure 4.13: The anomaly detection algorithm clearly indicates where an anomaly is detected. Data is from a flight where the transfer pump broke mid-air, forcing the pilot to return to the base. Note that the prediction interval is fairly constant, making the large deviation in the measured residual easy for the anomaly algorithm to detect.

It is of interest to see how the prediction interval behaves around the anomaly regions in flights. Ideally, the prediction interval would be unchanged in the anomaly regions, clearly marking that the data deviate from normal patterns. This can be seen in Figure 4.14, where the prediction interval is relatively constant between 200-500s, but a large (non-anomaly) deviation is covered up and followed by ML algorithm predictions, as opposed to the anomaly right after 500s. Additionally, figures showcasing anomaly flight 1175 are seen in Figure 4.15, where a large deviation in the measured data is seen mid-flight. The reason for this anomaly is unfortunately not known.

Zoomed in Anomaly Flight 1201: Transfer Pump Outlet Pressure

Figure 4.14: This figure clearly shows the constant prediction interval, which follows the deviation in the measured residual right before 300s. However, the deviation right after 500s is not tracked and is thus classified as an anomaly.

Anomaly Flight 1175: Transfer Pump Outlet Pressure

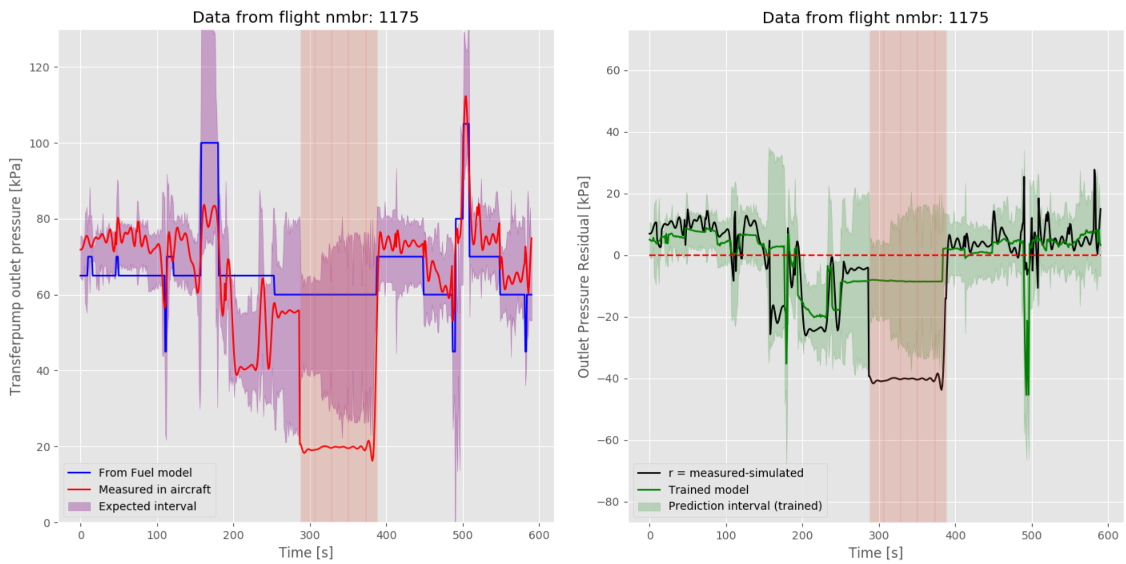

Figure 4.15: Another anomaly containing flight, where a large deviation is seen for an extended period of time. The reason for the anomaly is unknown. Although the prediction interval grows larger during the deviation, the anomaly is still detected with a good margin.

4.3 Results: Transfer Pump Outlet Flow

This section presents the results obtained when studying the outlet flow signal of the transfer pump. The main thought behind investigating the flow was not only to improve the RM for better anomaly detection but also to obtain an improved DM as the outlet flow DM is closely linked to the fuel mass DM. As the transfer pump outlet pressure and flow possess very similar properties, only the differing results obtained when studying the flow are presented. In short, the results of the outlet flow RM and the improved DM again show that the choice of features plays an important role in the ML algorithm's performance. As opposed to the outlet pressure, adding an artificial signal does not seem to improve the results. In a similar fashion to the outlet pressure, improving the DM of the flow does not seem to improve the overall ML algorithm performance.

List of Improvement Areas Studied

The areas investigated and improved follow the same pattern as the transfer pump outlet pressure, as the two signals are closely related and possess similar properties. What differs from the flow procedure as compared to the outlet pressure procedure is that no filtering of outliers is needed, as the flow does not suffer from the same stochastic behavior as the pressure. A more in-depth investigation of the features was also conducted, as the results from the outlet pressure showed the importance of using the right set of features. More focus was put into refining the accuracy in the DM, as the outlet flow model in DM is closely linked to the fuel mass DM. Thus, the main reason for improving the outlet flow DM was to obtain better simulation data of the fuel mass in tank 2, which is presented in the next sub-chapter. This was also a motivation to develop the most accurate flow model possible.

List of ML Features

Here, the results of using four different sets of features for the ML algorithms are presented. The performance when using the different settings can be seen in Table 4.6 and Table 4.7. Initially using features listed in feature set one, a few selected signals were chosen that were believed to have a connection to the outlet flow and thought to facilitate ML algorithm training. The selection is also based on linear correlation analysis between signals in the data set and the outlet flow, which gave a suggestion of which signals to select. More features were then added in feature set two, investigating the impact of adding the fuel consumption and aircraft velocity on the predicted residual and the prediction interval. Lastly, tank valve signals for specified tanks together with the transfer pump valve command signal were added. Results showed that slightly less accurate predictions were obtained when using feature set II, as compared to feature set I. Thus, the additional feature set IV shows the results of feature set III, without the fuel combustion and aircraft velocity. Analyzing different sets of features was of great use when looking into what signals ML algorithms can benefit from, making it possible to reveal signals which contained a lot of information about target signal behavior.

Feature Set I

- **Target signal - residual of outlet flow** $r_{fl}(t)$ [Kg/s] - Self explanatory. Included to train ML prediction output
- **Static surrounding pressure** [kPa] - Same as for outlet pressure.
- **Temperature of fuel to engine feed** [C] - Same as for outlet pressure.
- **Vertical flying angle** [Degrees] - Same as for outlet pressure
- **Vertical acceleration** [m/s^2] - Same as for outlet pressure

- **Volume of fuel tank 2** [L] - Fuel volumes are closely linked to the flow, as drastic changes in the fuel volume can indicate deviations in the flow. E.g. if the volume in tank 2 decreases rapidly, this could motivate a spike in the outlet flow.
- **Volume of fuel tank 3** [L] - Same as for volume in fuel tank 2.
- **Measured fuel mass in drop tank** [Kg] - Same as for volume in fuel tank 2.

Feature Set II

- *Plus features listed in feature set I*
- **Fuel consumption** [Kg/s] - It is believed that the fuel consumption should be closely linked to the fuel flow through the transfer pump, as the transfer pump supplies the engine with fuel.
- **Aircraft velocity** [Mach] - Greater aircraft velocities creates a larger engine need for fuel, and could thus be linked to an increase in the outlet flow, as the outlet flow supplies the engine with fuel.

Feature Set III

- *Plus features listed in feature set I and II*
- **Tranferpump valve command signal** [4 different levels] - The increased activity of the transfer pump should result in a larger outlet flow.
- **Tank Valve command signal for tanks 1,2,3,6, Right/Left-wing and drop tank** [On/off] - As the outlet pressure residual predictions improved a lot by including these signals, it is thought to help here as well due to the similarities of the flow and pressure.

Feature Set IV

- *Plus features listed in feature set I*
- **Tranferpump valve command signal** [4 different levels] - See feature set III.
- **Tank Valve command signal for tanks 1,2,3,6, Right/Left-wing and drop tank** [On/off] - See feature set III.

Numerical Results for Different Sets of Features

Results show that feature set three perform the best in regards to making predictions more accurate and shrinking the prediction interval, indicating that a lot of information about the flow can be gained by including these features when training ML algorithms. Only a slight score reduction can be seen across most flights when using feature set II, as compared to feature set I, with the exception of flight 1086 where the average prediction interval more than doubled. This suggests that the fuel consumption and aircraft velocity present information that makes it tougher for ML algorithms, which is an interesting result. In addition to this, flight 1086 can contain interesting information on the velocity and fuel consumption, and highlight data patterns that are rare or deviate from normal signal behavior. What is clear is that flight 1086 contains information on the fuel consumption or velocity that more than doubles the prediction interval. Another interesting detail is that feature set IV generally only performs slightly worse than feature set III, and the difference of 1086 between the two feature sets is not as large as between feature set I and II. Although ML algorithms with feature set IV perform well, feature set III seem to perform slightly better in general and are used in the next steps. This can also be seen in Figure 4.16, illustrating the results of the four different feature sets for flight 1088.

Outlet Flow: Average $I_{fl}(t)$, Flight 39-9				
Flight Nbr	Feature Set I	Feature Set II	Feature Set III	Feature Set IV
1044	0.932	1.133	0.534	0.602
1047	1.228	1.233	0.530	0.485
1048	0.905	1.031	0.538	0.691
1052	0.721	0.882	0.508	0.510
1055	0.686	0.740	0.448	0.472
1081	1.139	1.189	0.794	0.848
1086	0.884	1.747	0.675	0.771
1088	1.075	1.087	0.723	0.686
1091	0.793	1.138	0.702	0.801

Table 4.6: Results showing the average prediction interval $I_{fl}(t)$ for four different sets of features used when training the ML algorithms in the transfer pump outlet flow RM.

Interestingly enough, when looking at Table 4.7 the numerical differences between feature set I and II are small. Most predictions are even slightly better for feature set II. In regards to flight 1086, no large difference in scores can be seen there, which is a bit confusing. The scores between feature set III and IV are also very similar, but both scores outperform the ones of feature set I and II by far. As the average prediction interval is slightly smaller for feature set III, it can still be seen as the most favorable feature set even though the similar results between feature set III and IV here.

Outlet Flow: Average $r_{fl}(t)$ - $\hat{r}_{fl}(t)$: Flight 39-9				
Flight Nbr	Feature Set I	Feature Set II	Feature Set III	Feature Set IV
1044	0.147	0.146	0.079	0.074
1047	0.109	0.095	0.054	0.052
1048	0.144	0.131	0.063	0.074
1052	0.089	0.073	0.054	0.054
1055	0.115	0.086	0.054	0.052
1081	0.316	0.337	0.198	0.199
1086	0.169	0.162	0.082	0.083
1088	0.210	0.223	0.111	0.124
1091	0.170	0.228	0.125	0.126

Table 4.7: Results showing the average residual difference for four different sets of features used when training ML algorithms of the transfer pump outlet flow RM.

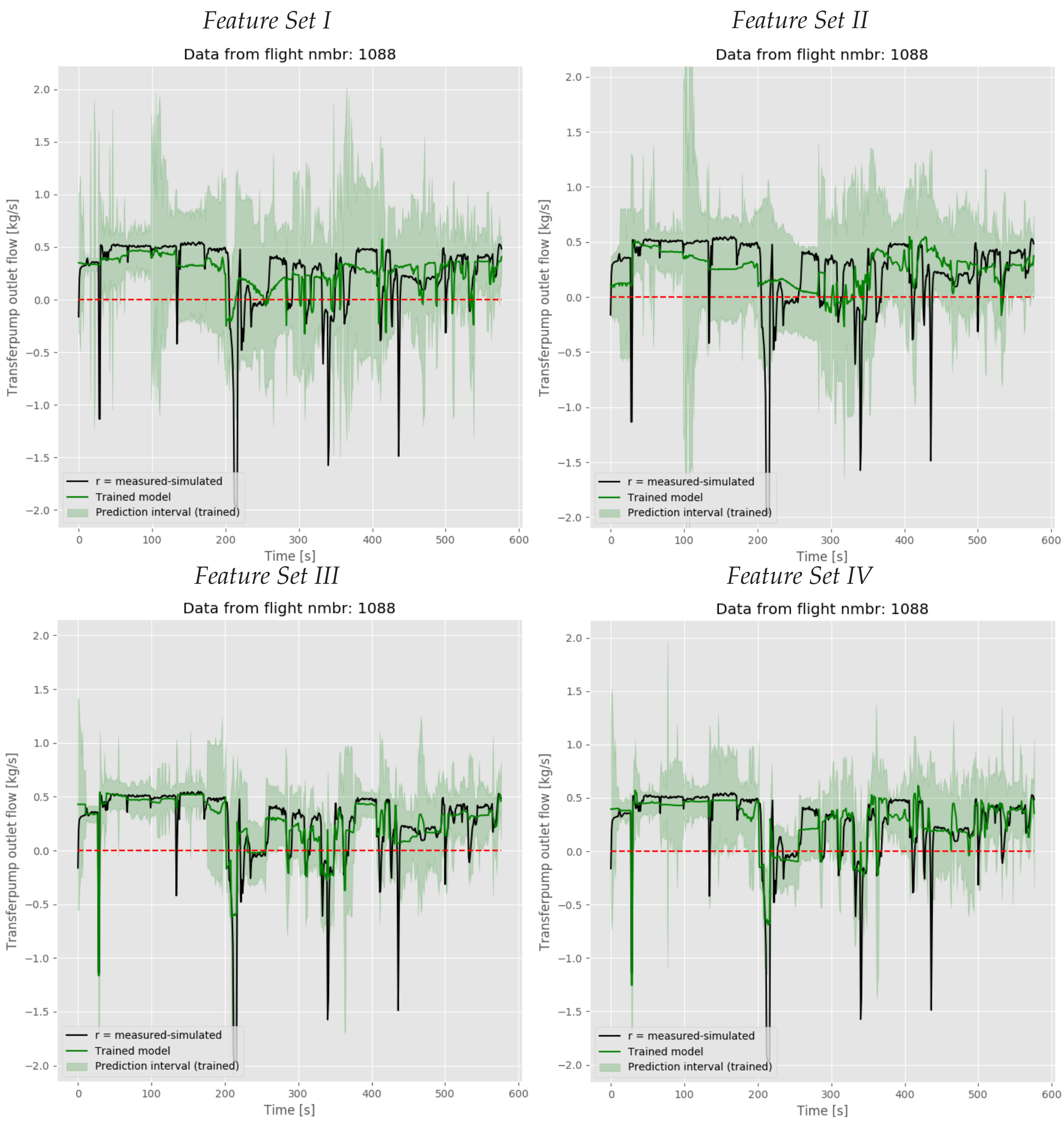

Figure 4.16: Illustration of how the different feature sets affect predictions of the residual and the prediction interval.

Addition of Artificial Signal

This section presents the results of adding an artificial signal to the data set, thought to help ML algorithm predictions. Feature set III is used when generating these predictions. The same artificial signal shown in Figure 4.5 is added, intending to show the dynamic behavior of the transfer pump, indicating when the pump is active and not. This information is lost when the data of the non-active regions of the transfer pump is removed. However, results show that adding the artificial signal does not help ML algorithms. Neither the prediction interval nor the difference between predicted and measured residual changes significantly, indicating that the artificial signal does not supply any additional information. This can be seen as a general trend in all the listed test flights of Table 4.8.

Addition of Artificial Signal: Transfer Pump Outlet Flow, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	No AR-signal	With AR-signal	No AR-signal	With AR-signal
1044	0.079	0.084	0.534	0.542
1047	0.054	0.057	0.530	0.458
1048	0.063	0.077	0.538	0.651
1052	0.054	0.057	0.508	0.486
1055	0.054	0.046	0.448	0.438
1081	0.198	0.187	0.794	0.766
1086	0.082	0.076	0.675	0.700
1088	0.111	0.118	0.723	0.597
1091	0.125	0.114	0.702	0.707

Table 4.8: The resulting scores for each of the nine test flights, showing the effect of adding an artificial signal to the data. Numerical values show that the prediction accuracy does not change significantly, meaning that the artificial signal does not contribute any additional information to ML algorithms.

With and without the addition of Artificial Signal: Transfer Pump Outlet Flow

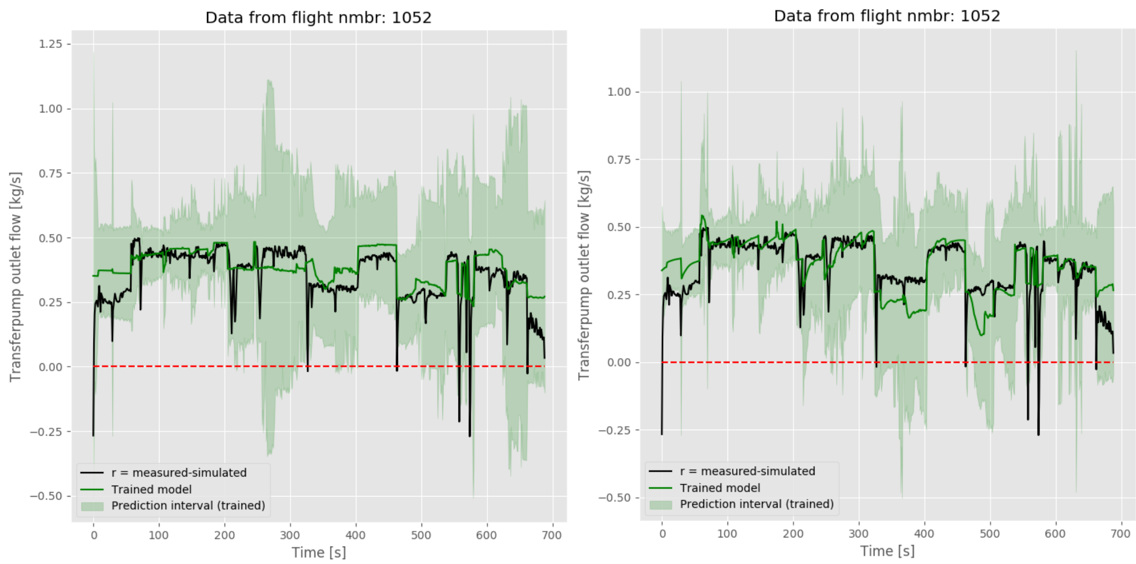

Figure 4.17: Results without artificial signal added to the left, and results with an added artificial signal to the right. No significant difference in the prediction accuracy can be seen.

Implementation and Effect of Improved Dymola Model

Here, figures and tables showing the effect of improving the Dymola model simulating the transfer pump outlet flow are shown. The total improved flow DM can be seen in Figure 4.18, and shows the overall structure of how the control signals are connected to produce the outlet flow. Results from these simulations are done with features listed in feature set three, and without the artificial signal. Results show that improving the Dymola model simulating the transfer pump outlet flow does not facilitate ML algorithms of the outlet flow RM, but it does give a much more accurate simulation signal which can be beneficial in other contexts.

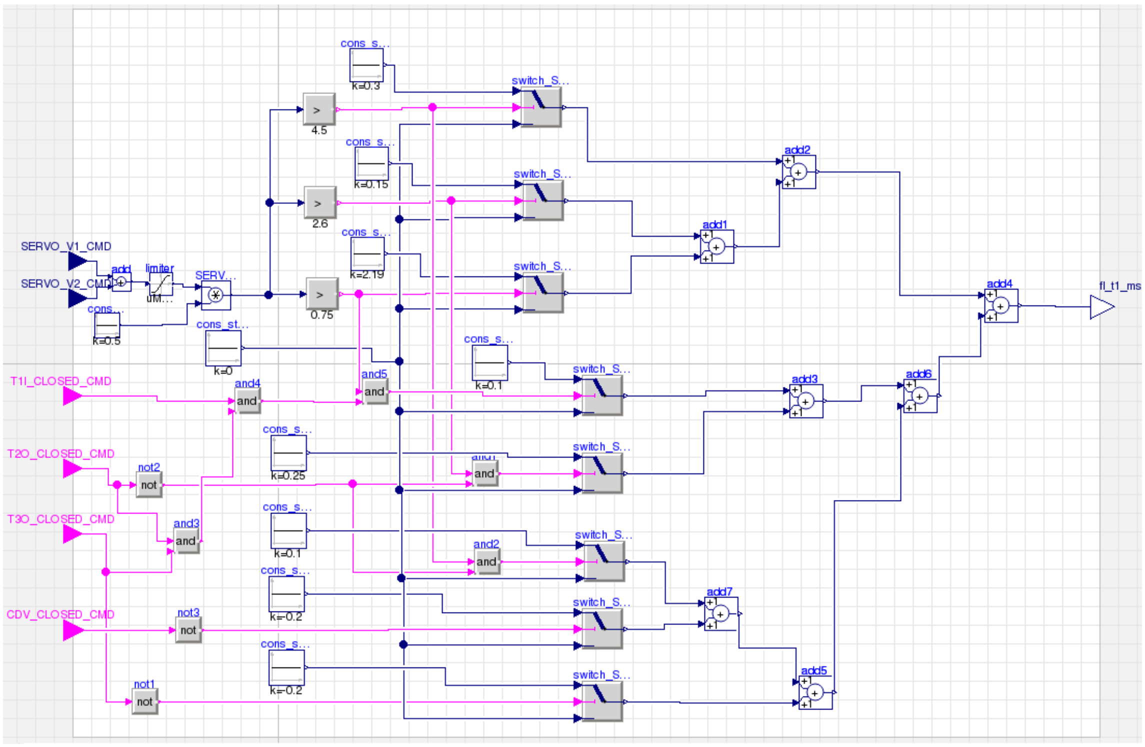

Figure 4.18: An overview of the implemented and improved flow DM. Control signals of tank valve command levels of tanks 1,2,3 and drop tank together with the transfer pump valve level (upper left) can be seen to the left. Binary signals can be seen in pink, and numerical signals in blue.

Accuracy scores of the prediction interval and predicted residual flow are shown in Table 4.9. In regards of the average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$, scores are mostly unchanged with the exception of flight 1086 and 1088. Flight 1088 shows a reduction of 37% in the average difference between measured and predicted residuals, which is a great improvement compared to other presented test flight results. Flight 1088 shows a reduction of 14%, which also suggests that improving the DM might help the accuracy of $\hat{r}_{pr}(t)$ output. In regards to the average prediction interval, the results are mixed. Flights 1044 and 1088 show a better score, meanwhile, flight 1052 shows an increase of 41% in the average prediction interval. It is hard to decide whether improving the DM improves the results, as various flights with different types of flight data perform both better and worse. In this case, it might be worth considering other beneficial aspects of improving the DM when analyzing the scores.

Effect of Improving Dymola Model: Transfer Pump Outlet Flow, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	Original DM	Improved DM	Original DM	Improved DM
1044	0.079	0.075	0.534	0.509
1047	0.054	0.052	0.530	0.538
1048	0.063	0.055	0.538	0.608
1052	0.054	0.049	0.508	0.720
1055	0.054	0.048	0.448	0.446
1081	0.198	0.191	0.796	0.859
1086	0.082	0.052	0.675	0.747
1088	0.111	0.096	0.723	0.605
1091	0.125	0.119	0.702	0.743

Table 4.9: Resulting scores when using simulated data from the improved flow DM, compared to the original DM. As can be seen from the columns, the average prediction interval and difference between $\hat{r}_{pr}(t)$ and $r_{pr}(t)$ are mostly unchanged even after implementing the improved DM.

Table 4.10 shows how much the improved DM reduces the average difference between $y_{meas,fl}(t)$ and $y_{sim,fl}(t)$. The first five flights show a great reduction, meanwhile, the last four flights are improved but not as much. However, the scores are generally much better when using the improved DM.

Average $y_{meas,fl}(t)$ - $y_{sim,fl}(t)$: Transfer Pump Outlet Flow, Flight 39-9		
Flight Nbr	Original Dymola Model	Improved Dymola Model
1044	0.370	0.085
1047	0.372	0.060
1048	0.349	0.061
1052	0.358	0.067
1055	0.388	0.060
1081	0.399	0.249
1086	0.357	0.076
1088	0.358	0.105
1091	0.281	0.137

Table 4.10: The average difference between simulated and measured flight data of the outlet pressure. The right columns gives a numerical value for the improved Dymola model.

Figure 4.19 and Figure 4.20 show how the improved DM affects the ML algorithm prediction accuracy. As can be seen in the figures and from Table 4.9, ML algorithms perform very well even with the original DM simulated flow, although a small improvement in the scores can be seen from Table 4.9.

Original Dymola Model: Transfer Pump Outlet Flow

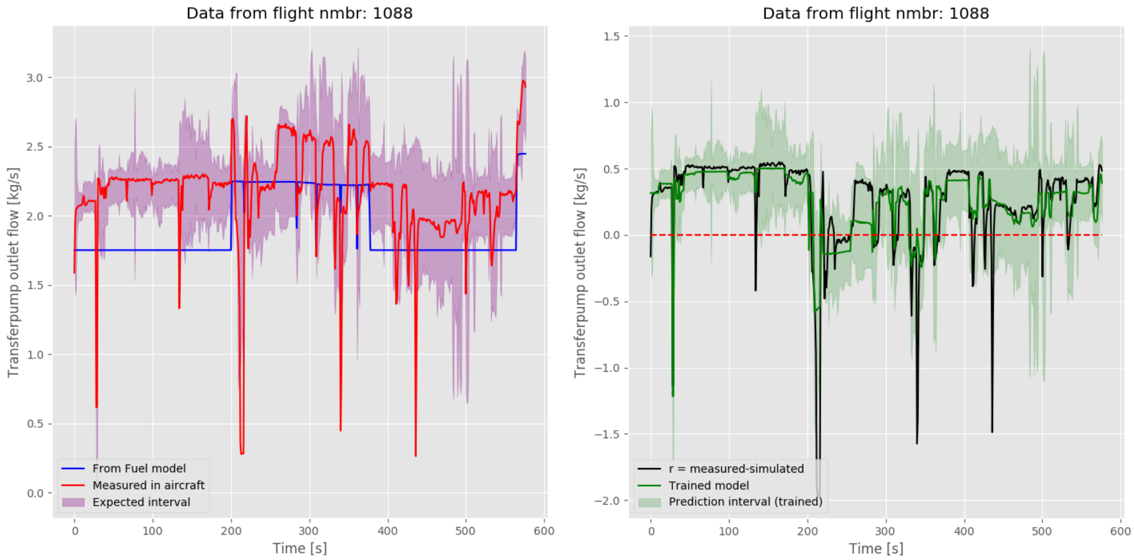

Figure 4.19: Simulated outlet flow $y_{fl,sim}(t)$ (red) and measured outlet flow $y_{meas,fl}(t)$ from the original DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Improved Dymola Model: Transfer Pump Outlet Flow

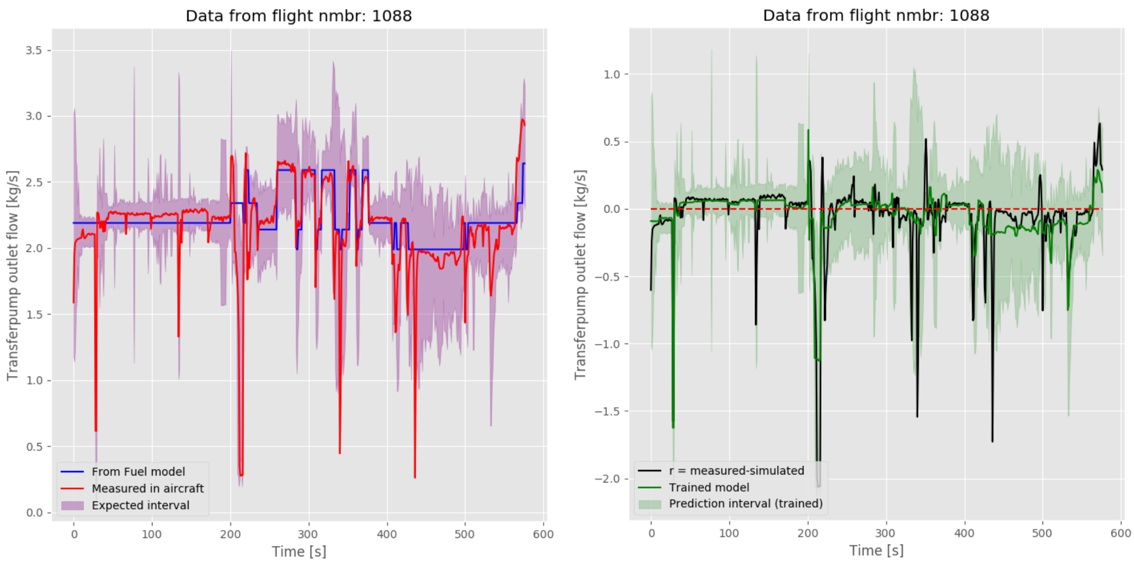

Figure 4.20: Simulated outlet flow $y_{fl,sim}(t)$ (red) and measured outlet flow $y_{meas,fl}(t)$ from the improved DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Anomaly Detection of Transfer Pump Outlet Flow

This subsection shows results from the anomaly detection algorithm. Figures from two flights containing anomalies are presented, when simulating data using feature set three and the improved Dymola model. Note that data at 400s in flight 1081 seen in Figures 4.22 and Figure 4.23 does classify as an anomaly for the outlet flow and fuel mass, but not in the transfer pump outlet pressure. A large spike in the flow can be seen, where a flow of 8 kg/s is measured. This is an anomaly as the flow can not reach such levels, indicating that there might be something wrong with the flow measuring equipment. The additional anomaly flight 1175 can be seen in Figure 4.21, where something happens in the aircraft between 300-400s.

Anomaly Flight: Transfer Pump Outlet Flow

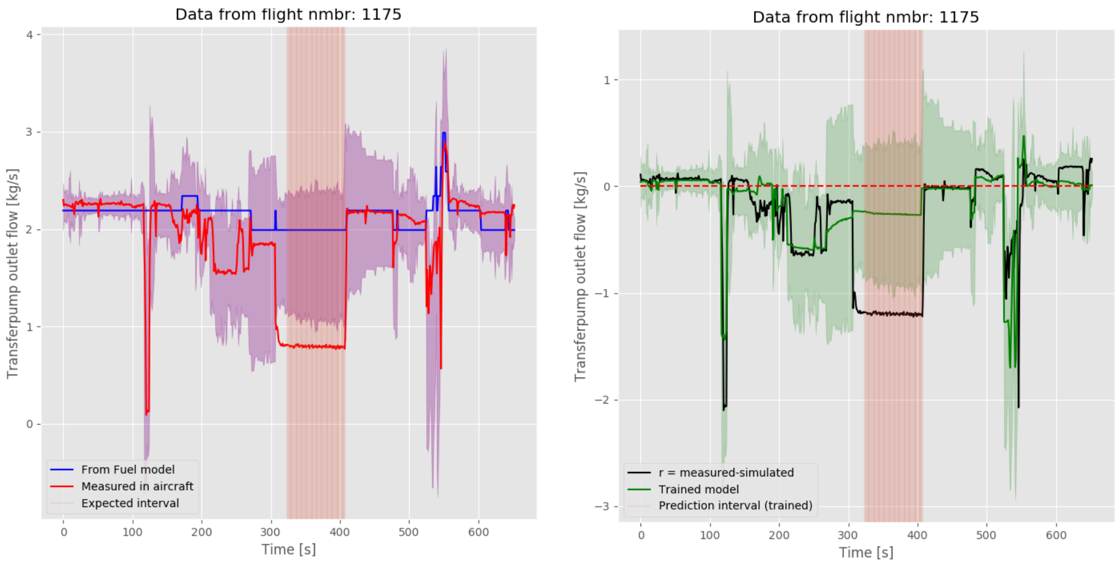

Figure 4.21: Data from an anomaly during flight 1175, where the measured data clearly deviates from the prediction interval during period 320-400s. The reason for the anomaly is unknown.

Anomaly Flight: Transfer Pump Outlet Flow

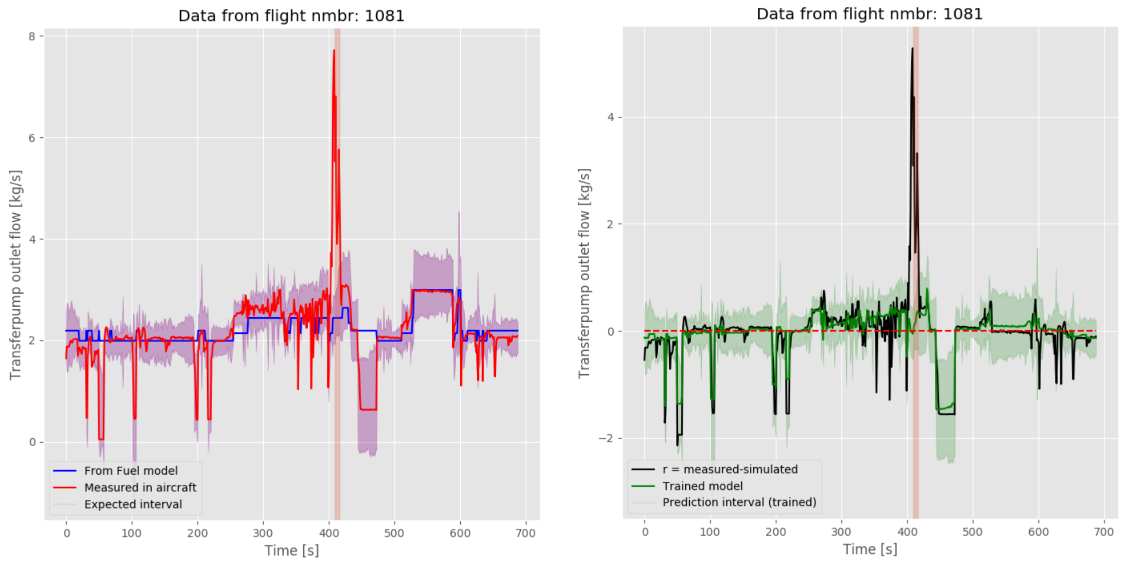

Figure 4.22: Data from anomaly containing flight 1081, where a significant deviation in the flow appears at 400s. The red area marks the anomaly alarm.

Zoomed in Anomaly Flight: Transfer Pump Outlet Flow

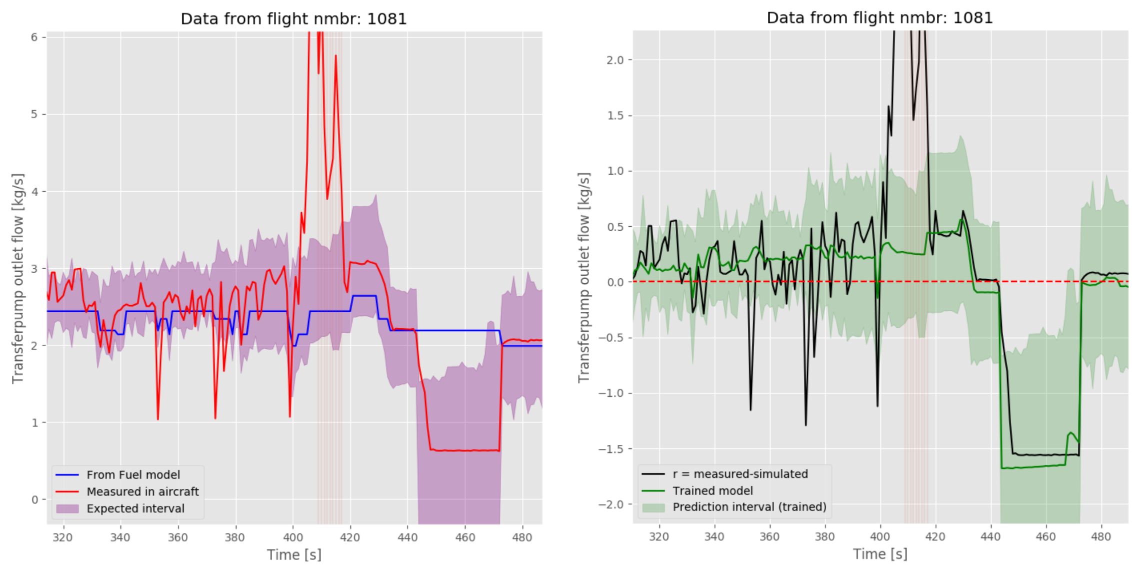

Figure 4.23: Zoomed in figures of the anomaly in flight 1081. Note that the small spikes around 340s are not classified as anomalies, but the larger, more extended spike at 400s is classified as an anomaly.

4.4 Results: Fuel Mass Tank 2

This section presents the results achieved when analyzing the fuel mass of tank 2, and when investigating the fuel mass residual $r_{fm}(t)$ to develop the corresponding RM. Properties in the fuel mass such as the accumulated effects of the signal make up vastly different signal behavior. The consequences of these accumulated effects on the ML algorithms are of extra interest to investigate and gain information about, as these properties are not present in the transfer pump outlet flow and pressure. Difficulties in modeling these effects in DM are also presented. The DM simulating the fuel masses is closely linked to the DM which simulates the transfer pump outlet flow presented in Section 4.3, and the improvement of the fuel mass DM model can be seen as a by-product of the improved outlet flow DM. However, the results of using an improved fuel mass DM simulating a more accurate signal of the fuel mass in tank 2 are presented in the following results. Results show that the tank 2 fuel mass RM is highly sensitive to the choice of features used and that they have to be chosen with great thought to achieve a good quality in ML predictions. In addition to this, data balancing has a significant impact on the ML algorithms and creates an unwanted stochastic behavior that was removed in this project. Lastly, results from the improved fuel mass DM show mixed results, where a slight improvement can be seen in a reduced average difference between $\hat{r}_{fm}(t)$ and $r_{fm}(t)$. On the other hand, the average prediction interval is generally unaffected by the improved DM simulated fuel mass.

List of Improvement Areas Studied

The areas of improvement follow the general list presented in Section 4.1, with the addition of two steps that are thought to take care of aggravating behavior in the model. First, a stochastic behavior in the RM is seen mainly due to the data balancing. Efforts were made to remove this stochastic effect so that the DM model improvements and effects of other changes to the RM could be investigated without random results. In addition to this, the previous RM only LP-filtered the measured fuel mass. LP-filtering comes with a shift in data, which is thought to complicate ML algorithm training. This is also looked into. An additional feature evaluation is also made, as the fuel mass signal suffers from accumulated effects making it difficult for ML algorithms to make accurate predictions.

- Additional feature selection investigation.
- Investigation of stochastic behaviour in data balancing - removing stochastic behaviour for improvement analysis.
- Investigation of effects in displacement due to Low-pass filtering of target signal.

Choice and Effect of ML Features

The features used to train ML algorithms of the tank 2 fuel mass RM are listed below. As opposed to the features used when training the transfer pump outlet pressure and flow, these features are chosen with extra caution as the fuel mass RM showed to be very sensitive to which features were included. One important aspect when choosing features are the fact that the target signal - fuel mass of tank 2 is an accumulated signal possessing different properties than the outlet pressure and flow.

- **Target signal - residual of fuel mass in tank 2** $r_{fm}(t)$ [kg] - Self explanatory. Target signal needed to train ML algorithm predictions.
- **Residual of fuel mass in tank 3 and 6** [kg] - Residual of fuel levels of closely connected tanks were thought to reveal general data deviations in the fuel mass, that could be traced back to the tank 2 fuel mass residual.

- **Temperature in engine feed tank** [C] - Higher temperatures have an effect on material properties, which could affect the general fuel system behaviour.
- **Temperature tank 1 (collector tank)** [C] - Higher temperatures have an effect on material properties, which could affect the general fuel system behaviour.
- **Vertical flying angle** [Degrees] - Different flying angles could reveal measurement errors in the aircraft tanks, that could be learned by ML algortihm training.
- **Horizontal flying angle** [Degrees] - Different flying angles could reveal measurement errors in the aircraft tanks, that could be learned by ML algortihm training.
- **Artificial signals from previous work** - Three signals representing accumulated errors that build up due to faulty measurements in the in- and out flow. The signals are built on a structured model for the flow as a function of the pressure, allowing the accumulated effects to be modelled when the corresponding tank valve is open. The thought behind this pressure and flow connection is inspired by Bernoullis equation, and was developed by Saab. Thus, the signals wont be discussed more in-depth as the authors of this thesis are not responsible.
- **Command signals to start jet pumps from tanks 2,3 and 6** [Binary] - The jet pumps can drain fuel even when the transfer pump is not active, which affects the fuel mass in all the tanks. If the command signal to start a jet pump is active, the fuel mass is reduced which is a pattern ML algorithms could learn.

More features were also tested, but with little success. Including a larger set of features oftentimes resulted in worse predictions, and many flights were misclassified as anomalies. All listed features from the transfer pump outlet flow and pressure were tested. Including a great portion of features such as fuel consumption and flight velocity resulted in large fluctuation in the prediction interval, causing a lot of flights to be misclassified as anomalies.

Illustration of Bad Feature Selection: Fuel Mass Tank 2

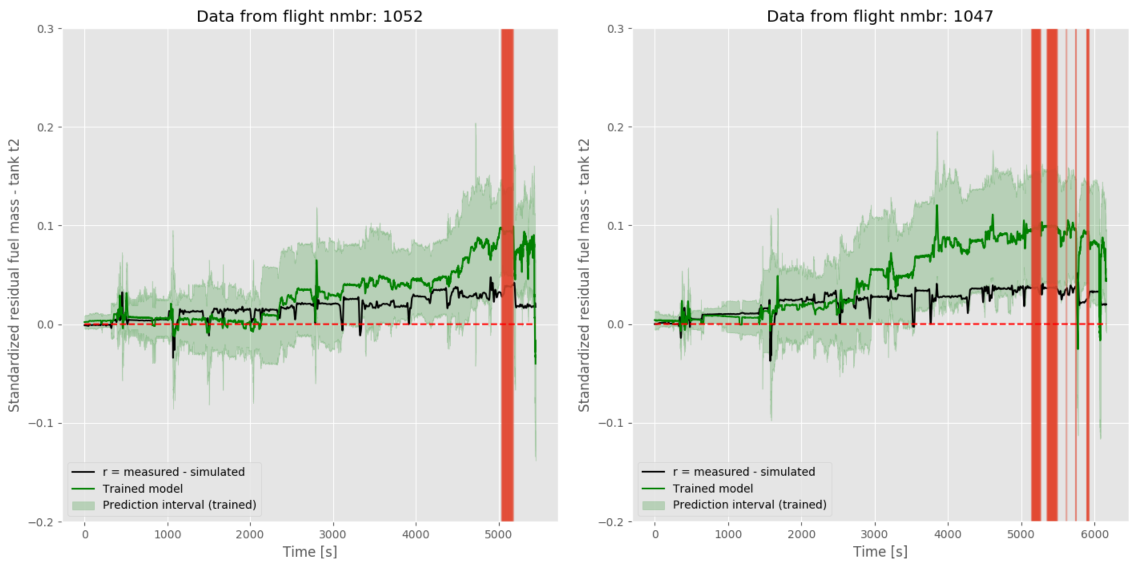

Figure 4.24: Illustration of two different flights, where the transfer pump valve level, tank valve commands of tanks 1,2,3, 6 and fuel combustion as features resulted in worse predictions, misclassifying flights as anomalies.

In addition to this, using the transfer pump valve level and tank valve commands of tanks 1,2,3, and 6 as features resulted in worse predictions where again, misclassification of anomalies was experienced. This is interesting as these features helped ML algorithm accuracy of the transfer pump outlet pressure and flow. The listed artificial signal included in the feature set was implemented in a previous master thesis and greatly benefits the ML algorithm predictions. In this thesis another signal for covering accumulated effect was created. This signal could be created since this data set now contained measurements from the jet pumps of the tanks. However, adding a signal to track accumulated errors from the jet pumps proved to have varying effect and not be a consistent improvment.

Low-Pass Filtering of Simulated Fuel Mass

One investigated procedure in the fuel mass RM is that only the simulated fuel mass previously is low-pass filtered. This is suspected to skew prediction accuracy as the low-pass filtering comes with a data-shift in the filtered signal, that creates a time gap between the measured and simulated fuel mass. The data shift that comes with signal filtering is thought to obstruct the ML algorithm accuracy, and the results of low-pass filtering both the measured and simulated fuel mass are seen in Table 4.11. In addition to this, the removal of spikes in the simulated data due to the low-pass filtering is also thought to facilitate ML algorithm training.

Effect of Low-Pass Filtering: Fuel Mass Tank 2, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	Non. Filt	Filtered	Non. Filt	Filtered
1044	6.92	6.45	59.37	59.81
1047	8.43	8.01	48.64	37.98
1048	9.05	8.70	57.52	58.75
1052	3.66	3.90	41.88	33.01
1055	15.52	14.41	60.87	62.03
1081	13.96	14.03	68.88	56.93
1086	8.59	10.14	58.51	52.45
1088	8.00	8.09	56.57	51.33
1091	11.36	11.11	50.39	49.57

Table 4.11: Numerical scores showing the results for the test flights, with and without low-pass filtering of the simulated fuel masses for tank 2,3, 6 and wing tanks.

Looking at Table 4.11, no general results can be drawn from the numerical scores. The results are mixed. While some flights seem to benefit from the filtering such as flights 1047 and 1052 where the average prediction interval is reduced, flights 1048 and 1055 record an increased prediction interval although not as large. The average $r_{pr}(t)$ and $\hat{r}_{pr}(t)$ difference is mostly unchanged for all test flights. However, as the data shift that comes with low-pass filtering is a fact, the low-pass filtering is included in the RM.

Data Balancing and Removal of Stochastic Behaviour

One flaw detected when analyzing the tank 2 fuel mass in the RM was a stochastic effect that came as a consequence of the data balancing, where the balancing process of picking data points from the different bins included a random choice of data points in the bins. As the ML algorithms are very sensitive to deviations in the training data, a consequence of the stochastic data balancing is that the predictions differed significantly each time the procedure was redone. Below, three different figures showing $\hat{r}_{fm}(t)$ and $r_{fm}(t)$ for the same fuel mass RM settings can be seen. The predictions and the prediction interval greatly differ each time, where only one of the flights is correctly classified as a non-anomaly containing flight. The stochastic effect of the data balancing made it difficult to analyze results from different model settings. This is resolved by saving the indices of one data balancing, thus freezing one data balancing procedure. The same indices representing the balanced data set are then used again when re-running the notebook.

Fuel Mass Tank 2: Three different Data Balancing Versions

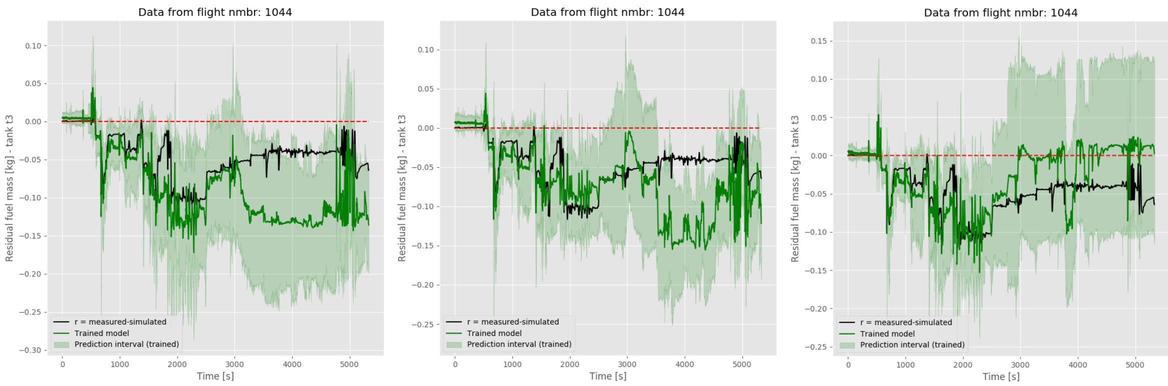

Figure 4.25: Three different prediction outputs of the same flight when re-runnig the fuel mass RM with the same settings. This shows the stochastic behaviour of the data balancing, making it tough to analyze different model settings, something that is crucial when developing the fuel mass RM.

This effect also reveals the importance of balancing the data correctly, and that a faulty balancing can damage model accuracy and skew information gained about the fuel system. Not having accurate data balancing can also make ML algorithms misclassify anomalies in flights. This is resolved by re-doing the data balancing procedure until a desirable distribution is gained. Not the best solution, but further improving the data balancing is out of scope for this thesis.

Implementation and Effect of Improved Dymola Model

A central area of investigation is the improvement of the fuel mass DM generating $y_{sim,fm}(t)$, and how a more accurate simulation signal and thus smaller fuel mass residual signal affects ML algorithm predictions. The improved DM simulating the fuel mass is mostly a result of improving the transfer pump outlet flow part of the DM, as the two are closely linked in Dymola. As the fuel mass is an accumulated signal, results show that making the perfect model is cumbersome. The effects of a small error early on in the flight remain during the whole flight period and make it difficult for the ML predictions. In addition to this, subparts of the fuel mass DM of jet pumps in the fuel system that can transfer smaller amounts of fuel remain unchanged. Small changes in the fuel mass due to the activation of jet pumps make the simulated data more inaccurate and complicate ML algorithm training.

Results from the average measurement scores show a great variety. The average $r_{pr}(t)$ and $\hat{r}_{pr}(t)$ difference was mostly reduced, where all test flights except for flight 1055 and 1088 showed a great improvement. Flight 1081 even showed an 60 % average $r_{pr}(t)$ and $\hat{r}_{pr}(t)$ difference reduction. On the opposite, the score of flight 1055 was increased by 41% and flight 1088 by 18% showing worse ML algorithm performance. As flight 1055 contained kinder flight data, and flight 1081 data from tougher flying maneuvers, no direct differences in results could be seen between the kinder and tougher flights. However, seven out of nine test flights showed increased accuracy, indicating that improving the fuel mass DM might help ML predictions. In regards of the average prediction interval, the results vastly differed from flight to flight. Five out of nine flights showed a smaller average prediction interval, where flight 1052 again showed the best improvement. Improvement results are again spread across the kinder and tougher flights, so no direct connections can be seen in those regards.

Effect of Improving Dymola Model: Fuel Mass Tank 2, Flight 39-9				
-	Average $r_{pr}(t)$ - $\hat{r}_{pr}(t)$		Average $I_{pr}(t)$	
Flight Nbr	Original DM	Improved DM	Original DM	Improved DM
1044	7.58	6.45	51.14	59.81
1047	8.55	8.01	45.36	37.98
1048	11.12	8.70	55.97	58.75
1052	6.50	3.90	47.39	33.01
1055	10.22	14.41	64.64	62.03
1081	35.03	14.03	68.99	56.93
1086	14.42	10.14	60.27	52.45
1088	6.83	8.09	46.85	51.33
1091	13.52	11.11	41.80	49.57

Table 4.12: Numerical scores showing the effect of improving the fuel mass DM. Note that the results differ a lot, while some flight data predictions are improved and some are worsened.

Fuel Mass Tank 2: Average $y_{meas,fm}(t)$ - $y_{sim,fm}(t)$, Flight 39-9		
Flight Nbr	Original Dymola Model	Improved Dymola Model
1044	22.98	11.10
1047	15.52	11.29
1048	22.11	8.53
1052	10.91	8.40
1055	33.05	18.07
1081	14.17	14.34
1086	31.39	17.22
1088	10.26	8.77
1091	27.25	22.57

Table 4.13: The average difference between simulated and measured flight data of the fuel mass. It can be seen that the difference between measured and simulated fuel mass has shrunk after improving the fuel mass DM.

Results of Table 4.13 show that each of the test flights with the exception of flight 1081 recorded a more accurate simulation signal of the fuel mass. This is an interesting result as scores from Table 4.12 show that the improved DM greatly improved results in flight 1081. However, the reason behind this is unknown but is believed to be connected to the recorded anomaly data in the flight. Figures 4.26 and 4.27 show how flight 1052 was affected by the improved fuel mass DM. A much more stable measured residual signal can be seen, and generally a much smaller prediction interval. Left-hand plot of Figure 4.27 shows that the average simulation signal error is rather constant, which seems to benefit ML algorithm prediction accuracy. In flight 1088, left hand plots of Figure 4.28 and Figure 4.29 show that the simulation data only is slightly improved. However, ML prediction accuracy is worsened where both the average prediction interval and the average difference between $y_{meas,fm}(t)$ and $y_{sim,fm}(t)$ is increased.

Original Dymola Model: Fuel Mass Tank 2

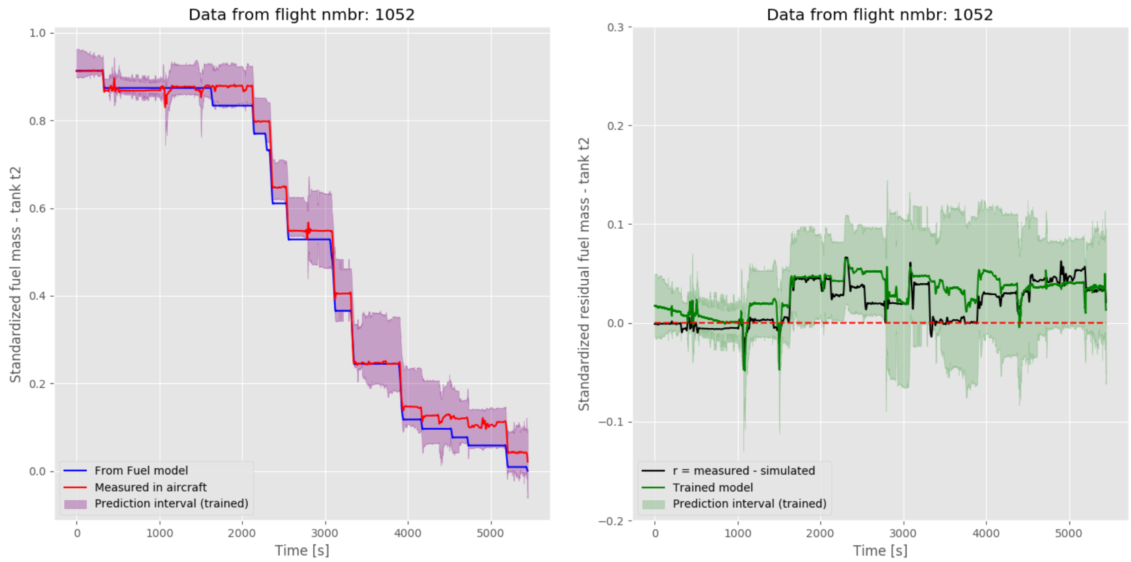

Figure 4.26: Simulated outlet flow $y_{sim,fm}(t)$ (red) and measured outlet flow $y_{meas,fm}(t)$ from the original DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Improved Dymola Model: Fuel Mass Tank 2

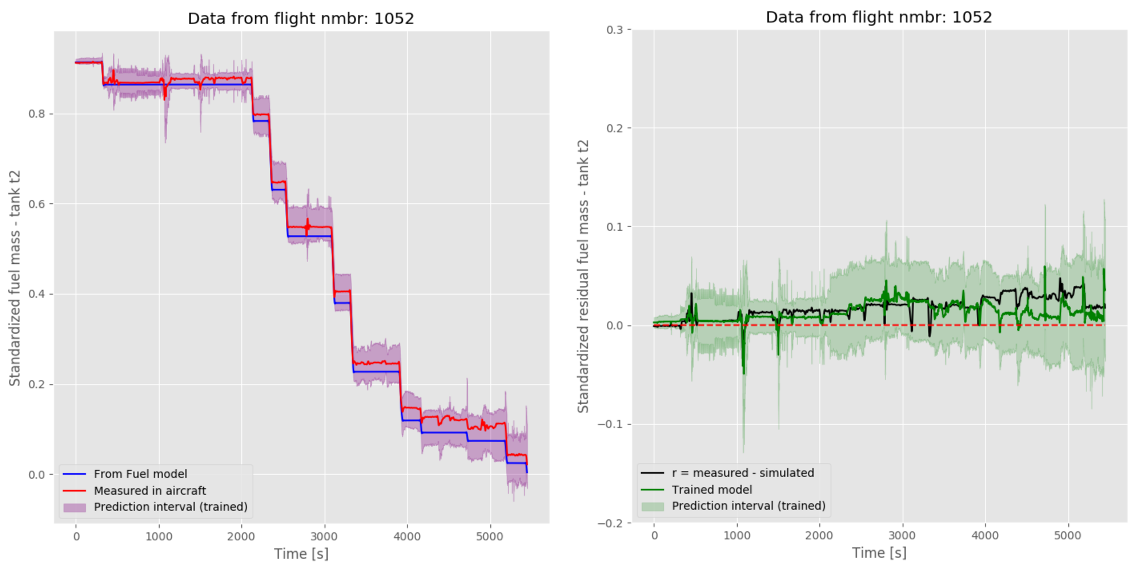

Figure 4.27: Simulated outlet flow $y_{sim,fm}(t)$ (red) and measured outlet flow $y_{meas,fm}(t)$ from the improved DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Original Dymola Model: Fuel Mass Tank 2

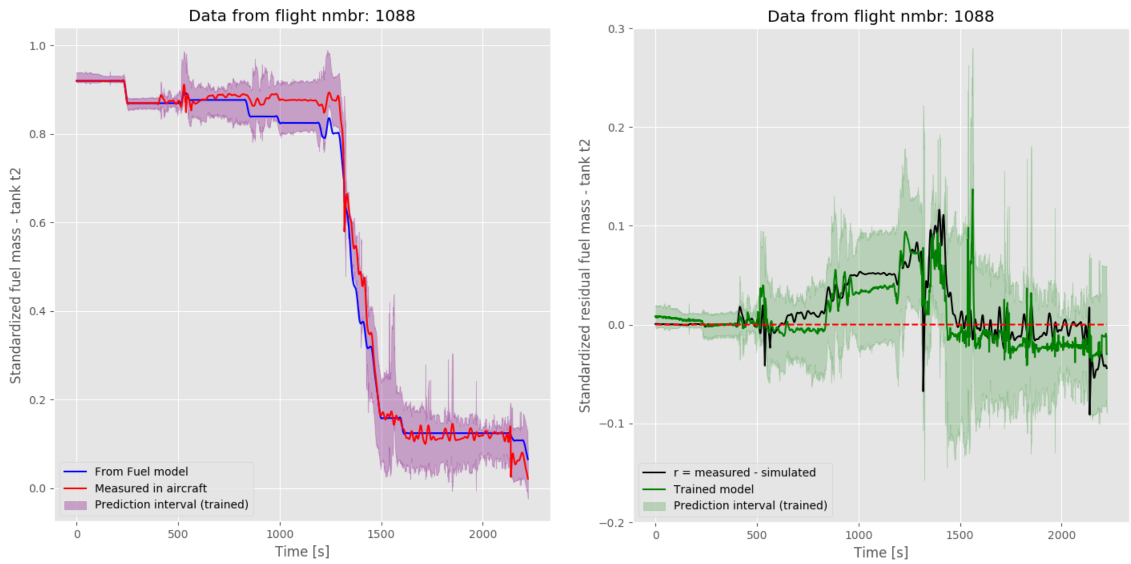

Figure 4.28: Simulated outlet flow $y_{sim,fm}(t)$ (red) and measured outlet flow $y_{meas,fm}(t)$ from the original DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Improved Dymola Model: Fuel Mass Tank 2

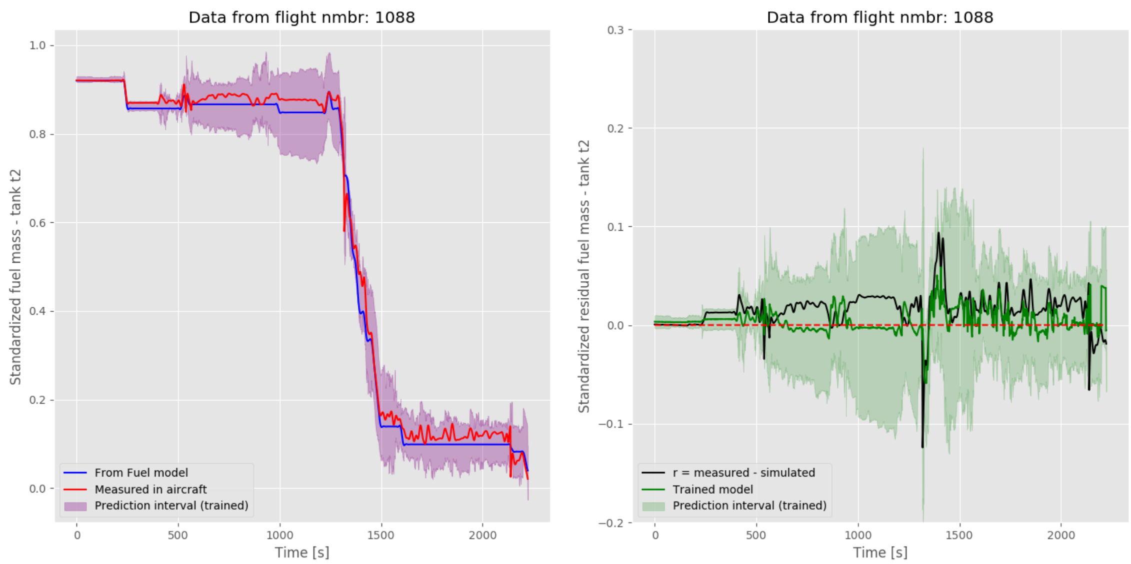

Figure 4.29: Simulated outlet flow $y_{sim,fm}(t)$ (red) and measured outlet flow $y_{meas,fm}(t)$ from the improved DM to the left. The corresponding measured residual $r_{fl}(t)$ and the predicted residual $\hat{r}_{fl}(t)$ to the right.

Detection of Anomalies in Fuel Mass of Tank 2

Lastly, results showing the anomaly detection in the fuel mass of tank 2 are presented. Shrinking the prediction interval would not have any effect if flights were misclassified to contain anomalies, so it is of great importance to accurately predict non-anomaly-containing flights as such and flights containing anomalies as anomalies. Out of the nine test flights, only flight 1091 was wrongly predicted as an anomaly. Data from the accurately predicted anomaly flight 1081 can be seen in Figure 4.32, where a large spike in the flow seen in Figure 4.22 causes the fuel mass to deviate from the prediction interval. In addition to this, the wrongly predicted anomaly flight 1091 can be seen in Figure 4.30 where a dip in the prediction interval causes the measured residual to differ from the prediction interval, triggering the anomaly prediction algorithm.

Faulty Anomaly Predicted Flight: Fuel Mass Tank 2

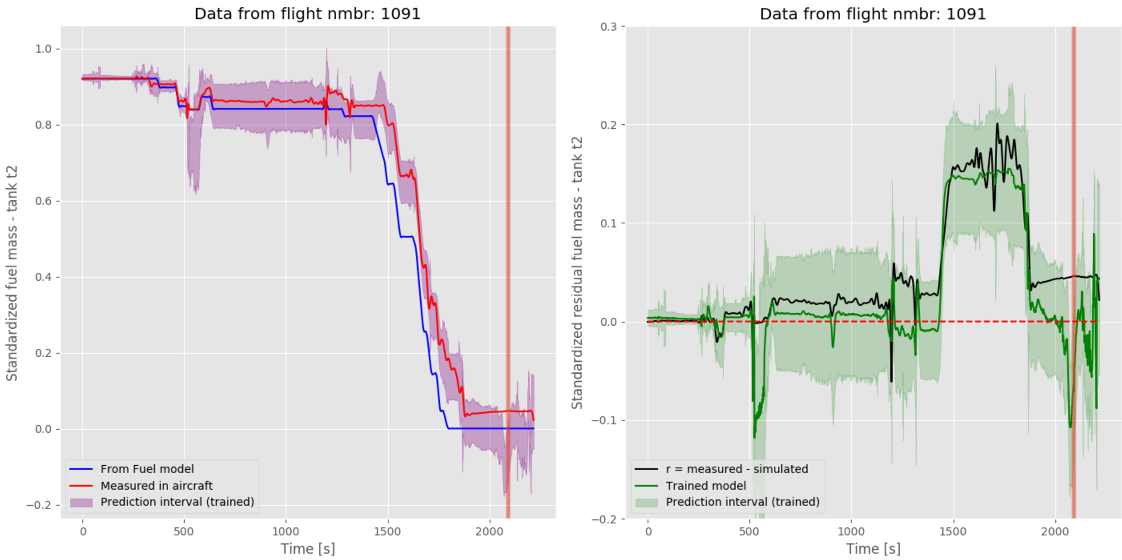

Figure 4.30: This is an interesting flight, as it is the only flight that was wrongly predicted as an anomaly. Right after 2000s, a dip in the prediction interval and predicted residual can be seen, making the measured residual deviate from the prediction interval enough to be classified as an anomaly.

Anomaly Flight: Fuel Mass Tank 2

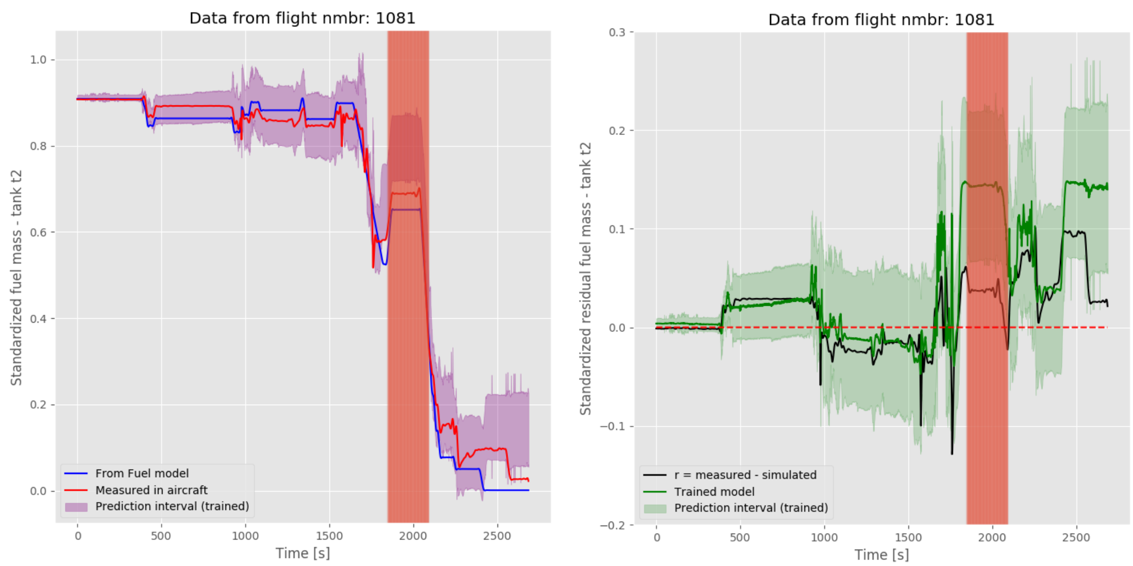

Figure 4.31: Data from anomaly flight where a clear deviation in data can be seen right at 2000s for an extended period of time, where the anomaly is marked in red.

Zoomed in Anomaly Flight: Fuel Mass Tank 2

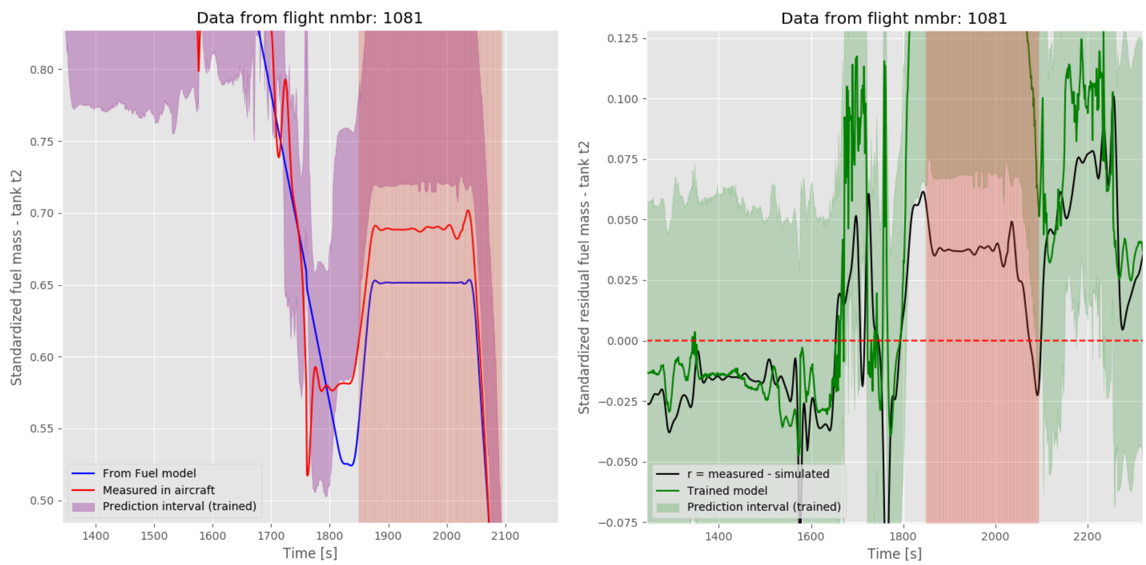

Figure 4.32: Zoomed in figure showing a close-up look at the anomaly. Note the small data deviation outside the prediction interval at 1760s.

5 Discussion

Analyzing flight data by using ML algorithms to detect anomalies is something that has been thoroughly researched [6], [10], [9] with great results. Moreover, seeing how an improved Dymola model affects anomaly detection and machine learning algorithms is something that differentiates this thesis, and acts as one of the central discussion points of this thesis.

This project has looked into three different target signals, the transfer pump outlet pressure, transfer pump outlet flow, and the fuel mass of tank 2, which all have vastly different properties. The variation in signal properties made it possible to explore a larger part of the Dymola model and shows where ML algorithms shine and perform worse. A signal analysis led to a better Dymola model of the three chosen signals. An in-depth analysis of the outlet pressure and outlet flow signals shows that ML algorithms can be trained to both predict the corresponding signal residual with great accuracy, as well as generate a precise prediction interval. The ML algorithms of said signals can be trained to predict anomalies with great precision. Signal analysis of the pressure and flow laid the ground for an improved Dymola model, which after being implemented simulates significantly more accurate signals of the pressure and flow. However, this did not facilitate the ML algorithm predictions, but other motivations for improving the Dymola model are still valid. In regards to the fuel mass, this project highlights the problems when training ML algorithms to predict an accumulated signal. A slight improvement of the predicted residual could be seen when generating more accurate simulation data as a result of the improved Dymla model. Lastly, looking at the three signals all results strongly highlight the importance of choosing the right features. In addition to this, achieving a good balancing of the data set is also crucial, which can also be backed up by related research of said subject [1]. This section presents an in-depth analysis of the results obtained in Section 4, where details that stand out are debated and the results presented are put into a context with regard to related research.

5.1 General Results

General Discussion of Fuel System Models

Generally, when comparing the original model (original RM settings and old DM) with the total improved models (new features, filters, etc, and improved DM) the results show that the outlet pressure and flow RMs have been improved where the prediction intervals have been reduced and the predicted residual is more accurate, while no false triggers of the anomaly detection alarm have been issued. This suggests that for these signals ML algorithms can be used with great results to both predict target signal residuals and to detect anomalies of an arbitrarily chosen signal of the fuel system with the same properties. The method used can then be extended to implement models of any signal of the fuel system with similar properties as the outlet flow and pressure. On the other hand, although the Dymola model generating the fuel mass in tank 2 has been improved, together with an improved fuel mass RM only a slight improvement can be seen in the reduced prediction interval and decreased nominal value. This points toward the difficulties when trying to model a signal with accumulated properties. The fuel mass RM performance seems to heavily depend on features selected together with the data balancing and is very sensitive to changes in said areas. However, by improving the DM submodel simulating the fuel mass a small improvement of the predicted residual can be seen, as shown in Table 4.12, suggesting that an accumulated signal such as the fuel mass might benefit from more accurate simulation data. To further explore this effect, the DM could be improved to also include the jet pumps, which are not accurately modeled in the DM as of now.

By looking at outlet flow score Table 4.6 and Table 4.7, as well as the outlet pressure Table 4.2, the greatest improvements can be seen by the adjusting the choice of features. As no significant improvements in the prediction interval or the predicted residuals can be seen by improving the DM generating the outlet flow and outlet pressure (See outlet flow Table 4.9, and outlet pressure Table 4.4), efforts should be put into altering the data set and finding the right features rather than improving the DM. The ML algorithms show to possess such power that there is no need to improve the DM of the flow and pressure. However, other projects or goals could still motivate the improvement of the Dymola models where the more accurate simulation data could be beneficial. The fact that feature selection and data preparation is of great weight for successful ML algorithm predictions is something that is also backed by *J. Sundell & C. Tysk* in the previous master thesis [20].

Feature Selection

As can be seen, by the results in Figure 4.4 and Figure 4.16 feature selection is of great significance when training the ML algorithms of the RMs. Specific features of the data set seem to contain patterns and a lot of information that can be connected to the target signals. More specifically the binary tank valve signals of the fuel system, together with the transfer pump valve level seem to help the outlet flow and pressure RMs a lot. Temperatures, pressures, and altitudes do only seem to have a slight influence on the prediction accuracy and they can be of great use to create a base set of features, as seen in Section 4.3 feature set one. However, these types of features do not seem to contain as much information relevant to the ML algorithms as the binary tank valve levels and the transfer pump valve level. The conclusion is that the level of pressure and flow largely depends on which tank is currently being drained and that these valve features should be of priority when choosing features.

The observation that the feature selection of the fuel mass RM is a tricky but important step is something also discussed by previous thesis writers in [20]. The problem seems to lie in the accumulated properties of the fuel mass, making the fuel mass more cumbersome to predict than the outlet flow and pressure. Different sets of features tested showed how sensitive the fuel mass RM is, leading to a significant variance in the results of residual predictions and the prediction intervals. Including a great portion of features such as fuel consumption and flight velocity resulted in large fluctuation in the prediction interval. Thus, this caused a lot of flights to be misclassified as anomalies, as seen in Figure 4.24. The otherwise important features of the transfer pump valve level and tank valve commands of tanks 1,2,3, and 6 resulted in worse predictions which are interesting, as these features greatly improved the results in the transfer pump outlet pressure and flow RMs.

Since the choice of features has such a large impact on the ML algorithm's capability to accurately predict the residual and create a reasonable prediction interval, it is believed that a more sophisticated method to choose the optimal set of features could be beneficial in all the RMs, but especially the sensitive fuel mass RM. The methods used to select the features in the RMs have their limitations, and a linear correlation analysis together with intuitive thought may be deficient. The importance of choosing a suitable set of features by using a more sophisticated method, e.g. by ML algorithms is discussed by *G. Fang et al.* in [3] where features are chosen by using various ML algorithms. In addition to this, it is important to remember that finding the right *set* of features can be of great significance, rather than finding single features. This aspect does of course make the process even more cumbersome but needs to be taken into consideration.

Level of Prediction Interval

The prediction interval is an important aspect when discussing the results. Not only does it decide how much of a deviation is allowed before data is classified as an anomaly, but it can also reveal important information about the DM and RM settings. Both the outlet pressure and flow RMs generally generated smaller prediction interval with the total model improvements (all improvements included), which can be seen in the figures for pump flow and outlet pressure, Figure 4.20 and Figure 4.12. Generally, this can be seen as an improvement as it indicates that ML algorithms in the pressure and flow RMs are more precise. However, it is important to remember that shrinking the prediction interval too much also can have negative consequences as flights can be misclassified as anomalies. On the other hand, using model settings that make the prediction interval become too large can make the anomaly detection algorithm miss anomaly flights. When looking at the general trends among test flights in regards to the prediction interval and accuracy in anomaly detection model accuracy, it is of great importance to always strive after a smaller prediction interval as long as none, or a small fraction of flights are misclassified as anomalies. The prediction intervals of flights tested from the improved pressure and flow RMs seem to be on a reasonable level, as the interval generally is small but no flights are misclassified as anomalies. What level of prediction interval is reasonable in the fuel mass RM is hard to decide, but the level of sensitivity seen when analyzing the fuel mass can motivate a (percentually) larger prediction interval rather than trying to acquire a smaller interval.

Difficulities of Modelling due to Part Degeneration and Faulty Measurements

One important aspect when investigating the fuel system is the fact that the system performance is affected by part degeneration and general wear and tear that comes from hours of flight [12]. Figure 4.3 shows that the general level of the transfer pump outlet pressure can differ up to 20 kPa. Not only does this create difficulties when training ML algorithms, but it also makes it tough to develop an accurate Dymola model simulating the outlet pressure, as this dynamic degeneration behavior is cumbersome to model. If this effect is due to uniqueness in the parts or due to wear and tear is unknown, but as the transfer pump that was put in was new it is believed that part degeneration is the main reason for the deviation. As flight data often is collected over some time this data deviation could result in a much greater data variance, larger prediction intervals, and inaccurate residual predictions. This effect could only be proved in the transfer pump outlet pressure, but it is believed that other signals in the fuel system also suffer from the same effect since the transfer pump is such a central part of the fuel system.

If it is true that the effect comes with part degeneration, a method to counteract this problem could be to model this decrease in performance in Dymola, where the flight hours of the aircraft could be included to create a model which considers the degeneration of the transfer pump. However, it might be challenging to find such a model, and the time and energy could be better spent elsewhere, perhaps improving various settings in the RMs to counteract this problem.

Anomaly Detection

The use of ML algorithms to detect anomalies is a well studied area [10], [9], [15]. *J Oehling et Al.* presents in [17] successful anomaly detection in flight data by using ML algorithms, where detection of anomalies in the flight data revealed dangers otherwise missed. By looking at Table 4.1, it can be seen that anomalies are detected with good precision for all test flights with the exception of the misclassified anomaly in the fuel mass of flight 1081. When analyzing the prediction interval of the test flights in general for all three target signals, it seems to be the case that non-anomaly-containing flights are at higher risk of being faulty predicted as anomalies rather than anomaly flights being missed.

There are different factors that may cause false alarms. The most likely reason is a lack of similar data of the misclassified anomaly in the training set due to bad data balancing or just a sparse training set, and the ML algorithms haven't been trained to handle such data. Other factors that could trigger false alarms include the absence of a specific feature that covers corner cases, e.g. not including features showing measured fuel mass in the drop tank or a bad version of the data balancing due to the stochastic behavior as seen in Figure 4.25. Even though this was resolved by freezing the indices, the frozen indices could still represent a bad version of the data balancing. The anomaly detection alarm should thus be used as an indication that something is suspicious and needs to be looked over rather than something catastrophic. This could instead reveal flaws in the fuel system or other kinds of information about the system that is beneficial to know.

Another important issue is how to choose the threshold that decides if a deviation outside the prediction interval should be classified as an anomaly or not, as seen in Section 3.1. The threshold is manually set, and largely decided by the general magnitude of the chosen signal (and studied anomalies). To gain a more accurate threshold level, *S Kumar Jasra et Al.* discuss in [6] the ability to use an ML algorithm to set the threshold. It is also discussed whether a supervised ML algorithm could be used to directly classify a set of flight data as an anomaly or not and shows that it could be beneficial to let ML algorithms decide a reasonable threshold. Another way to tackle the problem of anomaly detection could be to use an unsupervised ML algorithm, perhaps some sort of clustering algorithm, instead of the supervised neural network used in this thesis. The success of doing so is discussed by *L. Li et Al.* in [10].

Stochastic Behaviour and Data Balancing

As can be seen in Figure 4.25, a stochastic behavior in the ML predictions could be seen when re-running the fuel mass RM with the same settings. There are three steps in the RM training procedure that are stochastic, where data balancing is the most prominent. Training of the two ML algorithms is also stochastic, but this was resolved by setting a seed in the hyperparameters of the ML algorithms. Thus, the stochastic behavior can only be traced back to the data balancing. This can be seen as a huge flaw of the model as it resulted in different results each time the RM was re-run. In addition to this, the stochastic behavior can have skewed results from the previous master thesis [20], making conclusions from the previous thesis less reliable. As presented in the results, the stochastic behavior of the data balancing was resolved by freezing the indices of one balancing procedure, saving the indices, and re-choosing the same samples when running the notebook again. By freezing the indices the effect of other settings in the notebook can be distinguished, such as using different features. The procedure also introduces an uncertainty in the results as redoing this process could result in both better and worse predictions, depending on which data distribution is returned.

Note that the stochastic behavior of the data balancing still affects the outlet pressure and flow RMs, but not nearly as much as in the fuel mass RM. It is unknown whether the data balancing of the RM could be resolved by switching balancing features, but this is a possible way of further investigating the problem. The possible improvements of balancing an imbalanced data set are widely known and discussed by *M. C. Monard et Al.* in [1] where great improvements in ML algorithm performance can be seen by balancing an imbalanced data set. Although improving and investigating the data balancing further was out of the scope of this thesis, its importance is known. It is certain that the data balancing heavily affects the results of all models and could be further looked into for a better, preferably non-stochastic balancing model. Possible data balancing algorithms are suggested by *G. Lemaitre et Al.* in [8], where the *Scikit learn* open source library can be extended to test addition balancing methods.

5.2 Discussion of Method

The thesis goal creates a tough task of finding a good investigation method as the problem can be tackled in many different ways, thus there was no straight path to finding the optimal method of investigation. To research the main goal of this thesis - to expose areas where the models perform good and worse to improve anomaly detection, there was no obvious path to follow. Model flaws were mostly discovered just by working with the DM and RMs, iterating different RM settings, and trying to intuitive discover ways to improve the various sub-models. This points toward the unstructured nature of the project, which could be improved by a more structured working method. Another discussion point was that a lot of the settings tested did not make any significant difference, and a majority of changes did not improve the results. Although that could be seen as a result in itself, it was cumbersome to document each of the hundreds of small changes made. Generally, features were tested one at a time and small procedure changes in the RMs resulting in unchanged performance were not always documented. Only the settings of the most significant results were presented. This also shows the extremely large number of degrees of freedom. Differences due to, measurement errors, different features, and balancing settings creates such a large variety in flight data which makes model flaws and strengths tough to find, and which working method that best suits this type of problem is hard to see beforehand. Worth noting is that these results are only valid for one type of aircraft (Gripen E 39-9) which makes general conclusions for all Gripen E aircrafts hard to formulate. The fact that different results could be obtained when analyzing a new aircraft is not taken into consideration in this work.

The delimitations for this work also have their ups and downsides. To increase the structure of the project, a better pre-study could have been done on selected areas. The research question could have been stated in a way that targets specific model settings such as the data balancing or improving the DM instead of trying to improve the anomaly detection *in general*. A more elaborate related research study could have been done to suggest specific methods that could improve certain areas, e.g. the ML method to choose a set of features. The broadness of the work did make research difficult and is something to take into consideration for future projects of a similar type.

6 Conclusion

This section presents a summary of the main conclusions that are drawn from the results in Section 4 and discussed in Section 5. Additionally, suggestions for future work for further investigation are also presented.

6.1 General Fuel System Model Conclusions

This thesis presents a general method to investigate the DM of the fuel system, where anomaly detection of the explored signals plays a central part of the thesis. In addition to this, the level of trade-off between improving the DM and letting ML algorithms in the RMs handle vulnerabilities of the model is also investigated. Similar work where flight data is analyzed by ML algorithms to detect anomalies and flight hazards has been done with success [9], [10], [15], [17] and results introduced in this thesis also shows that anomalies can be detected in flight data of Saab 39 Gripen with good precision. However, this thesis is differentiated by the further exploration of improvement areas in the Dymola model and the previously mentioned tradeoff between improved simulation data and ML algorithms. This chapter will present a summary of the purpose and goals, and highlight the main achievements and findings of the project work. Furthermore, suggestions for future work areas where additional effort can be put are also shown. Listed below are the main conclusions that can be drawn from this thesis project.

- Improving the DM simulating the transfer pump outlet pressure and flow did not reduce the prediction interval nor increase the accuracy of residual predictions. Instead, the biggest factors that should be considered to improve ML algorithm performance lie in the feature selection. Data balancing and filtering do also improve said performance. The residual of target signals of the fuel system that do not possess accumulated effects can be predicted with great accuracy if proper preparations are done.

- In regards to feature selection for the transfer pump outlet pressure and flow, tank valve signals and the transfer pump valve level can reveal a lot of information and plays an important role in training ML algorithms.

- A considerable difference in measured outlet pressure can be seen between a new transfer pump and an older, worn-out transfer pump. Whether this effect was due to part uniqueness or wear and tear is still unknown. What is known is that this effect is present in the aircraft and needs to be taken into consideration when making changes in the DM.

- Signals possessing accumulated properties where errors are allowed to build up over time are much more challenging to predict for ML algorithms. The fuel mass is much tougher for ML algorithms to handle, and is very sensitive to variations in the data balancing and feature selection. An improved fuel mass DM can improve the prediction accuracy of the predicted fuel mass residual but does not make the prediction interval more precise. The problem is believed to lie in the accumulated properties of the fuel mass, and similar results are expected for signals of equal properties. This area needs more investigation before a more precise conclusion can be drawn.
- The current data balancing procedure possess such a stochastic behavior that it can significantly alter results, and make it hard to draw conclusions. The way this was resolved in this thesis might not be the best way to tackle the problem, and additional work should be put into this area.

6.2 Improvement Areas for Future Work

Work done in this thesis also discovered opportunities for further development, which are believed to contain even more information about the Dymola model. The list below shows areas that can lay the ground for future project work to gain additional information about the fuel system model.

- Trying to implement an RNN neural network, suited for time-series predictions to tackle the problem of predicting accumulated signals.
- Gathering more flight data for ML-model training - Doing this is believed to significantly help ML algorithm prediction accuracy, as sparse data can be increased. Obtaining more data on anomaly flights would increase the knowledge needed to decide a reasonable anomaly threshold, and could possibly enable the development of supervised classification ML algorithms for anomaly detection. In addition to this, a suitable ML algorithm could be developed to decide an anomaly threshold.
- A more sophisticated numerical measurement could be developed to measure the effect of different settings in the notebooks to decide whether the prediction interval is improved, or worsened. A measurement could also be developed to see how close the measured residual r(t) are to the prediction interval I(t), to see whether flights are on the verge of being classified as anomalies or not.
- It can be seen that a new transfer pump greatly affects the general signal pressure level investigated in this thesis. To look into and see if this is due to wear and tear or uniqueness of parts in the fuel system, or other factors can be of great interest. In addition to this, investigating the possibility of modeling this phenomenon in Dymola could be done, and creates an area of further Dymola model improvement.
- Results show that the features used when training ML algorithms in the models greatly affect the performance and anomaly detection ability. Studies also show that there are more sophisticated ML models available to tackle the problem of choosing a suitable feature set. As there is an interest in investigating additional signals with different properties besides the signals investigated in this project, an ML algorithm could be used to generate the best set of features of an arbitrarily chosen signal of the fuel system.

Bibliography

[1] Gustavo E. A. P. A. Batista, Ana L. C. Bazzan, and Maria Carolina Monard. *Balancing Training Data for Automated Annotation of Keywords: a Case Study*. Mar. 2003. URL: https://www.inf.ufrgs.br/maslab/pergamus/pubs/balancing-training-data-for.pdf.

[2] Mikel Canizo et al. *Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study*. June 2019. URL: https://www.sciencedirect.com/science/article/pii/S0925231219309877.

[3] Gang Fang, Wenbin Liu, and Lixin Wang. *A machine learning approach to select features important to stroke prognosis*. Oct. 2020. URL: https://www.sciencedirect.com/science/article/pii/S1476927120306307.

[4] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow*. Sept. 2019. URL: https://www.oreilly.com/library/view/hands-on-machine-learning/9781492032632/.

[5] Isabelle Guyon and Andre Elisseff. *An Introduction to Variable and Feature Selection*. Mar. 2003. URL: https://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf?ref=driverlayer.com/web.

[6] S Kumar Jasra et al. *Literature review of machine learning techniques to analyse flight data*. Oct. 2018. URL: https://www.um.edu.mt/library/oar/bitstream/123456789/58839/3/AEGATS_Paper.pdf.

[7] Vipin Kumar and Sonajharia Minz. *Feature Selection: A literature Review*. Mar. 2013. URL: https://faculty.cc.gatech.edu/~hic/CS7616/Papers/Kumar-Minz-2014.pdf.

[8] Guillaume Lemaitre, Fernando Nogueira, and Christos K. Aridas. *Imbalanced-learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning*. Jan. 2017. URL: https://www.jmlr.org/papers/volume18/16-365/16-365.pdf.

[9] Guoyi Li et al. *Operational Anomaly Detection in Flight Data Using a Multivariate Gaussian Mixture Model*. Sept. 2018. URL: https://papers.phmsociety.org/index.php/phmconf/article/view/474.

[10] Lishuai Li et al. *Anomaly detection in onboard-recorded flight data using cluster analysis*. Oct. 2011. URL: https://ieeexplore.ieee.org/abstract/document/6096068.

[11] Ingela Lind and Henric Andersson. *Model Based Systems Engineering for Aircraft Systems – How does Modelica Based Tools Fit?* Mar. 2011. URL: https://www.diva-portal.org/smash/get/diva2:473646/FULLTEXT01.pdf.

[12] M.Naeem, R.Singh, and D.Probert. *Implications of engine deterioration for operational effectiveness of a military aircraft*. July 1998. URL: https://www.sciencedirect.com/science/article/pii/S0306261998000245.

[13] Sebastián Maldonado, Richard Weber, and Fazel Famili. *Feature selection for high-dimensional class-imbalanced data sets using Support Vector Machines*. May 2014. URL: https://www.sciencedirect.com/science/article/pii/S0020025514007154.

[14] Maciej A. Mazurowski et al. *Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance*. Oct. 2008. URL: https://www.sciencedirect.com/science/article/pii/S0893608007002407.

[15] Milad Memarzadeh, Bryan Matthews, and Ilya Avrekh. *Unsupervised Anomaly Detection in Flight Data Using Convolutional Variational Auto-Encoder*. Aug. 2020. URL: https://www.mdpi.com/2226-4310/7/8/115/htm.

[16] Dipl-Ing Bettina Oehler. *Modeling and Simulation of Global Thermal and Fluid Effects in an Aircraft Fuselage*. Citeseer, Mar. 2005. URL: https://modelica.org/events/Conference2005/online_proceedings/Session6/Session6b3.pdf.

[17] Julian Oehling and David J Barry. *Using machine learning methods in airline flight data monitoring to generate new operational safety knowledge from existing data*. Apr. 2019. URL: https://www.sciencedirect.com/science/article/pii/S0925753518308269.

[18] P.E. Pintelas. *Feature selection for regression problems*. May 2014. URL: https://thalis.math.upatras.gr/~dany/Downloads/hercma07.pdf.

[19] Arjun Singh Saud and Subarna Shakya. *Analysis of look back period for stock price prediction with RNN variants: A case study on banking sector of NEPSE*. Apr. 2020. URL: https://www.sciencedirect.com/science/article/pii/S1877050920308851.

[20] Carl Tysk and Jonathan Sundell. *Adaptive detection of anomalies in the Saab Gripen fuel tanks using machine learning*. June 2021. URL: http://uu.diva-portal.org/smash/get/diva2:1446108/FULLTEXT02.pdf.

[21] A. Verikas and M. Bacauskiene. *Feature selection with neural networks*. Mar. 2002. URL: https://www.sciencedirect.com/science/article/pii/S0167865502000818.

[22] Yi Yao et al. *Balancing data for generalizable machine learning to predict glass-forming ability of ternary alloys*. Oct. 2021. URL: https://www.sciencedirect.com/science/article/pii/S1359646221006461.