

Prediction of Aircraft Take-off Weight using Machine Learning

AE5310: Thesis Control and Operations
Andrada Ioana Gheorghe

Delft University of Technology



Prediction of Aircraft Take-off Weight using Machine Learning

AE5310: Thesis Control and Operations

by

Andrada Ioana Gheorghe

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on the 27th of May 2024.

Student number: 4651995
Project duration: 01/06/2023 - 27/05/2024
Thesis committee: Dr. M.J. (Marta) Ribeiro
Dr. Junzi Sun
Pascal Hop
Benjamin Cramet
Prof.dr.ir. J.M. (Jacco) Hoekstra
Dr. Roberto Merino-Martinez

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Acknowledgements

This MSc thesis would not have been possible without the contribution of so many people around me, who supported and guided me along the entire process.

My sincere gratitude and appreciation goes to all my supervisors, who have contributed to an enriching traineeship experience, both from an academic but also professional perspective. On EUROCONTROL's side, I would like to thank Pascal for guiding and entrusting me with this exciting project. Thank you for offering me all the support I could ever ask for and for providing me with all the necessary information, data, and expertise. On TU Delft's side, I would like to thank Marta and Junzi for their valuable contribution to this research. Thank you for your guidance, feedback, and constant encouragement throughout the course of this MSc thesis. You have helped me gain confidence and get my ideas into shape.

Thank you Marilyn, Tamara, Pierre-Louis, and all the Aviation Sustainability Unit for welcoming me with open arms! You have truly made me feel part of the team and I genuinely appreciate your constant support, feedback, and positive energy.

I also want to thank Benjamin and his team, as well as Philippe and Gabriel who have extensively supported me throughout the development of this research, especially from an AI point of view. Thank you for always being so responsive, helpful, and for accommodating time for my study.

This acknowledgements section cannot be complete without mentioning my amazing friends with whom I shared so many ups and downs along our Aerospace Engineering journey at TU Delft. Alex, Crina, Nico, Octavian, and Tudor, thank you for being my continuous source of motivation and my support system, both emotionally and academically, since the beginning of our BSc. Nicolas and Ioana, thank you for an unforgettable MSc experience, I am glad to have met such genuine and encouraging friends with whom I share a passion for Air Traffic Management (ATM). Suyi, thank you for being the best housemate along all these years, your kindness and generosity are unparalleled.

Last but not least, I would like to thank my boyfriend Andy who has been by my side from the start of my traineeship at EUROCONTROL. Thank you for giving me programming and optimisation tips, I could not have asked for a better cybersecurity advisor. Your constant cheering and devotion has been a great source of motivation.

Most importantly, I want to thank my family. Many thanks to my grandparents and aunt, for always encouraging me and for showing me the importance of hard work and dedication. Thank you to my parents, Augustin and Daniela, for introducing me to the world of ATM, Air Traffic Control, and Aerospace Engineering from a very young age. Despite your best efforts to entice me into architecture, I regardlessly followed your footsteps into the aviation field. Thank you for making related dinner conversations so enthralling and for being my aviation role models. I cannot thank you enough for all your efforts, constant support, trust in me, and of course, the most straightforward expert advice.

Andrada Ioana Gheorghe
Delft, April 2024

Contents

List of Abbreviations	vii
Introduction	xi
I Scientific Paper	1
1 Introduction	3
2 Current Work	4
2.1 Literature Review	4
2.2 Operational Context	5
3 Methodology	5
3.1 Datasets & Features Selection	6
3.2 Features Encoding	9
3.3 Machine Learning Algorithms.	10
4 Description of the Case Studies	13
4.1 Amsterdam Airport Schiphol (AMS).	13
4.2 Aircraft Types	13
4.3 Validation Datasets	14
5 Results	15
5.1 AMS Case Study.	15
5.2 Extended Applications: CDG & CRL Airports	19
5.3 Rate of Climb Validation	21
6 Discussion	26
6.1 Comparison with Previous Studies	26
6.2 GBDTs Model Applicability and Improvements	27
7 Conclusions and Recommendations for Future Work	28
II Literature Study	
previously graded under AE4020	41
1 Introduction	43
2 Problem Statement	44
3 Background Information	46
3.1 EUROCONTROL	46
3.2 LTO Cycle	47
3.3 ICAO 2012 Flight Plan.	49
3.4 Trajectory Computation Infrastructure	50
3.5 Machine Learning.	55
4 Literature Review	61
4.1 Current Trajectory Prediction	61
4.2 Academic Research	61
4.3 Climb'n Descent Project - EUROCONTROL	64

5	Data Available	68
5.1	FlightRadar24	69
5.2	Aireon.	69
5.3	Flight Plans	71
5.4	Weather & Wind Forecast	75
6	Research Gap	77
6.1	Data Considered	77
6.2	Prediction Time Frame	77
6.3	Results Applicability	77
7	Research Proposal	78
7.1	Research Questions and Objectives	78
7.2	Methodology	79
7.3	Case Studies	82
7.4	Verification & Validation.	83
7.5	Work Breakdown	84
8	Conclusion	85

List of Abbreviations

ADS-B	Automatic Dependent Surveillance-Broadcast
AEDT	Aviation Environmental Design Tool
AGI	Artificial General Intelligence
AI	Artificial Intelligence
ALC	Aviation Learning Centre
AMS	Amsterdam Airport Schiphol
ANI	Artificial Narrow Intelligence
ANN	Artificial Neural Network
ANSP	Air Navigation Service Provider
APM	Aircraft Performance Model
ASI	Artificial Super Intelligence
ASU	Aviation Sustainability Unit
AT	At
ATC	Air Traffic Control
ATCO	Air Traffic Controller
ATFM	Air Traffic Flow Management
ATM	Air Traffic Management
BADA	Base of Aircraft Data
BECMG	Becoming
CAS	Callibrated Air Speed
CDG	Paris - Charles de Gaulle Airport
CDM	Collaborative Decision Making
CNN	Convolutional Neural Network
CO	Carbon monoxide
CPR	Correlated Position Report
CRL	Brussels South Charleroi Airport
CTR	Control Zone
DNN	Deep Neural Network
DTR	Decision Tree Regression
EAIP	Electronic Aeronautical Information Publication

EASA	European Union Aviation Safety Agency
ECAC	European Civil Aviation Conference
ECTL	EUROCONTROL
EFD	ETFMS Flight Data
EOBT	Estimated Off-Block Time
ETA	Estimated Time of Arrival
ETFMS	Enhanced Tactical Flow Management System
EU	European Union
FAA	Federal Aviation Administration
FDR	Flight Data Recording
FF-ICE	Flight and Flow Information for a Collaborative Environment
FIR	Flight Information Region
FL	Flight Level
FM	From
FMS	Flight Management System
FPL	Flight Plan
FR24	FlightRadar24
FTP	File Transfer Protocol
GBDT	Gradient Boosting Decision Tree
GBR	Gradient Boosting Regression
GBRT	Gradient Boosted Regression Tree
GFS	Global Forecast System
GRIB	General Regularly distributed Information in Binary form
HC	Hydrocarbon
IAS	Indicated Airspeed
ICAO	International Civil Aviation Organisation
IFR	Instrument Flight Rules
ISA	International Standard Atmosphere
LTO	Landing and Take-off
M	Mach number (cruise)
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
METAR	Meteorological Aerodrome Report
ML	Machine Learning

MLPNN	Multi-Layer Perceptron Neural Network
MSc	Master of Science
MTOW	Maximum Take-off Weight
MUAC	Maastricht Upper Area Control Centre
NM	Network Manager
NMOC	Network Manager Operations Centre
NO	Nitrogen oxides
ODE	Ordinary Differential Equation
OPSLOG	Operational Log
PQI	Performance Quality Indicator
PReLU	Parametric Rectified Linear Unit
PREQUAL	Prediction Quality
PTF	Performance Table File
QAR	Quick Access Recorder
R&D	Research & Development
RIU	Runway In Use
RMSE	Root Mean Squared Error
ROC	Rate of Climb
RR	Ridge Regression
RWY	Runway
SADIS	Satellite Distribution System
SAT	Sustainable Air Transport
SES	Single European Sky
SESAR	Single European Sky ATM Research
SID	Standard Instrument Departure
SReLU	Standard Rectified Linear Unit
STAR	Standard Terminal Arrival Route
TAF	Terminal Aerodrome Forecast
TARGETS	Terminal Area Route Generation and Traffic Simulation
TAS	True Airspeed
TEMPO	Temporary
TMA	Terminal Manoeuvring Area
TOC	Top of Climb
TOW	Take-off Weight
UIR	Upper Flight Information Region
WAFS	World Area Forecast System

Introduction

Air Traffic Management (ATM) is a highly dynamic sector relying on the input of numerous stakeholders, of which airlines, airports, Air Navigation Service Providers (ANSPs), and aviation authorities. In order to regulate the flow of air traffic accordingly, trajectory predictions are needed. Their computation is based on several aircraft performance parameters, of which Take-off Weight (TOW) plays a crucial role.

TOW is the starting point for fuel consumption trends and climb performance, and certainly one of the most-sought parameters by aviation stakeholders. Unfortunately, aircraft operators are generally not cooperative in disclosing this data as it is integral to calculating their cost index, potentially revealing sensitive insights into their operational practices. Such disclosure could leave them susceptible to market competition and possible penalties.

This MSc thesis was conducted within a traineeship, in collaboration with the Aviation Sustainability Unit (ASU) at EUROCONTROL Headquarters in Brussels (Belgium). EUROCONTROL being responsible for managing and coordinating the European network, is also interested in improving their flight planning and emissions calculations prior to take-off, but without TOW data, prediction accuracy is diminished. For this reason, this MSc thesis aims to treat the root of the problem by predicting aircraft TOW in a (pre-)tactical setting, that is up to seven days prior to the day of operation, including the latter. Based on this prediction horizon, only Flight Plan (FPL) and Terminal Aerodrome Forecast (TAF) operational parameters are considered.

For innovation purposes, and following the methodology and results of previous studies, Machine Learning (ML) algorithms are explored in this study, specifically Gradient Boosting Decision Trees (GBDTs) and Random Forests. Although these have previously been used for TOW estimations, there has been no attempt to predict aircraft TOW prior to take-off, solely based on FPL information and trends. Actually, none of the studies had full access to FPL data, whereas it was provided by EUROCONTROL for this MSc thesis.

This report is divided into two main parts. First, [Part I](#) provides the scientific paper, in which the topic is detailed from problem formulation, methodology followed, and results obtained. A discussion is also provided at the end of the paper, along with several recommendations for future work. In [Part II](#), the Literature Study is attached. The latter was conducted in the beginning of the project and contains the relevant information supporting this MSc thesis research.

I

Scientific Paper

Prediction of Aircraft Take-off Weight using Machine Learning

Andrada Ioana Gheorghe,*

Delft University of Technology, Delft, The Netherlands

Abstract

Predicting aircraft Take-Off Weight (TOW) has been a long-sought task by aviation stakeholders, especially for operational and regulatory bodies involved in flight planning. Unfortunately, TOW being a sensitive parameter to operational trends and cost indices, aircraft operators tend to keep it confidential. In recent years, Machine Learning (ML) algorithms have achieved increased prediction accuracy and capabilities in the field, provided the availability of TOW data. This paper studies the implementation of gradient boosting algorithms as well as Random Forests to better understand which algorithm is best-suited for aircraft TOW prediction (prior to take-off) solely based on Flight PLan (FPL) and Terminal Aerodrome Forecast (TAF) parameters. The study focused on flights at Amsterdam Airport Schiphol (AMS) for training the algorithms, using an 80-20% train-test split. Between Gradient Boosting Decision Trees (GBDTs), LightGBM, XGBoost, and Random Forests, GBDTs achieved the smallest Mean Absolute Percentage Error (MAPE) with 1.71 and 2.17% on the training and testing datasets, respectively. The most influencing feature proved to be the requested cruise speed, followed by great circle distance between airports, and aircraft type. The model was validated on Paris - Charles de Gaulle Airport (CDG) and Brussels South Charleroi Airport (CRL), proving its independence from airport type. However, the distribution of flights in the training dataset, especially that of aircraft and airline types, proved to be an influencing factor for the model's applicability to other airports. Future work includes expanding the training dataset to all flights in the European network, and introducing trajectory-based features such as aircraft speed intent. With a larger training dataset, neural network algorithms could also be explored. Finally, regarding the improvement of trajectory predictions, it was found that better accuracy of TOW predictions does not suffice and that other operational parameters' effect should be investigated, especially speed profiles.

1 Introduction

The prediction of aircraft Take-Off Weight (TOW) has been a difficult problem to solve for many aviation stakeholders. More than just a safety-critical parameter for take-off performance, TOW impacts fuel consumption and plays an important role in trajectory prediction computations, especially during the climb phase. Most operational and regulatory organisations involved in flight planning and network operations are looking to improve their flight planning and emissions calculations prior to take-off, but without TOW data, the accuracy of such predictions cannot be guaranteed. Unfortunately, aircraft operators are generally not willing to share this data because it is used for calculating their cost index, which may reveal sensitive information about their operational trends, making them vulnerable to market competition or even penalties. However, predicting aircraft TOW could enable aviation authorities to better compute emissions and other climate-oriented parameters, thus giving them the possibility of better regulating aviation's climate impact. To achieve this with current tools, an effective approach to reaching higher prediction accuracy is the improvement in the quality and fineness of their input data, of which TOW.

The current state of the art, for the most part, studied the estimation of aircraft TOW using supervised Machine Learning (ML) algorithms. However, these are highly dependent on data quality, quantity, and selection, and with the scarcity of TOW data, training a ML algorithm becomes difficult. For this reason, previous studies have relied on trajectory data (mostly sourced from Automatic Dependent Surveillance-Broadcast (ADS-B)) to build a training dataset by reverse engineering trajectories with a total energy model. This approach introduces assumptions and trains the algorithms on synthetic data, which may not be the most optimal approach. Furthermore, due to the sequential nature of the data, most predictions involved a sequence of mass estimations for the climb profile, although some attempted to estimate TOW using runway ADS-B data. Finally, all studies involved post-flight computations, which is not practical for flight planning and operational applications prior to take-off. Summarising, to the knowledge of the author, there has been no successful attempt to estimate aircraft TOW prior to take-off, solely based on flight plan information and trends. This work introduces Flight PLans (FPLs) and bases its TOW predictions solely on operational parameters contained in the latter. As

*Msc Student, Sustainable Air Transport, Faculty of Aerospace Engineering, Delft University of Technology

airlines have the option of filing their *operational* TOW in the FPL, the data used in this study is the closest to real TOW data and provides the best-achievable accuracy available to operational stakeholders. The use of FPLs, provided by EUROCONTROL for this study, captures airline preferences and enables a (pre-)tactical prediction horizon, that is one-to-seven days prior to take-off including day of operations.

This research paper is structured as follows. section 2 highlights the main take-away points from previous studies, including potential research gaps to be filled in this work. section 3 details the methodology followed to develop the model, including the ML algorithms and features selected. Next, section 4 describes the case studies and data selection procedure. The findings of the analysis are discussed in section 5, together with results from two validation activities treating the model’s applicability and its results’ relevance to trajectory prediction. Finally, the results are compared against previous studies and several points for improvement are discussed in section 6, followed by the presentation of conclusions of the project in section 7 along with some recommendations for future work.

2 Current Work

This section provides important background information regarding previous work done on the topic of aircraft TOW prediction. section 2.1 presents multiple studies done in recent years, while section 2.2 provides soon-to-be implemented operational concepts.

2.1 Literature Review

The most important consideration regarding previous research is the lack or scarcity of real TOW data. As previously stated, being a sensitive parameter, aircraft operators generally do not share this data outside of their entities, even though they do have the possibility of filing it in the FPL. For this reason, previous studies have attempted to deduce aircraft mass via analytical calculations, focusing on estimating the parameter *after* the flight has taken place. The computations are usually based on flight trajectory data such as ADS-B or radar Correlated Position Reports (CPRs). [Alligier and Gianazza, 2018], [Alligier et al., 2015], and [Sun et al., 2016], [Sun et al., 2017], [Sun et al., 2018], [Sun et al., 2019] have made use of The OpenSky Network [OpenSky, 2018], an open-source platform providing real-time and historical ADS-B data for research and academia, while [Chati and Balakrishnan, 2017] and [He et al., 2020] based their work on Quick Access Recorder (QAR) data. The latter is an airborne flight data recorder designed to provide raw flight data and is mainly used by aircraft operators for routine monitoring of their fleet and flight crew [flightrecorder.com, 2023] [Administration, 2004]. These data sources introduce constraints to the models’ accuracy and prediction capabilities.

Approaching the problem backwards involves building training datasets containing *synthetic* TOW data, before applying ML methodologies. Following the sequential nature of the available data (i.e. trajectories), the current state-of-the-art approach opts for reverse engineering a sequence of aircraft masses using a total energy model, generally over the climb profile. These methods adjust the mass to fit observed values of energy variation. Note that, although [Sun et al., 2016]-[Sun et al., 2019] used statistical methods instead of ML, they still take this approach for TOW and mass estimations. Not only does this introduce assumptions, and with it errors, it also limits the models’ capability of estimating TOW to a post-operations time frame, with no real (pre-)tactical prediction capabilities, which are essential in Air Traffic Management (ATM). At most, the predictions are computed using past trajectory points and with a 10-minute prediction horizon, as done in [Alligier and Gianazza, 2018].

Although the reverse engineering step is no longer needed when using QAR data, the prediction time frame issue persists. This is due to the capability of the flight data recorder itself, which provides real aircraft mass data at each point along the trajectory, yet only when the aircraft is airborne. Additionally, building a model on QAR data introduces limitations to its applicability due to the origin of such datasets. For example, the studies mentioned above ([Chati and Balakrishnan, 2017] and [He et al., 2020]) both used QAR data gathered from two airlines respectively; so although the data quality is improved, the predictions become airline-specific. While QAR data access is not limited to airlines, it is typically tightly regulated due to privacy and security concerns, so it becomes very difficult to build a dataset with a large variability in aircraft types, airlines, and origin and destination pairs.

With ML making its way through various fields of technology, its application to ATM has not been left untreated. Besides [Sun et al., 2016]-[Sun et al., 2019], that used statistical methods, all other studies mentioned above used different ML algorithms, from neural networks to conventional decision trees. While ML is built upon a statistical framework, their fields of application may differ. Statistical methods tend to perform better on smaller datasets, in the order of a couple of 10,000 samples. When applied to large datasets, they are not very efficient due to large memory requirements and fit time complexity [He et al., 2020]. Furthermore, statistical methods are based solely on probability spaces, making them more appropriate for finding relationships and patterns between features, rather than optimising predictions [Stewart, 2019]. For both reasons, it was decided

to focus on ML for this study, specifically supervised learning algorithms.

Following previous research findings, there is a clear lack of FPL integration in the studies, as well as long term prediction capabilities extended to at least a few hours before take-off. To the best of the author’s knowledge, there has been no successful study attempting to predict TOW solely based on FPL data, with a (pre-)tactical prediction horizon. While [Alligier et al., 2015] and [Vouros et al., 2022] have used some FPL information, they either did not have access to the entirety of the dataset, or simply did not use it as training features to their models. Omitting FPLs removes the airlines’ preferences from the analysis, and makes the results purely trajectory-based, without having intent or route planning information. Furthermore, it has no added value for ATM authorities, as the predictions cannot be applied prior to the flight’s execution. This study aims to better incorporate these features as well as increase the prediction horizon that is best suited for ATM flight planning applications. The goal of this research is to predict TOW *before* take-off, hence with parameters available in the FPL itself as well as Terminal Area (Aerodrome) Forecast (TAF) at the airport of destination to include weather impact. In terms of ML, the current problem is categorised as a single-output supervised learning regression task. Based on previous literature, two main algorithms (and some variations of it) will be explored: Gradient Boosting Decision Trees (GBDTs) and Random Forests. These proved to be the most effective and least error-inducing algorithms for predicting single variables.

2.2 Operational Context

Besides academic research on TOW prediction, it is important to acknowledge recent and ongoing developments concerning the new FPL format. The Flight and Flow Information for a Collaborative Environment (FF-ICE) format is expected to be implemented into operations by the end of 2025. The following information has been acquired from EUROCONTROL experts. The FF-ICE format was introduced to expand the information provided by aircraft operators for trajectory prediction improvement. The enriched data includes:

- 4D trajectory predictions, including the aircraft mass at each point along the trajectory;
- aircraft *operational* TOW predictions;
- flight-specific performance data, used to match the trajectories predicted by the EUROCONTROL Network Manager (NM) with those predicted by the aircraft operators, and to further refine the trajectory predictions in the NM systems.

Although this sounds promising, it is important to note that the mass profile and TOW predictions listed above are *optional* parameters to be provided by aircraft operators. Airlines generally classify this data as sensitive, hence not many may agree to actually share it with EUROCONTROL. Even if they do agree, these remain estimations which could be done with a less-efficient algorithm or one that is not ML-based. Nevertheless, it is expected that the mass estimations provided by aircraft operators will have better accuracy, since they have access to their own fleet-specific data. Lastly, while the mandate for FF-ICE FPLs implementation is set to the end of 2025, it is difficult to guarantee that this will be followed by all aircraft operators. For all stated reasons, the motivation of this study in predicting aircraft TOW remains valid, while FF-ICE FPLs could be considered for future applications.

3 Methodology

The methodology followed to build the appropriate ML model for TOW prediction is detailed in this section. An overview of the steps taken and datasets used is shown in Figure 1, where each white box represents one dataset. Note that, while potential applications of this study have been discussed in section 1, the main scope remains TOW prediction. A validation exercise linking the latter to potential improvements in Rate Of Climb (ROC) prediction is presented in section 5.3, yet this was conducted as an additional analysis.

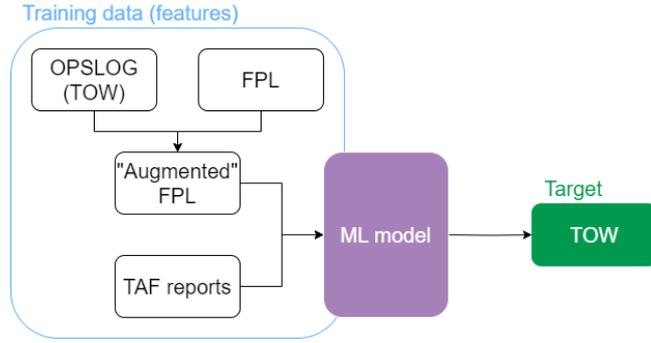


Figure 1: Methodology overview flowchart.

In ML, data plays a crucial role in defining the capability and applicability of the model to predict the selected target parameter(s). The data considered for this study, including the features selected for training, are presented in section 3.1, followed by their encoding in section 3.2. Regarding the ML algorithms, several have been tested in order to select the best-performing one for this particular application. Their selection as well as their basic working principle are detailed in section 3.3.

3.1 Datasets & Features Selection

This study has gathered data from different sources, as shown in Figure 1, all provided by EUROCONTROL. There are three main datasets: OPSLOG, FPL, and TAF; each containing information for one flight prior to take-off. As detailed in section 2.2, the aircraft TOW is an *optional* parameter which *can* be filed in the FPL, depending on the airline's willingness to share this data. Statistically, about 30% of the flights pertaining to EUROCONTROL's network - data source of this study - have a TOW associated with their FPL. Although this may seem like a small amount, it corresponds to circa 3 million flights in 2023 [EUROCONTROL, 2024].

The "OPSLOG" stands for operational logbook and it contains all the information about the flight in question, necessary to operational personnel in the execution of their duties. Generally, the operational logbooks are used by the EUROCONTROL NM for Air Traffic Flow Management (ATFM). Each flight has a series of operational logs depending on different changes or additional updates made to it. Its contributors include airlines, Air Navigation Service Providers (ANSPs), and airports, following aviation authorities' regulations such as the International Civil Aviation Organisation (ICAO) and the European Union Aviation Safety Agency (EASA). For the purpose of this study, the most recent and updated version of the operational logs was considered for each flight, since it contains the most updated and accurate information. Contrary to what one may suspect, the aircraft TOW is *not* stored in the FPL dataset, but in the OPSLOG dataset. Hence this dataset contains the target output label sought to be predicted by the model.

The official definition of a FPL given by ICAO is as follows: 'A navigation FPL is a document prepared in accordance with the instructions of the operator contained in the Operations Manual and used in flight by the pilot to assist in navigation and safe operation of the aircraft.' [(ICAO), 2005]. Detailed rules regarding contents, completion, changes to, and closing of a FPL can be found in ICAO's Annex 2 [(ICAO), 2005] and in national flight information publications [Skybrary, 2023a]. When referring to FPL data, this paper refers to the ICAO 2012 FPLs. This FPL format was fully implemented on the 15th of November 2012 and it is the current standard in air transport operations [Skybrary, 2023b]. FPLs are made up of 19 items. The majority of the items are completed by the aircraft operators, yet some fields require the input of ATC and communication services. The items that airlines must fill in when filing a FPL are listed in Table 1 along with their contents.

Once a FPL is filed by an aircraft operator, it is received by the NM Operations Centre (NMOC) at EUROCONTROL. The latter validates, corrects (if necessary), and distributes them to the ANSPs and operational partners concerned [EUROCONTROL, 2023b]. Only after NMOC has accepted the FPL is the aircraft operator able to use it for its subsequent flight. The validated FPL is the one considered in this study. Note that some of the parameters present in this dataset are almost identical to the ones in the OPSLOG dataset, however, the latter is more accurate and complete due to its operational nature. For example, the estimated taxi time filed in the FPL is generally an average standard value for each airport, however, the exact same parameter in the OPSLOG dataset may vary at an airport depending on the distance to the take-off runway, provided that the airport supports Collaborative Decision Making (CDM). For this reason, all identical parameters will take their value from the OPSLOG dataset, while the remaining features are directly extracted from the FPL dataset. The joining of these two is what yields the "Augmented" FPL shown in Figure 1, where the name refers to the association of the flight's FPL with its operating aircraft's TOW.

Table 1: FPL items completed by aircraft operators [Skybrary, 2023b].

FPL item number	Contents
Item 7	Aircraft identification
Item 8	Flight rules and type of flight
Item 9	Number and type of aircraft and wake turbulence category
Item 10	Equipment
Item 13	Departure aerodrome and time
Item 15	Crusing speed, altitude/level, and route
Item 16	Destination aerodrome, total estimated elapsed time, and alternate aerodrome(s)
Item 18	Other information
Item 19	Supplementary information

Finally, the TAF dataset contains TAF reports, namely forecasted weather conditions for a given area around an airport. Each report is updated four times a day and is valid for 30 hours [Centre, 2023]. TAF reports are only applicable to low altitudes and in the vicinity of the aerodrome of departure/destination. Specifically, they are valid in a radius between 8 to 16 [km], as shown in Figure 2. The altitude up to which they are applicable depends on the height of the ConTRol zone (CTR) around the respective aerodrome, which can be found in its Electronic Aeronautical Information Publication (EAIP). These tend to be in the same order of magnitude, for example at Amsterdam Airport Schiphol it is up to 3,000 [ft] [LVNL, 2023].

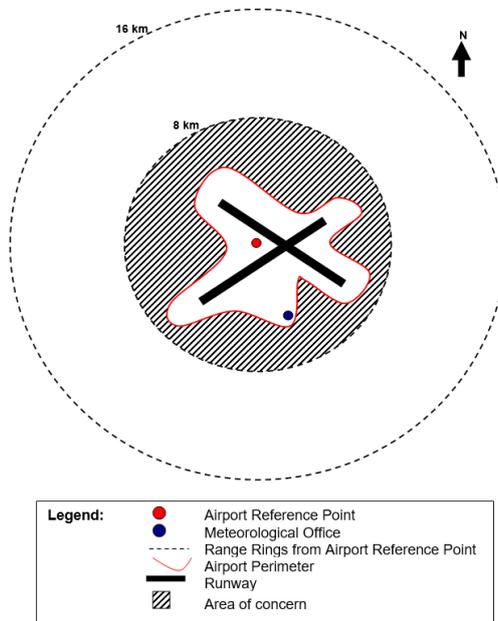


Figure 2: Illustration of aerodrome area around which the TAF is valid [Boase, 2008].

The weather data is produced by the Satellite Distribution System (SADIS) who delivers World Area Forecast System (WAFS) data. SADIS is developed and operated by the MET Office on behalf of the ICAO. All ICAO member states can have access to the SADIS data, by implementing a connection to the SADIS File Transfer Protocol (FTP). A connection was already present at EUROCONTROL and data has been gathered since 2015.

The airport of destination was selected to extract TAF parameters, as it is expected to have a higher influence on the aircraft TOW rather than the departure aerodrome. The reasoning behind this is that, the weather forecast at the airport of destination, given to the cockpit crew before take-off, may determine them to increase the extra fuel carried on board for potential exceptional situations such as holding or flight diversions. On the other hand, it is rare that the TAF report at the departure aerodrome would influence the aircraft TOW, hence it was left out.

TAF reports are given in the same format as a METeorological Aerodrome Report (METAR), a specific format for reporting weather information [Skybrary, 2024]. It is important to understand that within a TAF, there can be several forecasts for different time periods and with different probability of occurrence. These are called TREND forecasts. To illustrate an example, one TAF may have prevailing weather conditions throughout its validity time period, yet there may be a 40% probability that wind gusts of 20 [m/s] will be present for a period of 30 [min]. TREND forecasts are indicated by BECMG (becoming) or TEMPO (temporary) which

may be followed by a time group (hours and minutes UTC) preceded by one of the letter indicated FM (from), TL (until), AT (at). In order to get the most accurate results, the appropriate TREND forecast within the TAF must be considered (when applicable). First, the airport of destination and station at which the TAF was published must be the same. The next conditions involve timestamps. The Estimated Time of Arrival (ETA) of the flight must be enclosed within the start and end validity time period of the TAF report, while the Estimated Off-Block Time (EOBT) must be larger than the time at which the TAF was generated. Next, the following set of rules have been applied (in order!) to select the right TREND forecast within the identified TAF for the right flight:

1. The indicator must be TEMPO or AT without a probability associated (probability must be *Nan*); if this is not the case the following condition is:
2. The indicator must be TEMPO or AT with a probability associated (and ranked by probability in descending order, from highest to lowest). Only consider the TREND forecast with the highest probability; if there is no such forecast:
3. The indicator must be BCMG or FM; and otherwise:
4. Select the prevailing weather conditions of the TAF

For simplification purposes, the open-source Python library *Metafora* was used for parsing raw TAF reports [Ramon Dalmau-Codina, 2024]. The output is a dataframe with all TAF parameters of interest, which can now be used as features for the ML algorithms.

All training features are listed in Table 2, together with the data type and unit used, when applicable, as well as the dataset they were extracted from. Note that some of these features are not taken directly from the dataset and were altered for different reasons. First, the departure, destination, and alternate aerodromes were not considered as categorical features themselves. Instead, for generalisation of the model and to make it independent of ICAO airport codes, the great circle distance d between the aerodromes of departure and destination, as well as that between the destination and alternate aerodromes were calculated using Equation 1 [Corporation, 2024].

$$d = 2R \arcsin \left(\sqrt{\sin^2 \left(\frac{\phi_0 - \phi_1}{2} \right) + \cos(\phi_0) \cos(\phi_1) \sin^2 \left(\frac{\lambda_0 - \lambda_1}{2} \right)} \right) \quad (1)$$

In the above, R is the radius of the Earth in [km], while $(\phi_0; \lambda_0)$ and $(\phi_1; \lambda_1)$ are the coordinates of the starting and ending locations in [rad], respectively. ϕ represents latitudes and λ longitudes. Note that the Earth is assumed to be a perfect sphere in this calculation and no adjustments are made for the flattening at the poles.

Another altered parameter is the route available in the FPL and OPSLOG. Instead of considering the route itself, different features were extracted from it, specifically the requested speed and flight level for cruise. These will have an impact on aircraft performance and are also linked to TOW. Note that it was also considered to add the Standard Instrument Departure (SID) and STandard Arrival Route (STAR), however these are airport-dependent and may lead to overfitting or bias in the model predictions. Therefore, they have been left out from the features.

Finally, it is important to state that when computing a flight's total estimated elapsed time, the airlines make use of weather predictions along the route, especially regarding head or tail winds. These may have a significant impact on the flight duration, so although cruise weather forecasts are not taken into consideration as separate features in this study (as is done with TAF reports at the destination aerodromes), they are still accounted for along the cruise part of the flight via this feature.

Table 2: Description of features used for training.

Dataset	Feature	Description	Type	Units	Encoding
OPSLOG	great_circle_distance_ADEP_ADES	great circle distance between aerodromes of departure and destination	numerical	km	-
	great_circle_distance_ADES_ALTRNT1	great circle distance between aerodromes of destination and alternate	numerical	km	-
	AOARCID	aircraft operating agency ICAO ID	categorical	-	ordinal
	ARCTYP	aircraft type ICAO ID	categorical	-	ordinal
	EOBT	estimated off-block time	numerical	-	datetime cyclical
	TAXITIME	taxi time (taxi before take-off)	numerical	s	-
	TTLEET	total estimated elapsed time (flight duration)	numerical	min	-
	RFL	requested flight level	numerical	FL	-
FPL	SPEED	requested speed	numerical	kts	-
	ft_rvr_val	runway visibility range	numerical	m	-
	airac_cycl	AIRAC cycle	numerical	-	-
	ft_etot	estimated take-off time	numerical	-	datetime cyclical
	ft_eta	estimated time of arrival	numerical	-	datetime cyclical
	ft_f_rte_len	length of the route	numerical	nm	-
TAF	visibility_cavok	clouds and visibility ok	categorical	-	ordinal
	visibility_distance	visibility distance	numerical	m	-
	clouds_height	clouds ceiling height value	numerical	m	-
	clouds_amount	clouds amount	numerical	-	-
	wind_speed	mean wind speed	numerical	m/s	-
	wind_gust	wind gust speed	numerical	m/s	-
	wind_compass	mean wind direction	categorical	-	ordinal
	time	time of TAF report creation	numerical	-	-
	validity_start_time	start time of TAF report validity	numerical	-	-
	validity_end_time	end time of TAF report validity	numerical	-	-
	precipitation	presence of precipitation	categorical	-	one-hot
	obscuracion	presence of obscuracion	categorical	-	one-hot
	other	presence of extreme weather events (tornado, volcanic ash, etc.)	categorical	-	one-hot
	thunderstorms	presence of thunderstorms	categorical	-	one-hot
	freezing	presence of freezing	categorical	-	one-hot
	snow	presence of snow	categorical	-	one-hot
	clouds	presence of clouds	categorical	-	one-hot
indicator	trend forecasts indicator	categorical	-	ordinal	
probability	trend forecasts associated probability	numerical	%	-	

3.2 Features Encoding

As in all regression problems, all parameters must be numerical. For this reason, all categorical features must be encoded, including some numerical features given in "datetime" format. The type of encoding selected for each feature is indicated in the last column of Table 2.

For categorical features, ordinal encoding was used. The latter assigns a unique integer value to each category, according to an order defined by the number of occurrences of each category, in descending order. This type of encoding was selected to instore some meaning and order to the features, rather than simply attributing random numbers to each category.

One-hot encoding was also used for categorical features that presented a binary distribution, namely those oscillating between "True" and "False" values. One-hot encoding, also called dummy encoding, consists in creating a binary column for each category in the categorical variable. Such encoding is acceptable for features that do not present a large number of categories, as is the case for the ones selected. Note that it may also reduce the training speed of the model if too many new columns are created in the training dataset.

Finally, after categorical features have been dealt with, one last numerical feature type remains to be encoded: timestamps. For these, datetime cyclical encoding is used. The latter transforms datetime features into separate date and time periods to preserve the cyclical significance. Specifically, after selecting a time period from a feature value (seconds, minutes, day, month, etc.), it is converted to the corresponding number of seconds t . This allows to account for leap seconds and leap years. Next, using Equation 2, t is encoded into two numerical features.

$$\begin{cases} \sin\left(\frac{2\pi \cdot t}{T}\right) \\ \cos\left(\frac{2\pi \cdot t}{T}\right) \end{cases} \quad (2)$$

3.3 Machine Learning Algorithms

Two main types of algorithms were selected based on literature findings, amounting to a total of four algorithms: GBDTs, LightGBM, XGBoost, and Random Forests, where LightGBM and XGBoost are optimised implementations of GBDTs. Such algorithms are classified as "ensembles", meaning they are composed of many other models - in this case, decision trees. Ensembles are capable of boosting the performance of the underlying model [Géron, 2022].

Regardless of the algorithm, the goal in ML regression problems is to predict a target variable y , in this case the aircraft TOW, from a vector of features x listed in Table 2. In simple terms, this can be viewed as a regression problem, where the objective is to learn a function f such that $y = f(x)$. Since this is not feasible mathematically, the loss function is introduced. The latter quantifies the difference between the model's output predictions and the actual target values in the training dataset. In essence, it measures the model's performance. For the purpose of this research, the Mean Absolute Percentage Error (MAPE), depicted by Equation 3, was selected as loss function. When optimising a model with MAPE, the goal is to minimise this metric during the training process. This is achieved by tuning hyperparameters, such as the tree structure and learning rate.

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (3)$$

In the above, n is the number of samples, y_i is the actual target value from the training dataset, and \hat{y}_i is the predicted output value given by the model.

Since all algorithms selected are based on decision trees, it is important to introduce this notion. Decision trees are similar to flowcharts and are made up of decision nodes, leaf nodes, and branches. They are read from top to bottom and the first decision node (at the top) is called the root node. It partitions the input features recursively based on their value. Each subsequent decision node denotes an input feature of the dataset. Each branch and leaf node of the decision tree corresponds to a decision and its associated outcome, respectively.

3.3.1 Gradient Boosting Decision Trees

In gradient boosting, new decision trees are added sequentially to correct the residual error of the existing model, explaining the term *boosting* [Brownlee, 2016]. This algorithm typically uses shallow trees as weak learners and combines them through boosting to create a strong predictive model. Each decision tree is created using a greedy search procedure to select split points that best minimise the loss function. When referring to GBDTs, this paper specifically refers to Scikit-Learn's *Gradient Boosting Regressor* [Developers, 2024a]. This is an implementation of the "original" algorithm that is also the baseline for XGBoost and LightGBM.

The working principle of GBDTs consists of five main steps. The first step is to calculate the arithmetic mean of the target values x_i in the training dataset, in this case the TOWs. This is summarised in Equation 4, where n is the number of samples in the dataset.

$$\text{mean} = \frac{\sum_{i=1}^n x_i}{n} \quad (4)$$

In the second step, the residuals are calculated for each sample using Equation 5. Note that for the very first decision tree, the predicted value is the mean calculated in the previous step.

$$\text{residual} = \text{actual value} - \text{predicted value} \quad (5)$$

The third step is to populate the first decision tree with the computed residuals. When there are more residuals than leaf nodes, there will be more than one residual in the same leaf node. In such cases, the mean of the residuals in each leaf is computed, reducing the tree to single-valued leaf nodes. Notice that the goal of the decision tree is to predict the residuals, not the desired output label (TOW) itself!

The fourth step consists in assigning an updated residual value to each data sample, through the means of the newly-populated decision tree. Using these, one can compute the predicted value of the output label (TOW) for each data sample i . This is done as step four of the algorithm using Equation 6, where m is the number of iterations and α is the learning rate.

$$(\text{predicted value})_{m+1} = (\text{predicted value})_m + \alpha \cdot (\text{residual predicted by decision tree})_m \quad (6)$$

The learning rate is a hyperparameter introduced to prevent overfitting the model. Previous implementations of gradient boosting regression have shown that taking incremental steps towards the solution introduces bias, that is a deviation in the expected results. This achieves a lower overall variance, leading to more accurate predictions for samples outside of the training data. By introducing the learning rate, the model is forced to iterate more times and thus use more decision trees, which allows for the implementation of such an incremental

approach towards the final solution [Pal, 2020]. The learning rates explored in this study are 0.001, 0.01, 0.1, and 0.2.

The fifth step is to compute the new residuals once again, yet this time the predicted values from Equation 4 actually correspond to the ones from step four. The new residuals will be used for populating the second decision tree of the model, as explained in step three.

Finally, steps three to five are repeated m times, until the number of iterations allows to reach an acceptable level for prediction error, namely the loss function from Equation 3. Once the model is trained, the entire sequence of decision trees can be used to make a final prediction of the output label(s). The final computation for predicting the latter is given by Equation 7.

$$\text{prediction} = \text{average value} + \sum_{j=1}^m \alpha \cdot (\text{residual predicted by decision tree})_m \quad (7)$$

3.3.2 XGBoost

XGBoost stands for "Extreme Gradient Boosting" and is an optimised version of Scikit-Learn's *Gradient Boosting Regressor* [Developers, 2022]. Also based on GBDTs, XGBoost introduces several improvements compared to the traditional algorithm.

First, by using parallel and distributed computing, the training time is significantly reduced. As the construction of trees is performed in parallel, XGBoost is suited for large datasets and complex models, making efficient use of available computational resources. This is also enabled by pruning: a technique in ML that reduces the size of decision trees by removing sections of the tree that provide weak loss function values [Soyoung, 2024].

Another benefit of this algorithm is the early stopping criteria definition. This allows to stop the training process when further iterations no longer lead to improvements in loss function. Not only does this prevent overfitting, it also halts unnecessary computations, leading to reduced training time.

Finally, XGBoost is highly flexible in terms of hyperparameters tuning, which can be customised to achieve better performance.

3.3.3 LightGBM

LightGBM is an optimised version of XGBoost, developed by Microsoft. It uses the same tree-based gradient boosting principle described in section 3.3.1 with improved training speed and efficiency [Corporation, 2023].

The major improvement of LightGBM is the use of histograms during training, introducing the concept of histogram-based algorithms. In the latter, feature values are distributed across a series of discrete intervals to the feature, called bins. Bins are created for each feature, after which the bin that allows for the best loss function reduction is selected. This is the explanation behind the improved training speed of LightGBM over Scikit-Learn's *Gradient Boosting regressor*: "instead of creating a separate bin for each unique value of a feature, LightGBM creates a histogram of the feature values and then selects the best split based on the histogram" [Keldenich, 2023]. This enables LightGBM models to be trained on GPU and multiple cores, supporting parallel and distributed training.

Another important change from all the other gradient boosting algorithms, like XGBoost and Scikit-Learn's *Gradient Boosting Regressor*, is the way that trees are constructed during iterations. Contrarily to the level-wise tree growth strategy, shown in Figure 3, LightGBM uses a leaf-wise tree growth strategy illustrated in Figure 4.

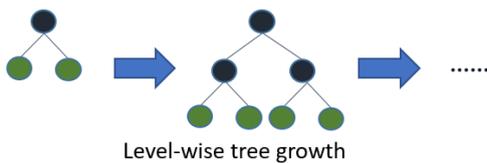


Figure 3: Level-wise tree growth strategy [Joseph, 2020].

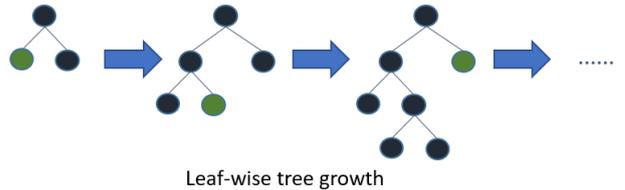


Figure 4: Leaf-wise tree growth strategy [Joseph, 2020].

This strategy identifies the leaf which yields a minimum loss function, and splits that leaf only, discarding the rest of the leaves at the same level. The leaf-wise tree growth therefore results in an asymmetrical tree where subsequent splitting can proceed, yet only on one side of the tree [Joseph, 2020]. This can also help speed up training, reduce memory usage, and improve model accuracy.

While this tends to achieve lower loss function values compared to the level-wise tree growth strategy, it also tends to overfit the data, especially for small datasets. This is why LightGBM is particularly recommended for large-scale data. On the other hand, the level-wise growth acts like a stopping mechanism for small datasets, restricting the complexity of the tree.

3.3.4 Random Forests

Random Forests follow a different approach to GBDTs, although they are still based on the same weak learners. Similarly to GBDTs, when referring to Random Forests, this paper specifically refers to Scikit-Learn’s *Random Forest Regressor* [Developers, 2024b]. This is once again an implementation of the "original" algorithm.

Random Forests rely on a bagging technique. Contrary to the boosting technique used by GBDTs, the trees in Random Forests run in parallel and there is strictly no interaction between them throughout the tree-building process [Chakure, 2023]. The working principle is shown in Figure 5.

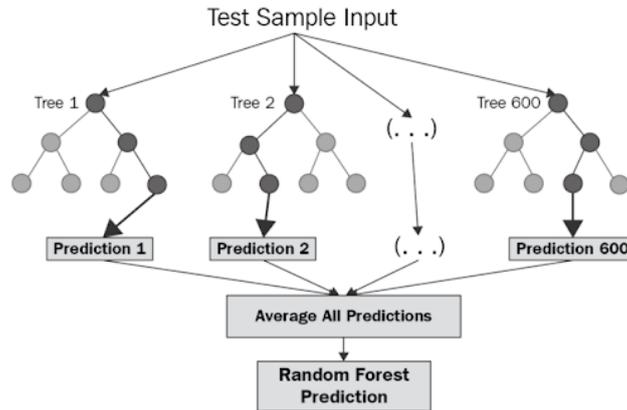


Figure 5: Random Forests working principle overview [Chakure, 2023].

A Random Forest is made of many decision trees trained on random sub-samples of the features. The predictions of each tree are then averaged out to obtain the final model prediction. This improves accuracy and controls overfitting.

3.3.5 Hyperparameters Tuning

Table 3 lists the different hyperparameter values explored for each algorithm presented in this section. When the value is empty (denoted by "-"), the corresponding hyperparameter is not applicable for the algorithm in question.

Table 3: Hyperparameters search space for considered algorithms.

	Boosting stages	Maximum number of trees	Number of trees	Maximum depth of tree	Learning rate	Maximum tree leaves	Minimum samples per leaf	Early stopping rounds
GBDTs	100 500 1,000 2,500 5,000 7,500	-	-	3	0.001 0.01 0.1 0.2	-	-	-
XGBoost	-	500	-	3	0.2	-	-	4
LightGBM	-	75 100 150 200 500	-	unconstrained	0.001 0.01 0.1 0.2	31	10 30	4
Random Forest	-	-	100 500 1,000 2,500 5,000	6 17	-	-	10	-

4 Description of the Case Studies

Following the algorithm selection, it is important to determine the case studies to be considered for this research. Sections 4.1 and 4.2 present the main case study along with the aircraft types considered, respectively. Finally, the validation datasets are detailed in section 4.3.

4.1 Amsterdam Airport Schiphol (AMS)

Several choices were made to narrow down the research space to a more computationally feasible dataset. The first decision was to narrow down the flights considered to those departing and arriving at Amsterdam Airport Schiphol (AMS). This choice was done to simplify the analysis and have a first understanding of the results at one single airport, instead of the entire EUROCONTROL network at once. In this way, potential lagging aspects of the model could be identified, especially regarding features considered for training. As AMS is a large airport in terms of traffic volumes and passengers carried, it was deemed suitable for this study. Furthermore, it generally accommodates legacy carriers traffic such as KLM and Air France while also having a wide range of low cost aircraft operators, hence it provides a good mix of traffic types.

It is important to note that, although these airlines may operate at AMS, only the flights which have the TOW information in their FPLs will be considered, since the latter provides the target output value for each flight. Furthermore, from these flights, only those Scheduled (S) and following Instrument Flight Rules (IFR) are considered. Regarding time range, the oldest FPL in EUROCONTROL's database with associated TOW dates back to February 2022. For this reason, all flights scheduled starting February 2022 and up to the end of December 2023 are considered in this research, amounting to **122,379** flights at AMS. These were split using an 80-20% ratio between training and testing datasets, resulting in 97,639 and 24,740 flights respectively. Note that this was not done randomly. The train-test split was conducted on a daily basis such to guarantee a robust training and to not omit potential cyclical patterns hidden behind "datetime" features.

Following this split, the distribution of flights across aircraft operators and aircraft types is shown in Figures 6 and 7 for the training set, respectively. Note that in Figure 6 only the top 10 airlines with the highest amount of flights are plotted. Enlarged versions of these bar charts can be found in Appendix A.

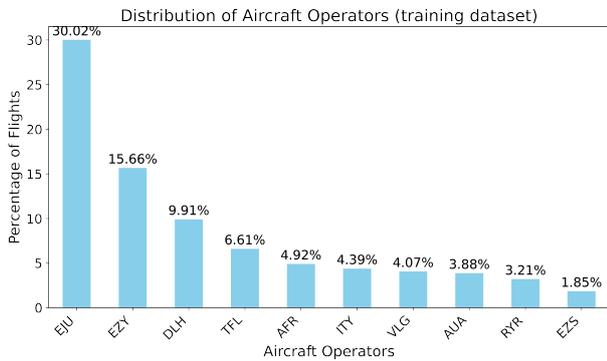


Figure 6: Distribution of aircraft operators across flights from training dataset (AMS). EJU and EZY both refer to easyJet - following Brexit, the airline set up a European-based subsidiary called easyJet Europe (EJU) based in Austria.

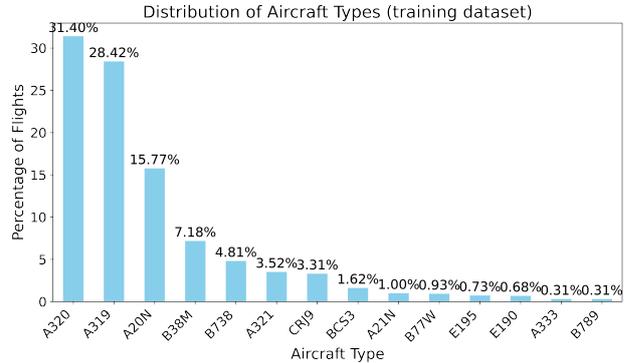


Figure 7: Distribution of aircraft types flown across flights from training dataset (AMS).

In the above, one notices that the main aircraft operator present in the training dataset is a low-cost carrier (EJU and EZY), amounting to almost 50% of the flights. The next airline in terms of flight count is Lufthansa, with almost 12% of flights from the training dataset, followed by TUI fly Netherlands with 8%. Interestingly, the second airline is a legacy carrier, while the third is a charter airline, giving a good variability for the training data, despite the large amount of low-cost carrier flights. Regarding Figure 7, almost 90% of most-flown aircraft types in the training dataset are classified as medium-range and narrow-body. As a consequence, it is expected that the algorithms will have better prediction accuracy for this type of aircraft.

4.2 Aircraft Types

Another important characteristic influencing aircraft TOW is the aircraft type itself. For this study, the top 15 aircraft types flown in the European Civil Aviation Conference (ECAC) region are considered. This also corresponds to the EUROCONTROL NM network. The rankings are made yearly, as shown in Table 4, although

there is not much difference between 2022 and 2023. The aircraft types across both years are generally the same, with some slight differences in ranking depending on traffic percentage.

Table 4: ECAC coverage report. Top 15 aircraft types flown with corresponding traffic percentage, years 2022 and 2023.

2022				2023			
Rank	ICAO ID	Traffic [%]	Cumulative traffic [%]	Rank	ICAO ID	Traffic [%]	Cumulative traffic [%]
1	B738	18.63	18.63	1	B738	18.11	18.11
2	A320	15.37	33.99	2	A320	15.48	33.59
3	A319	5.60	39.59	3	A20N	5.63	39.22
4	A20N	5.01	44.60	4	A319	5.37	44.59
5	A321	4.03	48.63	5	A21N	4.40	48.99
6	A21N	3.22	51.85	6	A321	4.29	53.28
7	B38M	2.93	54.78	7	B38M	3.86	57.14
8	AT76	2.30	57.08	8	AT76	2.68	59.83
9	E190	2.28	59.36	9	E190	2.44	62.26
10	B77W	1.77	61.14	10	B77W	1.87	64.13
11	B789	1.47	62.61	11	B789	1.53	65.66
12	A333	1.40	64.01	12	A333	1.37	67.03
13	E195	1.36	65.37	13	E195	1.36	68.40
14	CRJ9	1.22	66.59	14	BCS3	1.34	69.73
15	AT75	1.09	67.68	15	CRJ9	1.10	70.83

When comparing the traffic percentages per aircraft type from Table 4 with those from Figure 7 the ranking is not the same. Once again, the actual ranking does not determine the aircraft types distribution in the training dataset, this is solely attributed to the presence of TOW. Nevertheless it is similar, besides the B738 which ranks first in Table 4.

4.3 Validation Datasets

While the ML model is built on AMS data, two more airports are considered for validation purposes: Paris - Charles de Gaulle Airport (CDG) and Brussels South Charleroi Airport (CRL). CDG was selected to test the trained model on another airport with similar traffic volumes and size, both being major international hubs and some of the busiest airports in Europe. Furthermore both airports support a majority of legacy carrier operations, with KLM at AMS and Air France at CDG. On the other hand, CRL was chosen for its difference in size and operated flights, in order to analyse the model’s applicability to a completely different traffic mix. CRL is known for its low-cost carriers operations, of which Ryanair is one of the most important players.

Once again, these datasets contain flights departing from and arriving at the airports in question, for which TOW data is available. The aircraft types listed in Table 4 remain valid as well as the time period of 2022-2023 for appropriate validation and comparison with the original AMS testing dataset results. The CDG and CRL datasets amount to **320,032** flights and **54,788** flights, respectively.

The distributions of aircraft operators are shown in Figures 8 and 9 for CDG and CRL respectively. Note that enlarged versions of all bar charts displayed in this section can also be found in Appendix A.

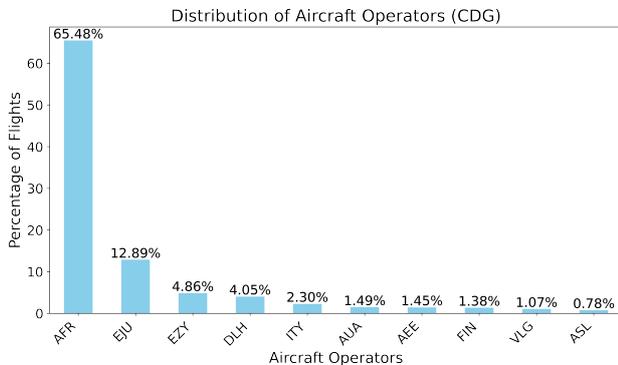


Figure 8: Distribution of aircraft operators across flights from validation dataset (CDG).

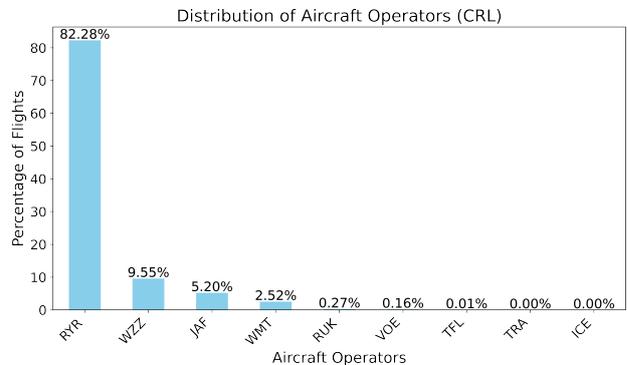


Figure 9: Distribution of aircraft operators across flights from validation dataset (CRL).

While the top 10 airlines are plotted for CDG, all airlines are plotted for CRL due to the distribution of the dataset, with Ryanair amounting to more than 80% of the flights. On the other hand, Figure 8 shows a promising distribution for CDG. As almost 70% of the flights are operated by Air France, a legacy carrier, this

dataset will serve as a good baseline for validation, since most of the flights in the training dataset are operated by a low-cost carrier (EJU and EZY).

Last but not least, the most-flown aircraft types of each validation dataset are given in Figures 10 and 11.

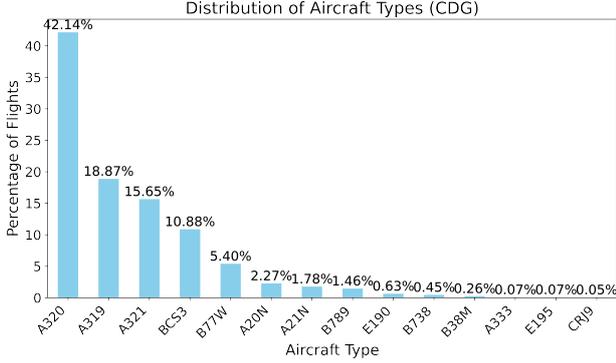


Figure 10: Distribution of aircraft types flown across flights from validation dataset (CDG).

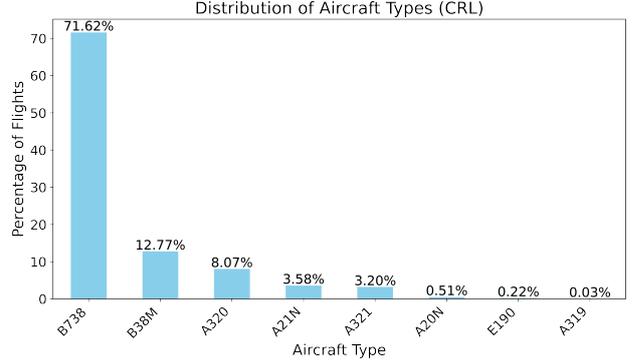


Figure 11: Distribution of aircraft types flown across flights from validation dataset (CRL).

Although the aircraft types distribution of CDG may be similar to the training dataset, that of CRL is not. More than 70% of flown aircraft are B738, while these correspond to less than 5% in Figure 7. Consequently, CRL will serve as a good baseline for validation regarding the aircraft types feature, in the same way that CDG serves as a good validation baseline for aircraft operators.

5 Results

This section presents the results of the analysis, starting with the selection of the most optimal ML algorithm trained with AMS data in section 5.1. section 5.2 discusses the applicability of the best performing model on CDG and CRL airports, as was described in section 4.3. Finally, an analysis is conducted on EUROCONTROL’s Enhanced Tactical Flow Management System (ETFMS) in section 5.3, specifically on the predicted ROC obtained for each flight if it were based on the TOW prediction model.

5.1 AMS Case Study

The ML algorithms presented in section 3.3 were trained with the same dataset from AMS, where a train-test split of 80-20% was followed. After training all the algorithms with 97,639 flights departing from and arriving at AMS, their performance could be analysed based on the testing dataset with 24,740 flights. Different error metrics were used to determine which ML algorithm performed best. These are listed in Table 5 along with the corresponding results.

Table 5: Error metrics overview across ML algorithms. Testing dataset used as reference.

Algorithm	Training time	MAPE [%]	MAE [kg]	R ² score
XGBoost	57s	2.59	1,629	0.9877
Random forest	6h 32m 30s	2.38	1,503	0.9851
GBDTs	12h 51m 36s	2.17	1,376	0.9907
LightGBM	4m 43s	2.18	1,373	0.9913

From this table, different observations can be made. Probably the most striking one is the training time, showing the significant improvements made to the "optimised" gradient boosting algorithms (XGBoost and LightGBM) - these are far from comparable with ScikitLearn’s *Gradient Boosting Regressor* (GBDTs). Nevertheless, in terms of error metrics, the latter does perform best in terms of MAPE, which is the selected metric for this study. On the other hand, LightGBM ranks almost identically to GBDTs, with better Mean Absolute Error (MAE) and coefficient of determination (R² score). The MAE and R² score are calculated as shown in Equation 8, where the symbols have the same meaning as the ones in Equation 3 and \bar{y} is the mean of the *actual* target values (TOW). The coefficient of determination measures how well the model predicts the TOW, with values reaching up to 1, 1 being a perfect fit.

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad \text{R}^2 = 1 - \frac{\sum_i^n (y_i - \hat{y}_i)^2}{\sum_i^n (y_i - \bar{y})^2} \quad (8)$$

The MAE is given in [kg] and provides a good understanding of the amount of divergence of the TOW predictions from the actual values. The results of this particular error metric show great potential for all algorithms, with less than 2 tonnes MAE. This is acceptable considering that TOW values range between 30 to 300 tonnes, so predictions given with ± 1.5 tonnes is not alarming.

Finally, the *Random Forest Regressor* and XGBoost performed worse than the previously discussed algorithms. These proved to give worse results for all error metrics with higher MAPE, MAE, and lower R^2 score.

Since the MAPE was used as the principal error metric and loss function, the scores of each algorithm were used for ranking them. Based on this, the GBDTs model was selected for further analysis and validation activities. Therefore, from now on, prediction results and further details all refer to the model built on ScikitLearn's *Gradient Boosting Regressor*.

The scatter plot of the regression achieved with GBDTs is shown in Figure 12, where the actual TOW values from the testing dataset are given on the horizontal axis and the TOW predictions generated by the model are given on the vertical axis. This graph gives a good illustration of the high R^2 score, with all data points located very close to the regression line and very few outliers.

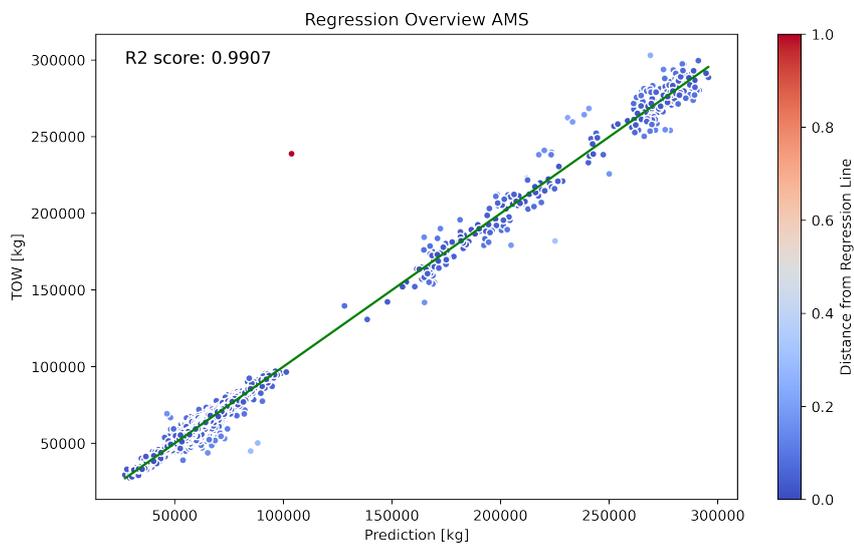


Figure 12: Scatter plot of GBDTs algorithm, testing dataset.

5.1.1 Error Distribution

The error distribution of the predictions done with the testing dataset are given in Figure 13.

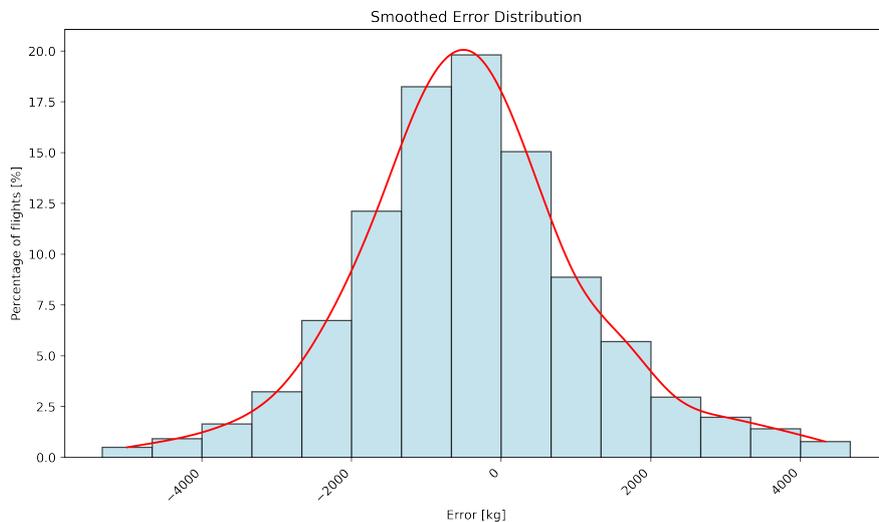


Figure 13: Error distribution of the testing dataset.

In this plot, the error is defined as the difference between predicted and actual values. The curve nicely depicts a normal distribution of the errors around 0, with limited spread. Statistical parameters defining the shape of this curve are given in Table 6. To reduce the effect of possible spurious outliers, error distribution is clipped at the 2nd and 98th percentiles.

Table 6: Error distribution statistics corresponding to Figure 13.

Min. (raw)	Min.	25th perc.	Median	75th perc.	90th perc.	Max.	Max. (raw)
-43,151	-4,714.7	-878.86	109.13	1,016.3	1,991.5	3,826.6	13,505
Average		-2.1139		Standard deviation		1,693.6	

An interesting observation from Table 6 is the comparison between minimum and maximum errors, which suggests that the model may tend to underestimate the predictions. Indeed, the minimum TOW prediction of -4.7 tonnes is more than one tonne (absolute value) over the maximum of 3.8 tonnes. For this reason, the testing dataset was studied to better understand the amount of over- and underestimations. Overall, it was found that 53.19% of flights' TOWs were underestimated and the remaining 46.81% were overestimated, which is not alarming. An overview of these over- and underestimations were also computed per aircraft type, as shown in Table 7, where there is no large deviation from the stated overall percentages. Note that the aircraft types are ordered according to the training dataset distribution (Figure 7) - from most to least flown.

Table 7: Over- and underestimations of TOW per aircraft type.

Aircraft ICAO ID	TOW predictions overestimated [%] (compared to FPL TOW)	TOW predictions underestimated [%] (compared to FPL TOW)
A320	46.56	53.44
A319	47.49	52.51
A20N	46.36	53.64
B38M	48.28	51.72
B738	45.19	54.81
A321	46.06	53.94
CRJ9	47.57	52.43
BCS3	41.21	58.79
A21N	45.23	54.77
B77W	47.81	52.19
E195	48.59	51.41
E190	40.12	59.88
A333	55.22	44.78
B789	57.53	42.47

While the error metrics for the *testing* dataset were given in Table 5, the errors on the *training* dataset were also computed for verification purposes. These are given below in Table 8. The MAPE and MAE remain low for the training dataset, but not excessively low to indicate overfitting, showing promising results for the model's capability of predicting TOW. The regression indicator, namely the R² score is also higher, depicting a better regression fit. This is expected when performing verification on the same dataset with which the model was trained and provides positive results.

Table 8: Error metrics comparison: training and testing datasets.

	MAPE [%]	MAE [kg]	R ² score
Training dataset	1.71	1,048	0.9965
Testing dataset	2.17	1,376	0.9907

Finally, one last interesting error visualisation is the MAPE grouped by aircraft type. Table 9 lists the average MAPE of the model for each aircraft type and for both training and testing dataset results.

Table 9: MAPE grouped by aircraft type, comparison between training and testing datasets.

Aircraft ICAO ID	MAPE [%] Testing dataset	MAPE [%] Training dataset
A320	2.40	1.93
A319	1.83	1.56
A20N	2.04	1.63
B38M	2.23	1.52
B738	2.07	1.43
A321	2.52	2.11
CRJ9	2.18	1.72
BCS3	2.48	2.09
A21N	2.20	1.19
B77W	2.57	0.63
E195	3.33	2.25
E190	3.98	2.38
A333	3.15	1.21
B789	3.14	0.64

Comparing this table with the aircraft distribution of the training dataset from Figure 7, the results are consistent. The more the model is trained with a specific aircraft type, the better it predicts the TOW for that aircraft type. For example, E190 and E195 (together) account for circa 1.5% of flights in the training dataset and they also have the highest MAPE.

5.1.2 Feature Importance

While aircraft type is expected to influence TOW predictions, other parameters may not be as obvious. For this reason, a feature importance analysis was conducted to rank all training features based on their contribution to the prediction of TOW (target output).

The results of this analysis are given in Table 10 for the top eight most-used and influencing features during training. A more extensive version listing all features can be found in Appendix B. These differ depending on the algorithms used, although the top four always remain the same, just ranked in a different order. Note that in Table 10 the order and importance values are solely given for the best performing model, namely GBDTs.

Table 10: Feature importance analysis

Feature	Importance [%]
requested speed	42.38
great circle distance between aerodromes of departure and destination	33.72
aircraft type ICAO ID	19.71
length of the route	1.57
runway visibility range	0.81
requested flight level	0.54
aircraft operating agency ICAO ID	0.45
total estimated elapsed time (flight duration)	0.14

This table shows that the requested speed in cruise is the parameter which has the highest influence on TOW predictions for this case study. This can be surprising, as one may tend to hypothesise that great circle distance between airports of departure and destination could have more influence on TOW due to fuel carried. The importance, given as percentages, suggests that the model output is essentially dictated by the top three training features, that is requested speed, great circle distance between aerodromes of departure and destination, and aircraft type.

These results are plausible for the following reasons. The requested speed at cruise affects the fuel consumption of the aircraft, so depending on this value, more or less fuel will be consumed. To reach higher cruise speeds, less fuel may be carried on board, affecting the overall value of TOW. Vice versa, when the requested speed is lower, the aircraft may accommodate a higher TOW. The great circle distance between the airports of departure and destination also influences TOW predictions. This suggests that there is a pattern between airport pairs and the fuel carried on-board to ensure that the aircraft reaches its destination. Finally, the aircraft type flown is an obvious factor, providing the model with a range of TOWs specific to each type.

5.1.3 Shapley Additive Explanations

The idea of Shapley Values originated in cooperative game theory, where they were used to fairly assign a player’s contribution to the game’s outcome. In ML, they capture the marginal contribution of each feature to the target output (TOW prediction). In essence, Shapley Values are determined by systematically perturbing input features and observing how these changes align with the model’s predictions. The Shapley Value of an input feature is then computed as the mean marginal contribution to the final model score [Gopinath, 2021].

Shapley Additive Explanations (SHAP) tend to provide more significant results than the feature importance analysis, and they are generally considered more reliable and interpretable. In feature importance, the most important features may not always be attributed the highest score, providing sufficient motivation for conducting the SHAP analysis [Rathi, 2022]. For example, in tree-based algorithms (such as those considered in this study) different scores may be given to equally important features solely based on their different levels of splitting.

Consequently, individual explanations were computed for the five most influential features, leading to the SHAP overview in Figure 14. Note that this was not done for the entire testing dataset; 2,000 sample flights were extracted randomly for Shapley Values calculations to reduce computational effort, without compromising results. Fortunately, the SHAP analysis is consistent with the feature importance from Table 10. The top three features are identical, although the Shapley Values suggest a different ranking, with aircraft type being the most influential feature, followed by great circle distance between airports of departure and destination, and requested speed.

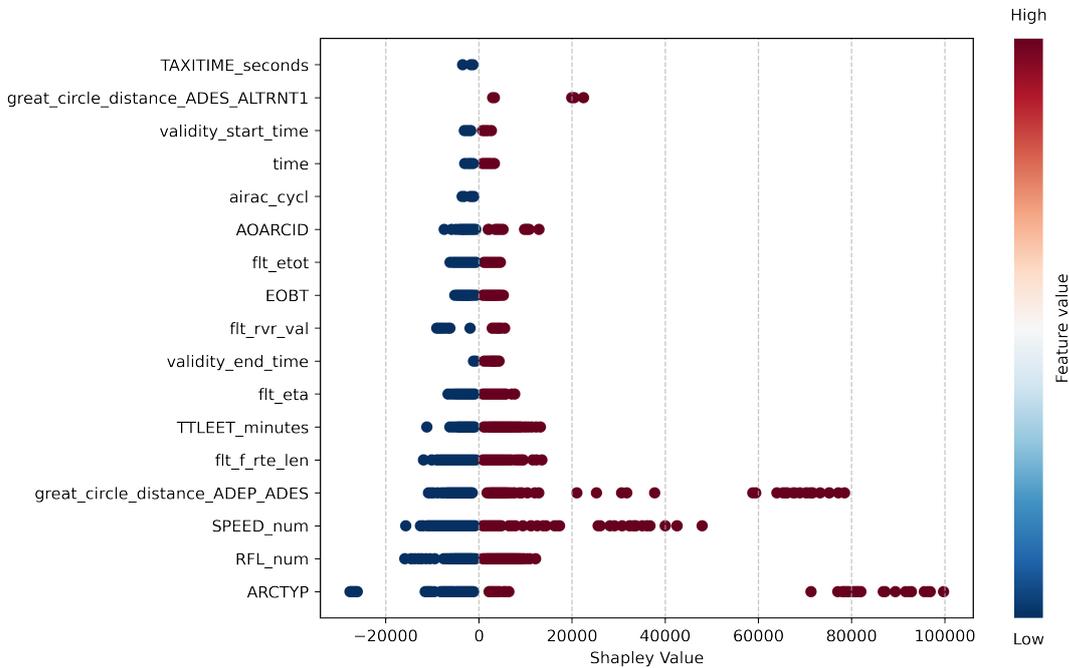


Figure 14: SHAP analysis results.

5.2 Extended Applications: CDG & CRL Airports

After performing the verification on both training and testing datasets and analysing the performance of the model, two more datasets were tested for validation purposes. To ensure that the model can be applied to other airports, it was tested on CDG and CRL airports, as explained in section 4.3. The error metrics are presented in Table 11.

Table 11: Error metrics comparison: CDG and CRL datasets (validation).

	MAPE [%]	MAE [kg]	R ² score
CDG dataset	4.07	4,032	0.9722
CRL dataset	3.41	2,237	0.4344

As expected, the MAPEs for CDG and CRL are higher than that for the testing dataset, although nothing *too* alarming. Most likely, this is due to the different distribution of flights in the dataset. Using the feature importance ranking and SHAP analysis overview from Table 10 and Figure 14, as well as Tables 6, 8 and 9, these values can be explained. As the distribution of the training dataset with which the model was fed consists

mainly of low-cost carrier flights (easyJet) it makes sense that the errors are smaller for CRL rather than CDG. The latter's traffic was mainly operated by AirFrance, a legacy carrier. Low-cost carriers tend to have lighter aircraft, and therefore lower values of TOW. This is due to limited fuel carried on-board for better aircraft performance and reduced costs, but also due to the constraints in luggages carried by passengers.

Regarding the coefficient of determination, the score is positive for CDG, yet the CRL dataset does not seem to fit with the predicted TOW very well. Scatter plots of both regressions are shown in Figures 15 and 16 for CDG and CRL, respectively. Note that for comparison purposes, the same range of predictions and TOWs was taken for the x- and y-axis of both plots, with a maximum value of 100 tonnes - leaving out remaining points. This could be due, once again, to the distributions of aircraft operators. With Ryanair having the highest traffic slice for CRL, the model is exposed to a completely different distribution of the data, with an airline that is barely present in the training dataset. Furthermore, the distribution of aircraft types at CRL in Figure 11 shows that almost 80% of the flights are operated by B738 aircraft, while the training dataset only contains about 4.5% of its traffic with this aircraft type (see Figure 7), suggesting that the target output distribution in the CRL dataset does not match the training data distribution.

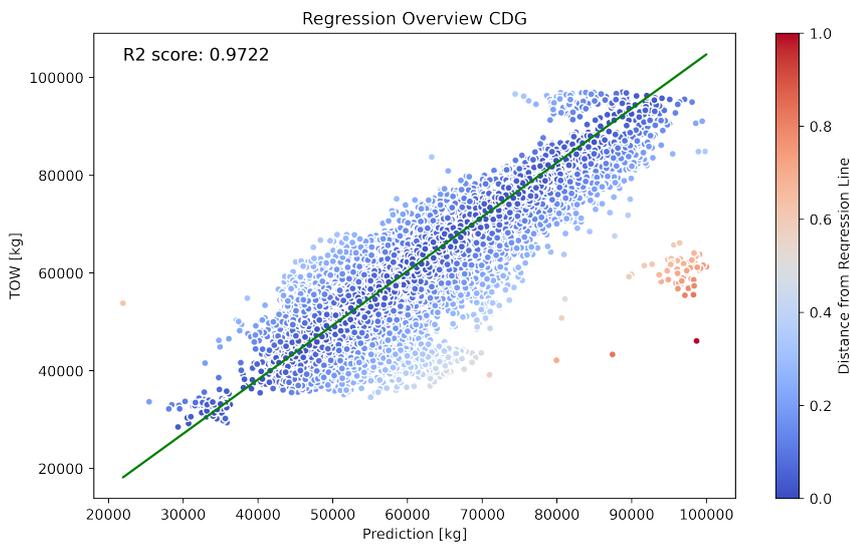


Figure 15: Scatter plot of GBDTs algorithm, CDG dataset.

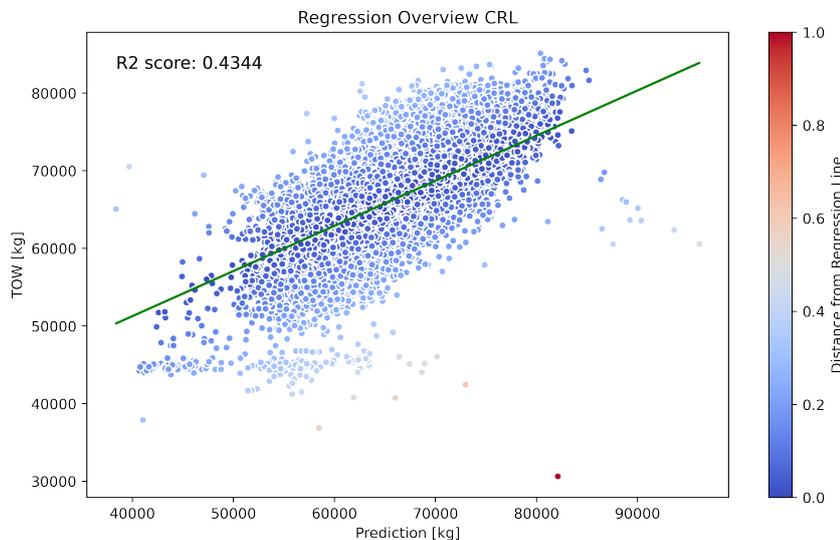


Figure 16: Scatter plot of GBDTs algorithm, CRL dataset.

The MAPE grouped by aircraft type is given in Table 12 for both CDG and CRL datasets. Note that the values missing for CRL are simply due to the aircraft types not being present in the dataset. As explained above, the errors can be explained by comparing the distribution of aircraft types across flights in the training dataset, shown in Figure 7, with the same distributions of the validation datasets, given in Figures 10 and 11. Once again, the aircraft with which the model has been trained more (those ranked higher in Figure 7) are associated with lower MAPE in Table 12.

Table 12: MAPE grouped by aircraft type, comparison between CDG and CRL datasets.

Aircraft ICAO ID	MAPE [%] CDG dataset	MAPE [%] CRL dataset
A320	3.14	5.33
A319	2.95	10.74
A20N	3.37	5.73
B38M	4.16	3.27
B738	2.77	2.93
A321	3.61	5.44
A333	4.22	-
B77W	11.16	-
B789	5.33	-
BCS3	5.22	-
CRJ9	5.23	-
E195	4.25	-
E190	31.99	36.06
A21N	3.83	5.15

5.3 Rate of Climb Validation

To test the potential of the model for a purpose other than TOW prediction, a validation exercise is performed on ETFMS. As stated in section 1, potential applications of the TOW prediction model include the improvement of flight planning and emissions calculations prior to take-off. In order to achieve this, more accurate trajectory predictions are needed. This section analyses the GBDTs model's impact on the latter. More specifically, the TOW predictions generated by the model are used to evaluate potential improvements in trajectory predictions via better associated ROCs in ETFMS.

ETFMS is a system used by EUROCONTROL's NMOC to calculate traffic demand in every sector of the NM area of operations, using the FPL information received from the aircraft operators [EUROCONTROL, 2023a]. To achieve this, each FPL received in the system is processed to generate an associated predicted trajectory, such that traffic demand can be estimated at network level. Trajectory predictions are built on the Base of Aircraft Data (BADA), an aircraft performance model developed at EUROCONTROL. However, ETFMS does not use the extensive version of BADA, but its standard Performance Table Files (PTFs). PTFs contain aircraft-specific information regarding their performance, namely ROC, cruise speed, rate of descent, and fuel consumption. Each table lists standardised values of these parameters per flight phase and flight level range. An example is given in Figure 17 for the A320 [EUROCONTROL, 1998].

The "mass levels" refer to TOWs, which are classified as low, nominal, or high. Fuel consumption during cruise as well as ROC during the climb phase have separate values associated to each TOW class. When TOW is *not* given in the FPL, the trajectory predictions are computed with nominal values, considering nominal TOW. However, when TOW *is* filed in the FPL, a linear interpolation is done between the three points with following "coordinates": low, nominal, and high ROC and their associated low, nominal, and high TOW. This is done at each flight level range to estimate the ROC corresponding to the provided TOW. The same procedure is followed for cruise fuel consumption.

Since TOW is closely associated to the climb phase, it was decided to analyse the ROC estimation potentials linked to the TOW prediction model described in this study. That means, for a FPL that does not have TOW associated, the ML model is used to predict its value. Next, a linear interpolation is done between low, nominal, and high values of TOW and ROC to compute the "predicted" ROC associated with the predicted TOW. This generates a list of ROCs, each corresponding to a flight level range. These computations are done using the testing dataset (at AMS) from section 4.1 with 24,740 flights. The same approach is also followed for the actual TOW values (provided in the FPLs) corresponding to each flight.

BADA PERFORMANCE FILE										98/03/12		
AC/Type: A320__			Last BADA Revision: 3.0			Source OPF File: 3.0			98/03/12			
			Source APF file: 3.0			98/03/12						
Speeds:		CAS (LO/HI)		Mach	Mass Levels [kg]			Temperature: ISA				
climb		- 250/300		0.78	low - 50160							
cruise		- 250/300		0.78	nominal - 62000			Max Alt. [ft]: 39000				
descent		- 250/300		0.78	high - 73500							
FL	CRUISE				CLIMB				DESCENT			
	TAS [kts]	fuel [kg/min]			TAS [kts]	ROCD [fpm]		fuel [kg/min]	TAS [kts]	ROCD [fpm]	fuel [kg/min]	
		lo	nom	hi		lo	nom	hi		nom	nom	
0					157	2180	2140	1890	129.4	142	1100	9.4
5					158	2160	2120	1870	128.1	143	1100	9.3
10					159	2150	2100	1850	126.8	149	1100	9.3
15					166	2250	2180	1910	125.5	160	1100	9.2
20					167	2230	2160	1890	124.2	192	1150	9.2
30	261	35.9	40.4	45.7	190	2620	2450	2140	121.6	230	1330	9.1
40	265	35.8	40.4	45.7	225	3100	2800	2440	119.1	233	1340	9.0
60	272	35.8	40.4	45.7	272	3690	3010	2520	114.1	240	1370	8.8
80	280	35.7	40.3	45.7	280	3540	2880	2400	109.1	280	1640	8.6
100	289	35.7	40.3	45.7	289	3390	2750	2270	104.1	289	1680	8.4
120	356	44.6	47.8	51.6	356	3260	2670	2230	99.4	356	2310	8.1
140	366	44.4	47.6	51.5	366	3060	2490	2070	94.6	366	2340	7.9
160	377	44.2	47.5	51.3	377	2860	2310	1900	89.8	377	2370	7.7
180	388	44.0	47.3	51.2	388	2650	2130	1740	85.1	388	2400	7.5
200	400	43.8	47.1	51.0	400	2430	1940	1560	80.5	400	2740	7.3
220	412	43.5	46.9	50.8	412	2220	1740	1390	76.0	412	2760	7.1
240	425	43.2	46.6	50.6	425	1990	1550	1210	71.6	425	2770	6.9
260	438	43.0	46.4	50.4	438	1770	1350	1030	67.2	438	2780	6.7
280	452	42.7	46.1	50.2	452	1540	1150	840	62.9	452	2790	6.5
300	459	41.5	45.1	49.3	459	1880	1360	950	58.6	459	3830	6.3
320	455	39.1	43.0	47.7	455	1690	1170	750	54.5	455	3620	6.1
340	451	37.0	41.3	46.4	451	1480	970	540	50.4	451	3430	5.8
360	447	35.2	40.0	45.6	447	1250	750	320	46.3	447	3270	5.6
380	447	33.8	39.0	44.1	447	940	470	70	42.4	447	2910	5.4
400	447	32.6	38.3	40.0	447	710	240	0	38.5	447	2840	5.2

Figure 17: Example PTF contents for A320 aircraft type [EUROCONTROL, 1998].

After generating the list of "predicted" ROCs, the idea is to compare them against their corresponding real values at each flight level range, after the flight has taken place. For validation purposes, the computed ROC values are compared against real ADS-B data gathered from FlightRadar24 (FR24). The FR24 trajectory consists of a sequence of 4D points, measured every 3-5 seconds, containing aircraft position and time at which the ADS-B report is generated. In terms of performance parameters, each point also contains vertical speed measures (corresponding to ROC), sourced directly from the aircraft's altimeter. However, for this exact reason, these speeds contain high levels of noise in their measurements, attributed to atmospheric fluctuations. This is a downside of this validation exercise; there is a significant risk that errors could be introduced in the ground truth data due to noise. Additionally, ADS-B data coverage and accuracy decays with altitude. This is mostly due to a higher presence of obstacles at low altitudes, making the measurements more susceptible to radio frequency interference.

To reduce noise and compute the sought comparison values of vertical speeds, the Savitzky-Golay filter is used for noise filtering. This filter uses convolution techniques to smooth the data points and improve precision without changing the trend of the signal. Essentially, linear least squares are used to fit successive subsets of neighbouring data points with a low-degree polynomial. An example is shown in Figure 18 for the climb phase of flight EJU39QN. The blue line corresponds to raw ROC data and the red line is the filtered curve after applying the Savitzky-Golay filter [Gallagher, 2020].

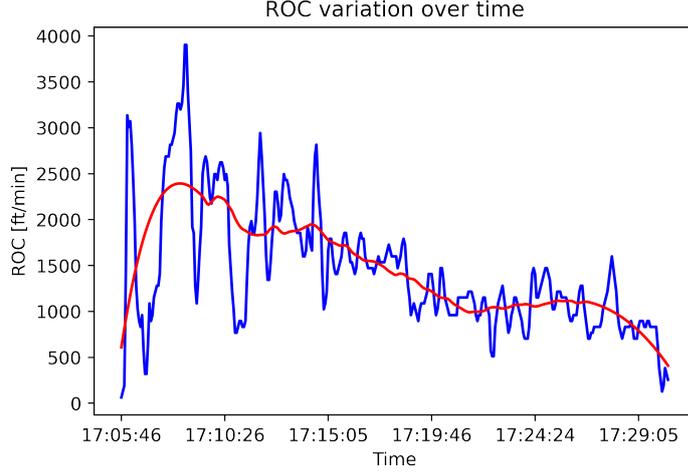


Figure 18: Noise filtering on FR24 vertical speed (ROC) data. Flight EJU39QN operated by easyJet from Tenerife South Airport (TFS) to AMS on the 12th of March 2022.

When applying this filter, it is important to select an appropriate window length. The latter determines the number of neighbouring data points in a subset over which the local polynomial approximation is performed. The window length W was computed based on the total number of data points N in the climbing segment by using a ratio defined in Equation 9. This was done for consistency purposes, since the number of data points in climb varies per flight. The higher the ratio, the higher the smoothing effect. A value of 0.3 was selected after repeated trial and error attempts.

$$\text{Ratio} = \frac{W}{N} = 0.3 \quad (9)$$

The last step in processing vertical speeds from FR24 is to compute the average of all filtered values for the corresponding flight level ranges of the ROC values. To summarise, three values are compared:

- *Predicted* ROC: value of ROC associated with the TOW predictions of the GBDTs model
- *Planned* ROC: value of ROC associated with the TOW values extracted from the FPLs
- *Nominal* ROC: value of ROC associated with the nominal TOW values from BADA PTFs, used when *no* TOW is given in the FPL

For clarity, these are illustrated in Figure 19, where the bottom cell depicts the ground truth ADS-B data.

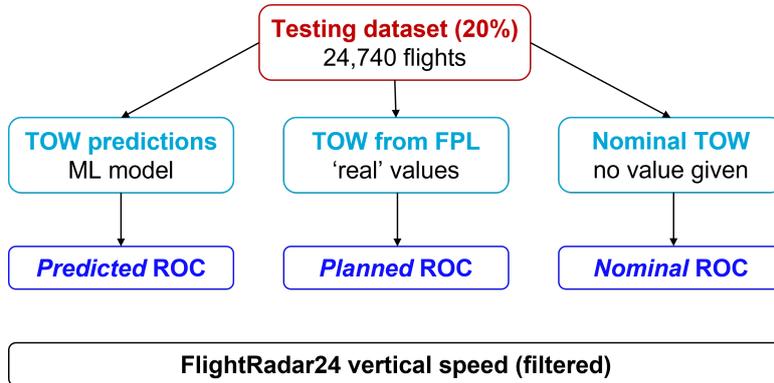


Figure 19: ROC validation flowchart. Overview of computed ROCs and their origin TOWs.

As ROC values vary per flight level range, each climbing segment of a flight will be associated with three *lists* of ROCs corresponding to the three types detailed above. The MAPE between the mean vertical speeds from FR24 and the ROCs listed above are computed for each altitude range. The general results are given in Table 13 per aircraft type, where the last two columns indicate the percentage improvement in MAPE achieved with the predicted ROC, compared to nominal and planned conditions respectively. Overall, the MAPE for predicted, planned, and nominal ROC amounts to 22.02%, 21.84%, and 22.00% respectively, showing that the

error between predicted ROC and flown ROC is the highest. However, before coming to any conclusions, it is important to understand what happens to the MAPE per aircraft type, and further, per altitude range.

Looking at Table 13, it is clear that the improvement compared to nominal ROC is higher for the aircraft types which are most present in the training dataset, and vice versa for those that are least present. Furthermore, the last column of this table highlights a deterioration in MAPE for most of the aircraft types, compared to planned ROC. This is expected since the planned conditions are based on the data with which the GBDTs model was trained, so the difference between the two cannot be positive. Nevertheless, the A333 and B789 show a slight improvement, which in this case, is attributed to the high variability of the noisy vertical speed data from FR24.

Table 13: MAPE overview for predicted, planned, and nominal ROC, compared to ground truth data (FR24).

Aircraft ICAO ID	MAPE [%] Predicted ROC	MAPE [%] Planned ROC	MAPE [%] Nominal ROC	Improvement [%] compared to <i>Nominal</i> conditions	Improvement [%] compared to <i>Planned</i> conditions
A320	19.98	19.71	21.39	1.41	-0.27
A319	22.34	22.29	21.89	-0.45	-0.05
A20N	19.79	19.62	21.37	1.58	-0.17
B38M	24.78	24.03	29.05	4.27	-0.75
B738	28.28	27.61	32.37	4.09	-0.66
A321	24.98	24.24	27.17	2.19	-0.73
A333	18.38	18.51	20.40	2.01	0.13
B77W	27.50	26.55	28.30	0.80	-0.95
B789	31.65	32.09	28.96	-2.69	0.44
BCS3	68.13	65.21	43.18	-24.95	-2.92
CRJ9	20.53	20.37	20.32	-0.21	-0.16
E190	34.96	27.58	24.08	-10.88	-7.37
E195	25.06	23.92	25.39	0.34	-1.13
A21N	30.42	29.33	32.03	1.60	-1.09

Figures 20 and 21 show the variation of MAPE per flight level range for the A320 and the E190, respectively. The first being the most-flown aircraft in the training dataset, while the second is one of the least flown. The plots of remaining aircraft types are provided in Appendix C, along with enlarged versions of Figures 20 and 21. In both graphs, there is a high MAPE at the extremities which, once again, is attributed to inaccuracies present in the flown ROC dataset (FR24). Since the extremities belong to sensitive phases, namely take-off and top of climb, the divergence in ROC combined with the noise induced by the altimeter is thought to lead to measurement errors. Furthermore, the absolute value of ROCs at these altitudes is very low, so even a small error can result in a large MAPE. Comparing both plots, it is clear that, overall, the A320 displays higher nominal ROC MAPE with lower MAPE achieved by the predicted and planned ROCs. Vice versa, the E190 displays higher MAPEs for the predicted and planned ROCs, while the nominal ROC MAPE is lower. This is attributed to the distribution of aircraft types of the training dataset from Figure 7. As was seen in Table 13, the more the aircraft type is present in the training dataset, the better the GBDTs model is at reducing the MAPE for predicted ROC.

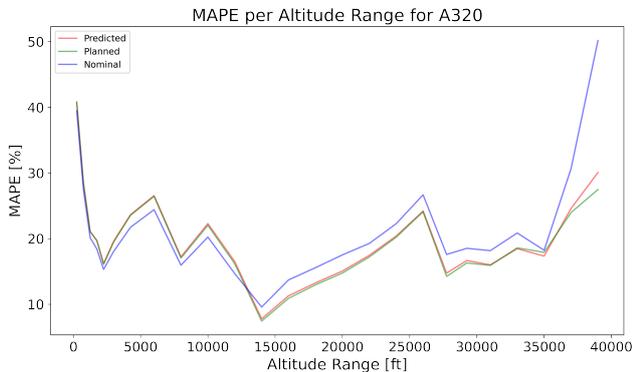


Figure 20: MAPE per flight level range for A320.

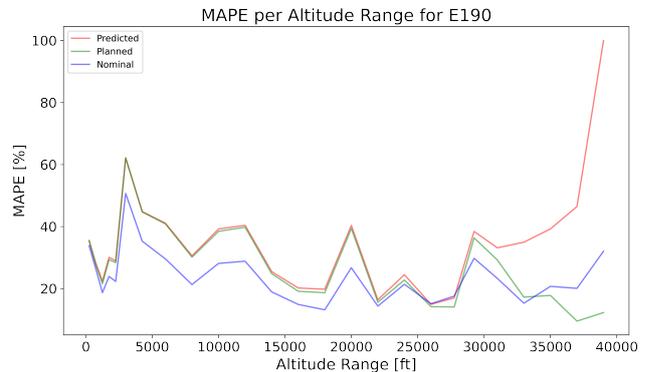


Figure 21: MAPE per flight level range for E190.

One last interesting behaviour can be visualised in Figure 20, specifically at altitudes below FL150, corresponding to the Terminal Manoeuvring Area (TMA). At these flight levels, even for the most-flown aircraft type in the training dataset (A320), the MAPE corresponding to nominal ROC is lower than that of the predicted and planned ROC. While strange and worrying, this can be explained with Table 14, where the percentage of flights for which the GBDTs model over- and underestimates the TOW compared to the flown aircraft type's nominal

TOW is given. Overall, the predicted TOW is underestimated for 73.02% of the flights, while only 26.98% are overestimated. This means that, compared to nominal TOW, the model tends to heavily underestimate the predictions. The reason is the general tendency of the model to underestimate predictions, previously discussed in Table 7, but also the important presence of low-cost carriers in the training dataset highlighted in Figure 6. The latter tend to fly on minimum fuel and constrain their passengers' luggage weight limits, leading to lower overall TOW. Looking back at Figure 17 and from conventional flight dynamics, when TOW is low ROC is high, meaning that indirectly, the TOW predictions tend to overestimate the corresponding ROC predictions.

Table 14: Over- and underestimations of TOW per aircraft type, compared to nominal TOW (BADA).

Aircraft ICAO ID	TOW predictions overestimated [%] (compared to nominal TOW)	TOW predictions underestimated [%] (compared to nominal TOW)
A320	28.02	71.98
A319	14.73	85.27
A20N	10.9	89.1
B38M	71.35	28.65
B738	68.09	31.91
A321	64.2	35.8
A333	25.37	74.63
B77W	2.19	97.81
B789	46.58	53.42
BCS3	0	100
CRJ9	26.28	73.72
E190	6.17	93.83
E195	57.63	42.37
A21N	34.02	65.98

The TMA is an area with many uncertainties, one of them being the way that aircraft *actually* fly. It is not uncommon for airlines to perform reduced thrust take-off and climb, allowing them to reduce fuel consumption and cut on costs. This would explain why the nominal ROC is closer to flown ROC than predicted (and planned) ROC(s). While this hypothesis is plausible, it can only be confirmed by the aircraft operators' cost indices, which are not available for this study. Nevertheless, since the MAPE above FL15 shows positive results achieved by the predicted ROC (reduced MAPE), the interpretation of the MAPEs behaviour in Figure 20 is reasonable.

Finally, it is important to discuss different aspects of the FR24 data including its noise and applicability to this validation exercise. While TOW impacts ROC, speed profiles and temperature also have an influence on the latter. The speed profile is composed of two Calibrated Air Speeds (CAS) and one Mach speed (M), namely (CAS₁, CAS₂, M). It is given for climb in Figure 17, yet these are *standard* values given for each aircraft type. When the real (FR24) trajectory is flown with a different speed profile, there is little to no hope of getting a good ROC fit between predicted and flown ROCs, even with an exact TOW prediction from the GBDTs model. Next, regarding temperature, the BADA PTFs are only valid at International Standard Atmosphere (ISA) conditions. The dataset considered contains both hot and cold weather flights, introducing atmospheric fluctuations. Therefore, it is also difficult to achieve significant improvement in ROC predictions. Lastly, when applying the Savitzky-Golay filter to FR24 vertical speeds, while smoothing out noise, part of the *real* data variability is also lost. Hence, it is reasonable to state that any significant improvement in predicted ROC is likely to have been diluted in the error induced by having different speed profiles and temperatures or via ground truth data filtering.

To conclude, although the model proved to improve the prediction of ROC, this was subjective to the aircraft type availability in the training dataset as well as the flight level. The latter depicted the operational context of the TMA with high uncertainties. Furthermore, one must keep in mind the high noise of the FR24 vertical speed data as well as its applicability to this validation exercise, especially regarding speed profiles and ISA conditions. Finally, while the improvements in MAPE between predicted and nominal ROC is found to be between 2-4% for the most-flown aircraft types in the training dataset, it is not enough to conclude that ROC predictions can be improved *solely* through the improvement of TOW predictions. Contrarily, the TMA uncertainties have shown that the cost index most likely plays an important role, while the feature importance analysis depicted the utmost importance of requested speed in cruise, thus the importance of speed profiles.

6 Discussion

This section provides a discussion of the results presented in section 5. First, the GBDTs model is compared with some of the previous studies reviewed in section 2. Then, several conditions of the model’s applicability are provided in section 6.2 along with potential improvements to consider for better results.

6.1 Comparison with Previous Studies

In order to assess the results of the GBDTs model, it is important to compare the obtained results against other studies’ results. For this purpose, [Alligier and Gianazza, 2018], [He et al., 2020], and [Sun et al., 2016]-[Sun et al., 2019] were evaluated against this work.

[Alligier and Gianazza, 2018] focused on improving aircraft climb prediction by better estimating operational factors, specifically the mass and speed profiles during climb. As there was no access to FPL data, hence TOW, the total energy model was used to reverse engineer the flown trajectories and build a dataset containing aircraft mass sequences. The trajectory data was ADS-B data extracted from The OpenSky Network [OpenSky, 2018]. Next, a stochastic gradient boosting tree algorithm was trained to predict sequences of aircraft masses. Since the Root Mean Square Error (RMSE) was provided in this study’s results, it was also computed for the GBDTs model using Equation 10, where the variables are the same as the ones in Equation 3.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}} \quad (10)$$

The results are listed in Table 15 for each aircraft type and the last column contains the percentage differences between RMSEs. For some aircraft types, where the RMSE difference is negative (A319, B738, and CRJ9), the GBDTs provide better predictions than the ones from [Alligier and Gianazza, 2018]. However, for all other aircraft types, this is not the case, suggesting that the use of trajectory data plays an important role in the quality of mass predictions. Generally, [Alligier and Gianazza, 2018] proved to achieve lower RMSEs per aircraft type, despite the synthetic nature of the training data. As proven by the feature importance and SHAP, requested cruise speed was a highly influencing factor for the GBDTs model. Even though TOW is a static parameter, it is part of a sequence of masses which are influenced by other trajectory factors, especially during climb. Consequently, completely discarding trajectory features may not capture the entirety of the picture, having an effect on the reliability of the TOW predictions.

Table 15: Comparison with [Alligier and Gianazza, 2018] results - RMSE between predicted TOW/mass and real TOW/estimated mass (with reverse engineering) used as reference values (for training).

Aircraft ICAO ID	RMSE [kg] (real TOW vs predicted TOW) Testing dataset	RMSE [kg] (estimated mass vs predicted mass) Reverse engineering method	RMSE difference [%]
A320	2,027	1,929	5.06
A319	1,442	2,362	-38.95
A20N	1,774	-	-
B38M	2,369	-	-
B738	2,075	2,508	-17.25
A321	2,300	2,212	3.98
A333	7,019	-	-
B77W	12,614	10,742	17.43
B789	7,668	-	-
BCS3	1,817	-	-
CRJ9	1,003	1,294	-22.49
E190	2,746	2,539	8.17
E195	3,407	2,126	60.23
A21N	2,409	-	-

[He et al., 2020] predicted the initial-climb aircraft mass using QAR data. While this is not the same as TOW, the methodology and nature of the target output is similar. Three different algorithms were tested: Multi-Layer Perceptron Neural Network (MLPNN), Ridge Regression (RR), and Decision Tree Regression (DTR). The results are given in Figure 22 for two training scenarios: single aircraft and all aircraft.

Aircraft number	Dataset (number of flights)	Metrics	MLPNN	RR	DTR
Single aircraft	Training (114) & Validation (38)	MAPE (%)	0.11	1.70	0.47
		NRMSD	0.005	0.089	0.030
		R ²	0.999	0.749	0.972
	Test (38)	MAPE (%)	1.17	1.90	2.46
		NRMSD	0.084	0.118	0.193
		R ²	0.869	0.738	0.295
All aircrafts	Training (2088) & Validation (696)	MAPE (%)	0.30	1.75	0.63
		NRMSD	0.010	0.063	0.025
		R ²	0.994	0.769	0.964
	Test (696)	MAPE (%)	0.61	1.72	1.48
		NRMSD	0.033	0.088	0.082
		R ²	0.970	0.782	0.808

Figure 22: Error metrics results from [He et al., 2020].

Compared with the 1.71% and 2.17% MAPE of the GBDTs model’s training and testing datasets, respectively, most algorithms used in [He et al., 2020] perform better. When trained with flights from a single aircraft, the large difference between training and testing MAPE of the MLPNN and DTR algorithms indicates overfitting. However, when considering all aircraft in the dataset, the MLPNN performs best, with merely 0.61% MAPE on the testing dataset. The other two algorithms also proved to achieve lower MAPE than the GBDTs model. Nevertheless, it is important to note that QAR data is not comparable to the features used by the model developed in this study. First, the data used by [He et al., 2020] comes from one single airline, which has comparable trends, and especially, uses the same cost index. Additionally, the dataset with which the models are trained is significantly smaller than the training dataset detailed in section 4.1. Therefore, there is potentially less variability in the data considered, leading to lower MAPE. Nevertheless, this error is still very low, and it would be interesting to test the MLPNN algorithm for the current study.

Finally, [Sun et al., 2016]-[Sun et al., 2019] used runway ADS-B data and statistical methods to predict TOW. Once again, the total energy model was used to deduce the TOW from ADS-B data. While the TOW estimations were within the bounds of BADA TOW (low - high), no quantitative comparison was available, although some of the conclusions are applicable to the results of the GBDTs model. First, it is stated that using the total energy model for mass estimations introduces uncertainty due to multiple combinations of thrust and mass being able to satisfy the equation. The series of studies also confirm that it is common for aircraft to perform reduced thrust take-off and climb, and that knowledge of ratio of reduced thrust would be needed to improve TOW estimations. This was also stated in section 5.3 for the ROC validation exercise, explaining the reason for higher predicted ROC MAPE. Consequently, it is concluded that the level of correctness of aircraft mass estimations is dependent on the knowledge of thrust settings, which is related to the most important feature of the GBDTs model: requested cruise speed. Finally, due to lack of validation data (real TOW data), [Sun et al., 2016]-[Sun et al., 2019] relied on cross-validation between several models, so there is still an error margin in the predictions.

6.2 GBDTs Model Applicability and Improvements

Based on the comparisons made in section 6.1 and the results from section 5, a list of conditions for the applicability of the model can be drawn. Naturally, ML relies on data, giving high importance to the quality, distribution, and quantity of flights available in the training dataset. The GBDTs model was essentially trained with narrow-body medium-range aircraft, with a majority of the flights operated by low-cost carriers. Its general tendency of underestimating aircraft TOW is mostly attributed to these factors. On a positive note, the model’s behaviour was found to be independent of aerodromes of departure and destination, even though the MAPEs were found to be higher for CDG and CRL airport, however this was attributed to their traffic distribution. If one were to set conditions for applying the GBDTs model to another dataset, for example at another airport, the main ones would be to have sufficiently similar distributions of aircraft and aircraft operator types. The latter tend to have a significant effect on TOW due to luggage and fuel limitations (within safety bounds).

Regarding potential improvements of the model, several areas can be explored. The most straightforward one is training the model with more data. As the model proved to be independent of airports, it would be interesting to see its capabilities and MAPE when trained on the entirety of the European network, for which EUROCONTROL is responsible. It is expected that, incorporating more diverse data, especially in terms of aircraft and airline types, will improve the capabilities of the model and its applicability. In terms of features, an

important addition would be aircraft speed *intent*, capturing the selected settings in the cockpit and introducing more trajectory-based information.

Next, the MLPNN algorithm used in [He et al., 2020] proved to achieve lower MAPE than the GBDTs model, raising the question: *Could neural networks be a better alternative to gradient boosting?* The reality is that, once again, it all depends on the training dataset. Tree-based algorithms, such as the ones considered in this study, often outperform neural networks [Ye, 2020]. Their ability of capturing non-linear relationships and feature interaction is more effective. Already proven by the MLPNN in [He et al., 2020], neural networks tend to quickly overfit, especially when applied to small datasets. However, when the size of the dataset is significantly increased, neural networks tend to provide better prediction accuracy. Their complex structure enables them to capture complex patterns and automatically learn relevant features from raw data, which greatly facilitates data processing. All these aspects characterising neural networks show promising improvements when applied to very large datasets. To assess potential improvements of these algorithms, including MLPNN, one should also combine the previous area of exploration, namely the enlargement of the training dataset to European network-level. This would avoid overfitting as well as unnecessary usage of computational resources.

Regarding the operational context, specifically the implementation of FF-ICE, more accurate variables will be shared by aircraft operators, including more accurate climb trajectory predictions. Currently, because trajectory parameters are not considered among the features, neither the reduced thrust take-off and climb nor the corresponding cost index are captured by the model. Nevertheless, the requested cruise speed is found to be the most-influencing feature for TOW predictions, depicting the importance of performance and trajectory aspects. FF-ICE will provide a speed schedule defined by (CAS_1, CAS_2, M) , from which a more accurate requested cruise speed could be deduced. Alternatively, the reverse engineering approach could be used on provided FF-ICE climb trajectory predictions, to potentially better estimate TOW with the total energy model. This would expand the training dataset to other aircraft operators that do not share TOW data and could provide more insight regarding their operational trends.

7 Conclusions and Recommendations for Future Work

In this paper, different supervised learning algorithms were explored for the development of a ML-based TOW prediction tool at AMS. The model was trained solely on FPL and TAF data for weather integration. The study demonstrated the reliability of the approach along with several limitations regarding its applicability.

GBDTs proved to be the best algorithm for this particular application in terms of MAPE achieved, closely followed by LightGBM, with 2.17% and 2.18% respectively. The "winning" GBDTs model proved that (pre-)tactical TOW prediction, solely based on features available prior to take-off, is possible and reliable enough provided certain conditions are met. Feature importance revealed that the most influencing parameters, in order, were cruise requested speed, great circle distance between aerodromes of departure and destination, and aircraft type. Furthermore, the model proved to be independent of airports of departure and destination in terms of traffic volumes and passengers transported. Nevertheless, flight-specific parameters did have an impact.

The limitations of the model included its dependence on the distribution of AMS flights as well as its applicability to ROC and (therefore) trajectory predictions. When testing the model on CDG and CRL, similar- and different-sized airports, it was found that aircraft and airline *types* distribution influenced TOW predictions the most, but that size or similarity of the airport itself compared to AMS did not matter. Since training was essentially conducted with medium-range aircraft and low-cost carriers, these categories showed better prediction accuracy, limiting the model's applicability to the distribution of flight types in the training dataset. Next, the ROC validation proved that by using TOW predictions from the GBDTs model, although achieving between 2-4% improvement in ROC predictions (only for most-present aircraft types in the training dataset), it is not enough to state that improved TOW predictions enable significantly better accuracy of ROC predictions. This validation also highlighted the model's tendency of underestimating TOW, leading to overestimating ROC. In the TMA, this has an impact on the improvement of ROC predictions, since reduced thrust take-off and climb - often practiced by airlines for financial purposes - are not captured by the model, leading to higher discrepancies with ground truth data.

Several recommendations for future work can be listed to reduce the limitations of the TOW prediction model. First, it would be interesting to incorporate aircraft intent information (if available) in the features of the training dataset. As was seen in the ROC validation, airlines' tendencies are not captured by the model, limiting its applicability on ROC prediction. Secondly, training the model with more data is expected to improve its applicability to more airports, having a larger coverage of aircraft and airline types. Increasing the training dataset would also enable to test neural network algorithms, which showed better prediction accuracy for larger datasets. Next, other operational parameters could play a bigger role in achieving better ROC and trajectory predictions, and should be investigated for future work. Based on the feature importance analysis and previous studies' results, speed profiles and trajectory-based parameters showed promising benefits when the goal is

improving trajectory predictions. Finally, the FR24 ADS-B data may not be very well suited for the ROC validation exercise in terms of comparison with the ROCs generated through the ETFMS methodology. For more significant results, this validation exercise should be repeated with different ground truth data other than the noisy vertical speeds from FR24. A possibility would be to consider a subset of the FR24 data with conditions closer to the "ideal" conditions, that is flights for which the speed profiles in climb and the temperatures are close to the values of the BADA PTFs used in ETFMS. This would isolate the impact of TOW predictions on ROC predictions. An alternative could also be to use the trajectory data from EUROCONTROL's Current Tactical Flight Model (CTFM) within ETFMS to compute ROCs and compare them against the predicted values. CTFM is based on radar data, which shows more accurate measurements resolution compared to ADS-B data, especially at low altitudes.

References

- [Administration, 2004] Administration, F. A. (2004). Advisory circular.
- [Alligier and Gianazza, 2018] Alligier, R. and Gianazza, D. (2018). Learning aircraft operational factors to improve aircraft climb prediction: A large scale multi-airport study. *Transportation research. Part C, Emerging technologies*, 96:72–95.
- [Alligier et al., 2015] Alligier, R., Gianazza, D., and Durand, N. (2015). Machine learning and mass estimation methods for ground-based aircraft climb prediction. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3138–3149.
- [Boase, 2008] Boase, B. (2008). Geographical area for which the taf and metar are defined.
- [Brownlee, 2016] Brownlee, J. (2016). A gentle introduction to the gradient boosting algorithm for machine learning.
- [Centre, 2023] Centre, A. W. (2023). Taf overview.
- [Chakure, 2023] Chakure, A. (2023). Random forest regression in python explained.
- [Chati and Balakrishnan, 2017] Chati, Y. S. and Balakrishnan, H. (2017). Statistical modeling of aircraft takeoff weight.
- [Corporation, 2023] Corporation, M. (2023). Lightgbm documentation.
- [Corporation, 2024] Corporation, P. (2024). Great circle distance calculation.
- [Developers, 2024a] Developers, S.-L. (2024a). Gradient boosting regressor.
- [Developers, 2024b] Developers, S.-L. (2024b). Random forest regressor.
- [Developers, 2022] Developers, X. (2022). Xgboost tutorials.
- [EUROCONTROL, 1998] EUROCONTROL (1998). Aircraft performance summary tables for bada.
- [EUROCONTROL, 2023a] EUROCONTROL (2023a). Enhanced tactical flow management system.
- [EUROCONTROL, 2023b] EUROCONTROL (2023b). Flight plan filing and management.
- [EUROCONTROL, 2024] EUROCONTROL (2024). European aviation overview 2023.
- [flightrecorder.com, 2023] flightrecorder.com (2023). Quick access recorder (qar).
- [Gallagher, 2020] Gallagher, N. B. (2020). Savitzky-golay smoothing and differentiation filter.
- [Gopinath, 2021] Gopinath, D. (2021). The shapley value for ml models.
- [Géron, 2022] Géron, A. (2022). *Hands-On Machine Learning with Scikit-Learn, Keras, & TensorFlow*. O'Reilly Media, California, United States of America, 3rd ed. edition.
- [He et al., 2020] He, X., He, F., Zhu, X., and Li, L. (2020). Data-driven method for estimating aircraft mass from quick access recorder using aircraft dynamics and multilayer perceptron neural network. *ArXiv*, abs/2012.05907.
- [(ICAO), 2005] (ICAO), I. C. A. O. (2005). Rules of the air.

- [Joseph, 2020] Joseph, M. (2020). The gradient boosters iv: Lightgbm.
- [Keldenich, 2023] Keldenich, T. (2023). Lightgbm – what and how to use ? – best simple guide.
- [LVNL, 2023] LVNL (2023). Schiphol eaip.
- [OpenSky, 2018] OpenSky (2018). Publication data.
- [Pal, 2020] Pal, A. (2020). Gradient boosting trees for classification: A beginner’s guide.
- [Ramon Dalmau-Codina, 2024] Ramon Dalmau-Codina, L. G. (2024). Metafora.
- [Rathi, 2022] Rathi, P. (2022). A novel approach to feature importance — shapley additive explanations.
- [Skybrary, 2023a] Skybrary (2023a). Flight plan.
- [Skybrary, 2023b] Skybrary (2023b). Flight plan completion.
- [Skybrary, 2024] Skybrary (2024). Meteorological aerodrome report (metar).
- [Soyoung, 2024] Soyoung, L. (2024). Simple explanation of xgboost without complicated mathematics.
- [Stewart, 2019] Stewart, M. (2019). The actual difference between statistics and machine learning.
- [Sun et al., 2019] Sun, J., Blom, H. A. P., Ellerbroek, J., and Hoekstra, J. M. (2019). Particle filter for aircraft mass estimation and uncertainty modeling. *Transportation Research Part C: Emerging Technologies*.
- [Sun et al., 2016] Sun, J., Ellerbroek, J., and Hoekstra, J. (2016). Modeling and inferring aircraft takeoff mass from runway ads-b data.
- [Sun et al., 2017] Sun, J., Ellerbroek, J., and Hoekstra, J. (2017). Bayesian inference of aircraft initial mass.
- [Sun et al., 2018] Sun, J., Ellerbroek, J., and Hoekstra, J. M. (2018). Aircraft initial mass estimation using bayesian inference method. *Transportation Research Part C: Emerging Technologies*, 90:59–73.
- [Vouros et al., 2022] Vouros, G. A., Tranos, T., Blekas, K., and Santipantakis, G. (2022). Data-driven estimation of flights’ hidden parameters.
- [Ye, 2020] Ye, A. (2020). When and why tree-based models (often) outperform neural networks.

Appendices

A Appendix 1: Datasets Distributions (Enlarged)

This appendix provides enlarged versions of Figures 6 to 11 in Figures 23 to 28, respectively.

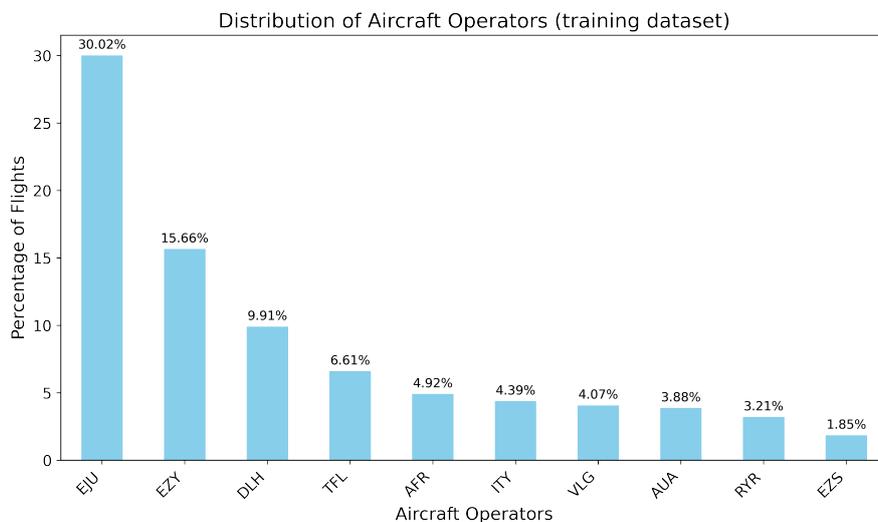


Figure 23: Distribution of aircraft operators across flights from training dataset (AMS, enlarged version). EJU and EZY both refer to easyJet - following Brexit, the airline set up a European-based subsidiary called easyJet Europe (EJU) based in Austria.

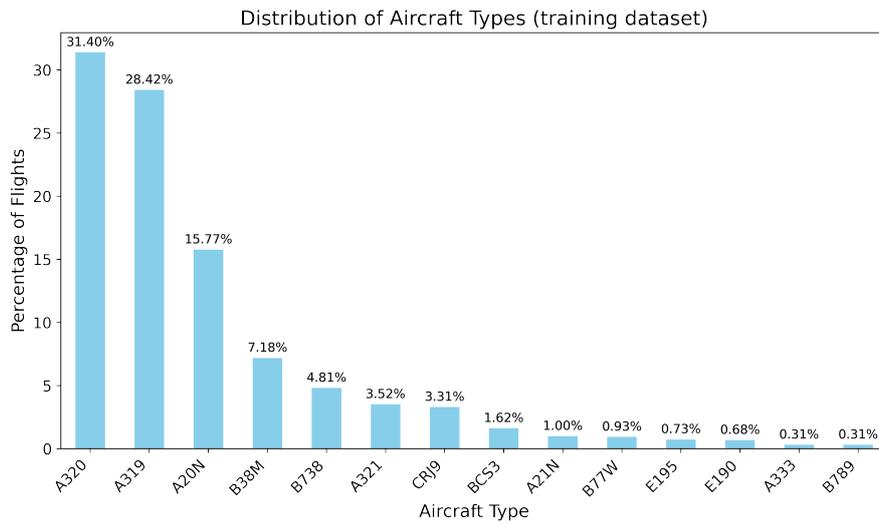


Figure 24: Distribution of aircraft types flown across flights from training dataset (AMS, enlarged version).

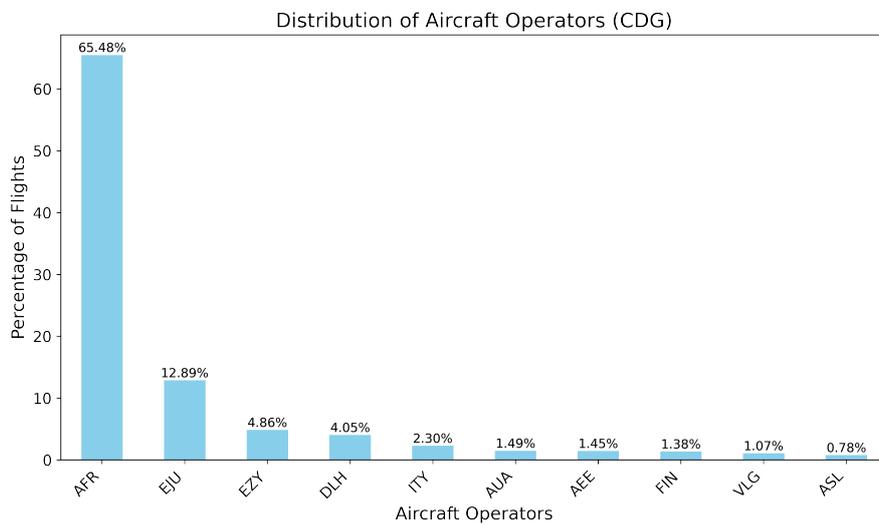


Figure 25: Distribution of aircraft operators across flights from validation dataset (CDG, enlarged version).

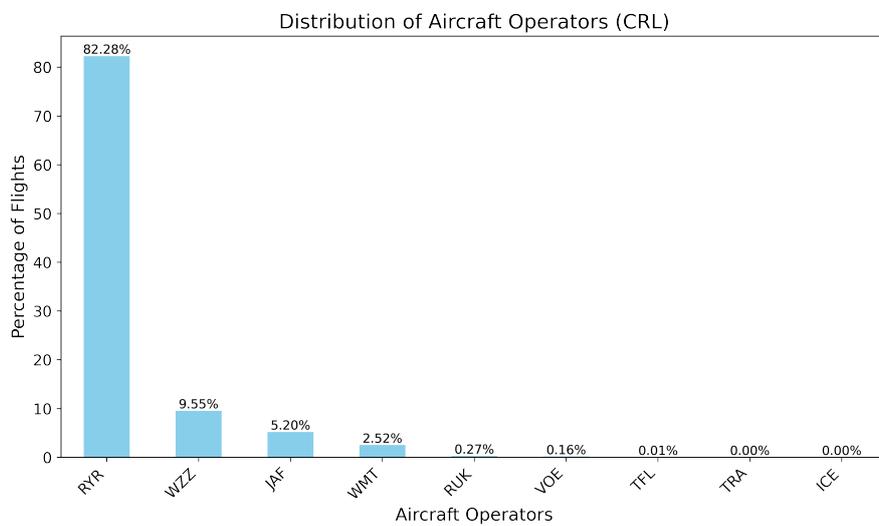


Figure 26: Distribution of aircraft operators across flights from validation dataset (CRL, enlarged version).

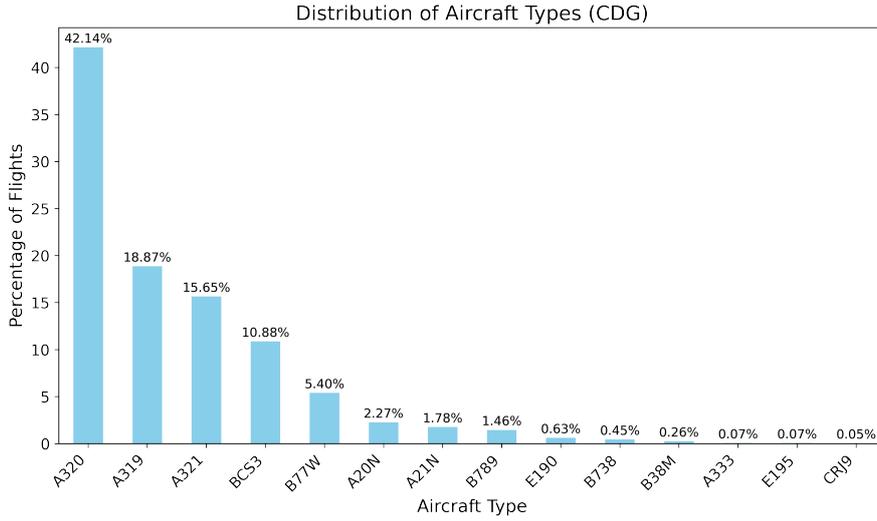


Figure 27: Distribution of aircraft types flown across flights from validation dataset (CDG, enlarged version).

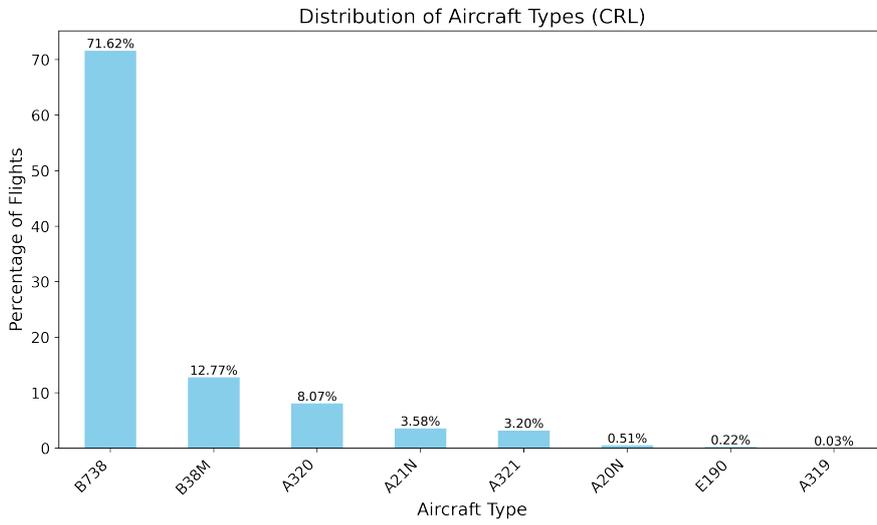


Figure 28: Distribution of aircraft types flown across flights from validation dataset (CRL, enlarged version).

B Appendix 2: Feature Importance (Extensive)

Table 16 provides the complete ranking stemming from the feature importance analysis.

Table 16: Feature importance analysis (extensive).

Feature	Importance [%]
SPEED_num	42.38
great_circle_distance_ADEP_ADES	33.72
ARCTYP [ordinal encoded (count)]	19.71
ft_f_rte_len	1.57
ft_rvr_val	0.81
RFL_num	0.54
AOARCID [ordinal encoded (count)]	0.45
TTLEET_minutes	0.14
TAXITIME_seconds	0.1
airac_cycl	0.07
great_circle_distance_ADES_ALTRNT1	0.07
ft_eta [week cycle (sin)]	0.03

ft_eta [day cycle (sin)]	0.03
ft_eta [day cycle (cos)]	0.02
EOBT [week cycle (sin)]	0.02
time [week cycle (cos)]	0.02
EOBT [day cycle (cos)]	0.01
time [week cycle (sin)]	0.01
ft_eta [week cycle (cos)]	0.01
ft_etot [day cycle (sin)]	0.01
ft_etot [day cycle (cos)]	0.01
time [hour cycle (cos)]	0.01
validity_start_time [week cycle (sin)]	0.01
validity_end_time [quarter cycle (sin)]	0.01
ft_eta [hour cycle (cos)]	0.01
time [quarter cycle (cos)]	0.01
ft_eta [quarter cycle (sin)]	0.01
ft_eta [minute cycle (sin)]	0.01
time [quarter cycle (sin)]	0.01
validity_start_time [day cycle (cos)]	0.01
ft_eta [month cycle (cos)]	0.01
ft_eta [hour cycle (sin)]	0.01
EOBT [day cycle (sin)]	0.01
ft_etot [week cycle (sin)]	0.01
validity_end_time [quarter cycle (cos)]	0.01
ft_eta [quarter cycle (cos)]	0.01
time [hour cycle (sin)]	0.01
time [month cycle (sin)]	0.01
validity_end_time [month cycle (cos)]	0.01
ft_etot [quarter cycle (sin)]	0.01
ft_eta [minute cycle (cos)]	0
ft_etot [hour cycle (cos)]	0
ft_etot [week cycle (cos)]	0
time [month cycle (cos)]	0
EOBT [week cycle (cos)]	0
validity_start_time [quarter cycle (sin)]	0
EOBT [month cycle (cos)]	0
EOBT [quarter cycle (sin)]	0
time [day cycle (sin)]	0
validity_end_time [week cycle (cos)]	0
time [day cycle (cos)]	0
ft_etot [quarter cycle (cos)]	0
validity_start_time [week cycle (cos)]	0
wind_compass [ordinal encoded (count)]	0
ft_eta [month cycle (sin)]	0
validity_end_time [week cycle (sin)]	0
wind_speed	0
validity_start_time [quarter cycle (cos)]	0
ft_etot [hour cycle (sin)]	0
ft_etot [month cycle (cos)]	0
EOBT [quarter cycle (cos)]	0
validity_end_time [month cycle (sin)]	0
ft_etot [month cycle (sin)]	0
EOBT [month cycle (sin)]	0
EOBT [hour cycle (sin)]	0
validity_start_time [month cycle (sin)]	0
visibility_cavok is True	0
validity_start_time [month cycle (cos)]	0
visibility_cavok is False	0
wind_gust	0

clouds_height	0
validity_end_time [day cycle (sin)]	0
visibility_distance	0
validity_start_time [day cycle (sin)]	0
validity_end_time [day cycle (cos)]	0
EOBT [hour cycle (cos)]	0
probability	0
other is False	0
other is True	0
clouds_amount	0
indicator [ordinal encoded (count)]	0
precipitation is True	0
thunderstorms is True	0
snow is False	0
thunderstorms is False	0
clouds is False	0
clouds is True	0
precipitation is False	0
snow is True	0
obscuration is False	0
obscuration is True	0
freezing is True	0
freezing is False	0

C Appendix 3: ROC Validation Results

This appendix provides an overview of the ROC validation results detailed in section 5.3. The variation of MAPEs per flight level range is plotted in Figures 29 to 42. The MAPEs are computed between the predicted, planned, and nominal ROCs with the ground truth data, namely *real* vertical speeds from FR24.

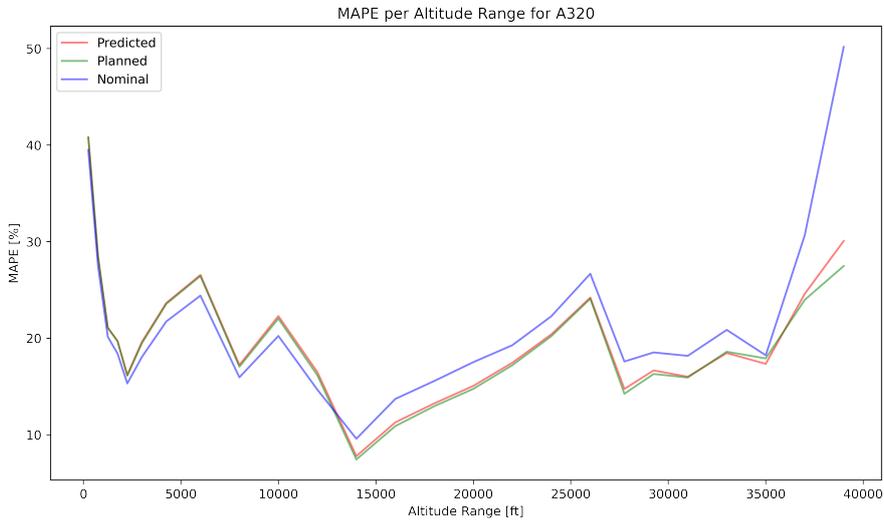


Figure 29: MAPE per flight level range for A320 (enlarged version).

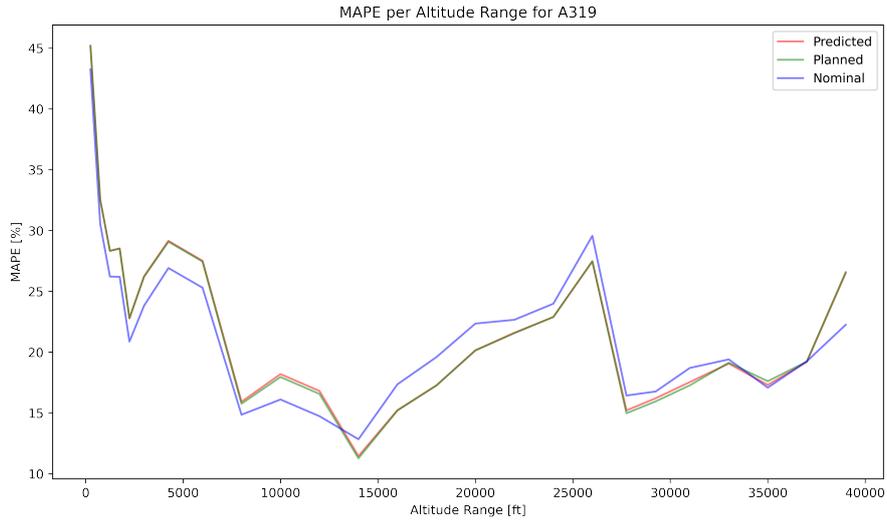


Figure 30: MAPE per flight level range for A319.

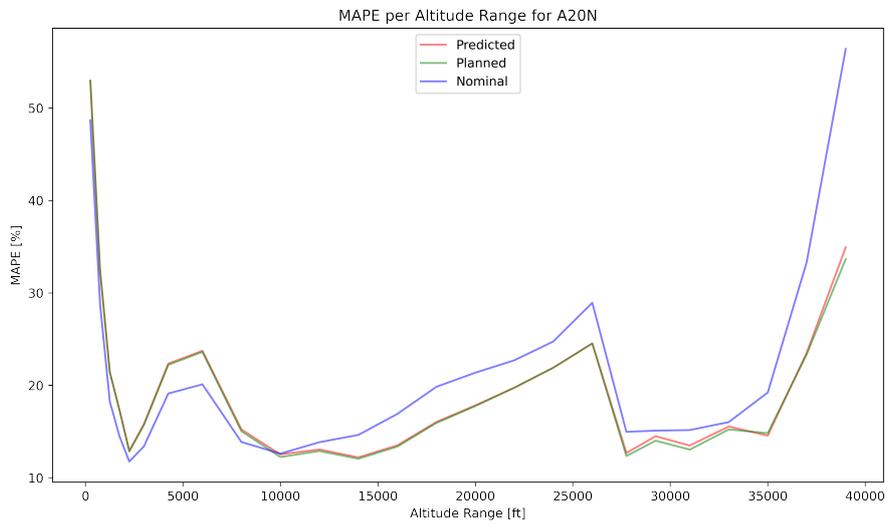


Figure 31: MAPE per flight level range for A20N.

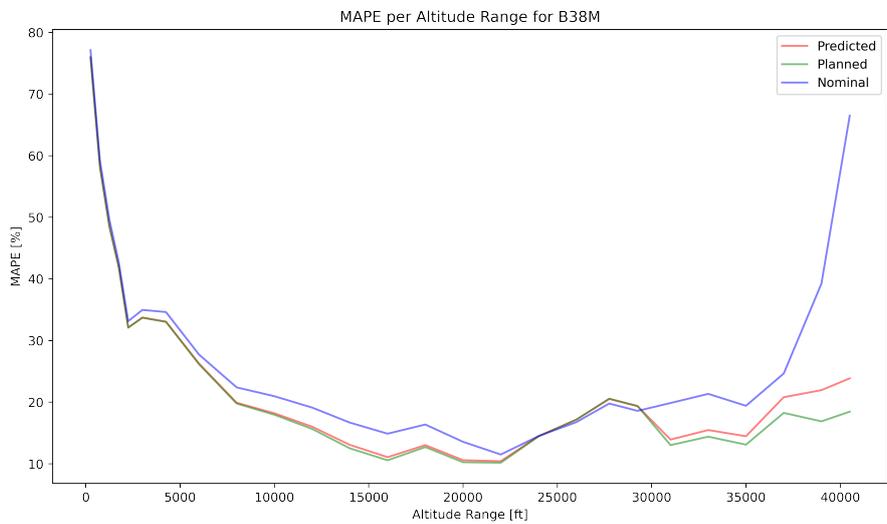


Figure 32: MAPE per flight level range for B38M.

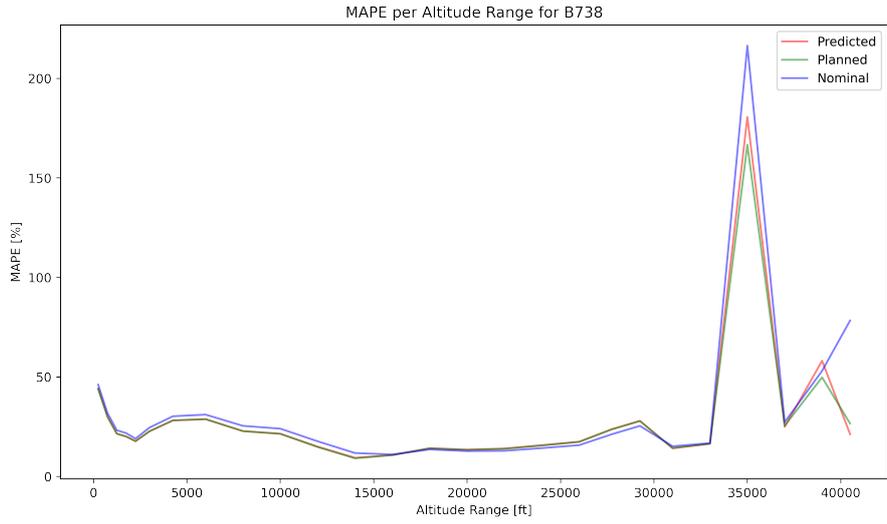


Figure 33: MAPE per flight level range for B738.

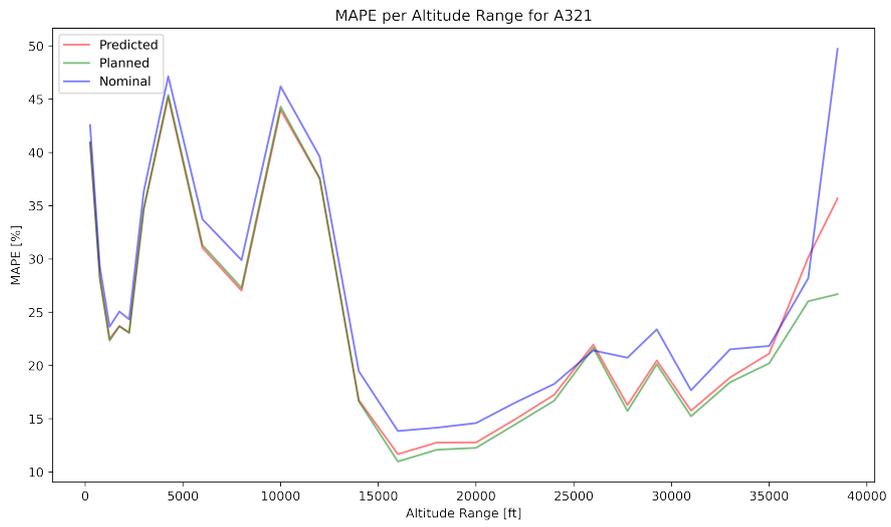


Figure 34: MAPE per flight level range for A321.

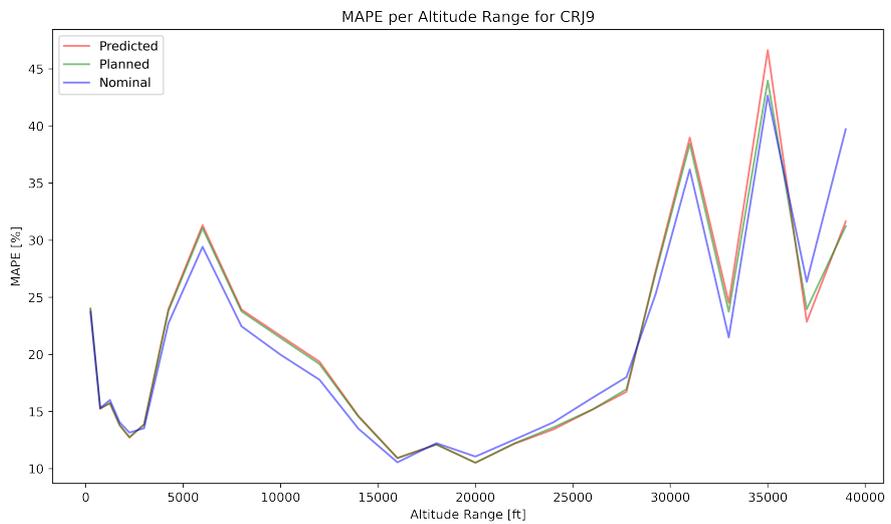


Figure 35: MAPE per flight level range for CRJ9.

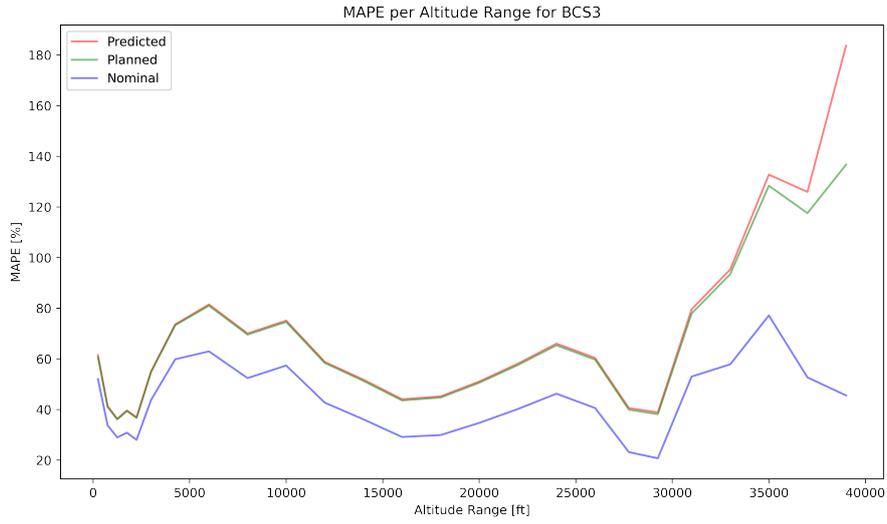


Figure 36: MAPE per flight level range for BCS3.

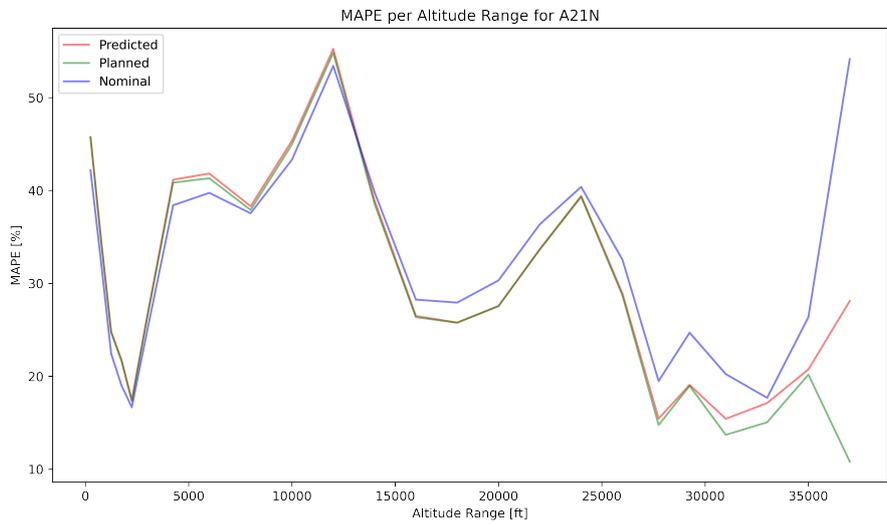


Figure 37: MAPE per flight level range for A21N.

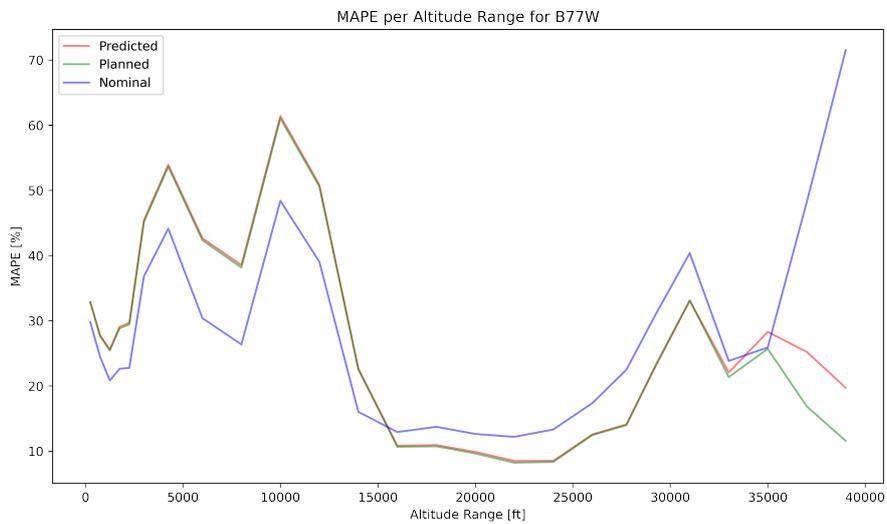


Figure 38: MAPE per flight level range for B77W.

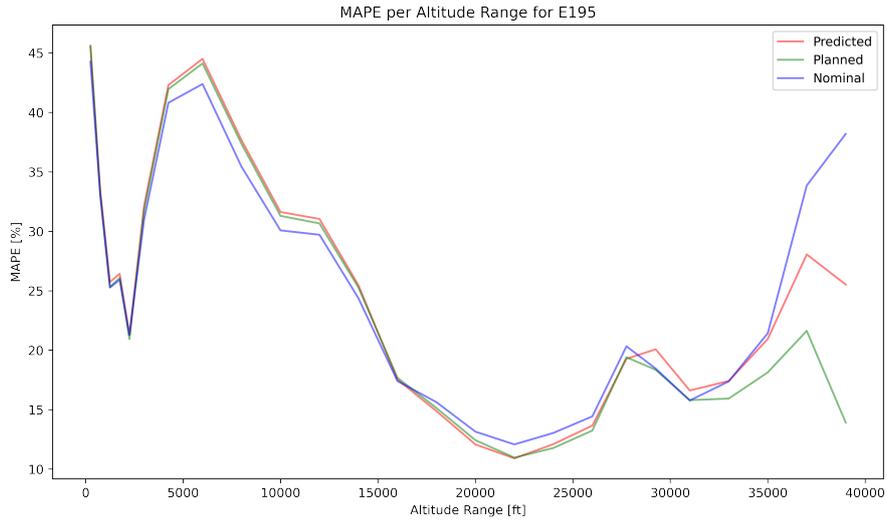


Figure 39: MAPE per flight level range for E195.

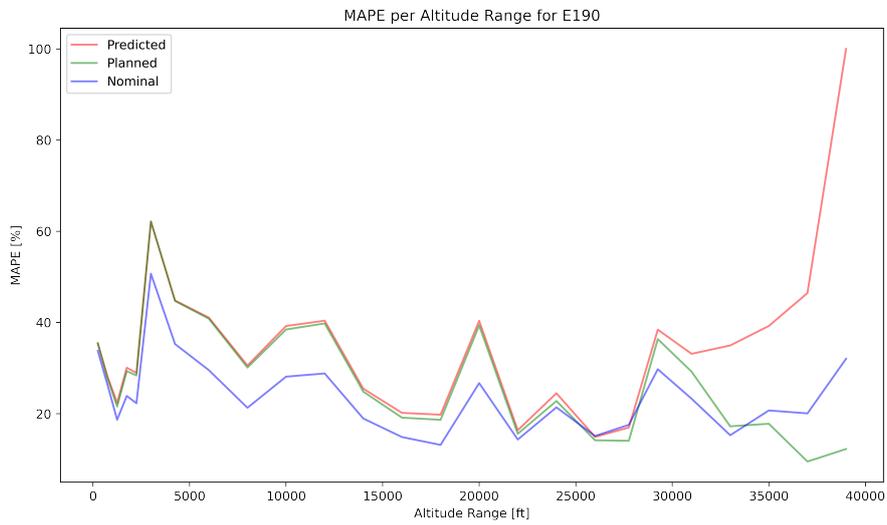


Figure 40: MAPE per flight level range for E190 (enlarged version).

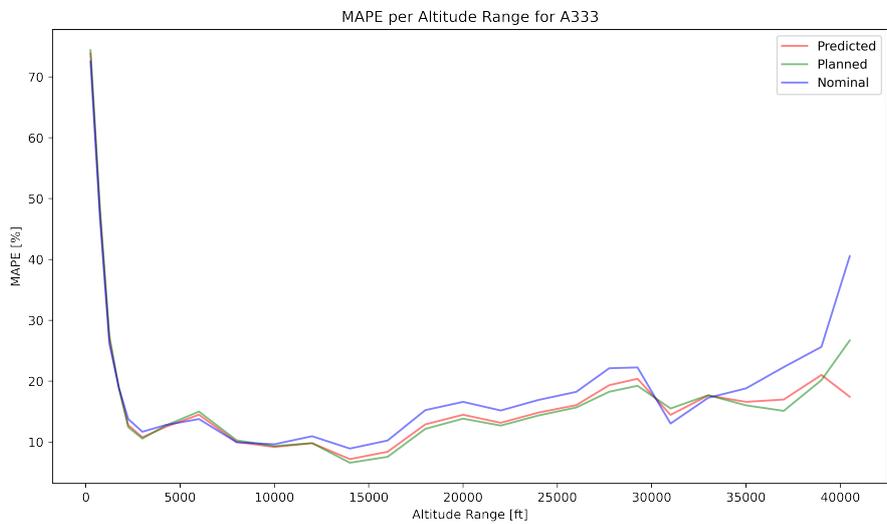


Figure 41: MAPE per flight level range for A333.

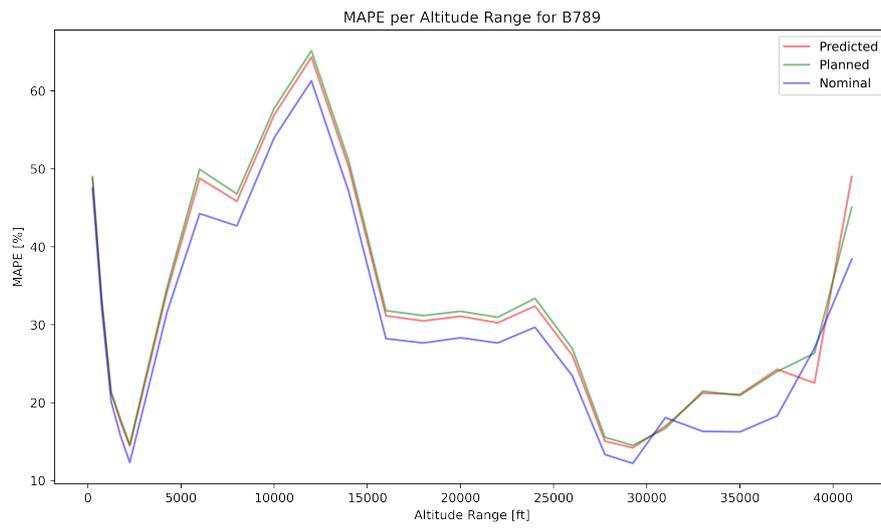


Figure 42: MAPE per flight level range for B789.

II

Literature Study
previously graded under AE4020

1

Introduction

Aircraft take-off weight (TOW) has been a long-sought value by aviation stakeholders. Not only is it a safety-critical parameter for aircraft longitudinal stability and weight balancing, it impacts fuel consumption and plays an important role in trajectory prediction computations. Unfortunately, aircraft operators are generally not willing to share this data because they use it for calculating their cost index, which may reveal sensitive information about their operational trends, making them vulnerable to market competition. Climb and descent phases are particularly affected by TOW and mass variation over time. These parameters essentially dictate the aircraft's performance over the vertical profiles. Besides aircraft weight, speed profiles during climb and descent are also useful for understanding airlines' trends. Combining the two operational parameters, namely TOW and speed profiles, could enable aviation authorities to compute emissions and other climate-oriented parameters, thus giving them the possibility of better regulating aviation's climate impact.

The goal of this project is to develop a machine learning tool to predict aircraft TOW and speed profiles during climb and descent. This MSc Thesis is conducted in collaboration with the Aviation Sustainability Unit at EUROCONTROL, the European Organisation for the Safety of Air Navigation. EUROCONTROL is interested in improving their fuel consumption and emissions estimations via more accurate trajectory predictions. Since the latter are highly influenced by TOW and speed profiles, an effective approach to achieving higher accuracy is the improvement in the quality and fineness of such input data. It is for this reason that this project will focus on correctly predicting these operational parameters.

The structure of the report is as follows. Chapter 2 introduces the problem statement of the MSc Thesis, detailing the need to focus on the topic. Next, Chapter 3 provides all the background information necessary to the understanding of the research topic, followed by Chapter 4 in which previous academic research treating the estimation of aircraft TOW and speed profiles is given. Chapter 5 details all the datasets that will be used during the project development, especially for training the machine learning model. This is followed by a presentation of the research gap and research proposal in Chapters 6 and 7 respectively. These highlight current research opportunities in regards to TOW and speed profiles predictions and propose an innovative approach to tackling the problem. Finally, a conclusion is given in Chapter 8.

2

Problem Statement

While harshly affected by the pandemic, air traffic demand is returning to 2019 levels and continuing to grow rapidly [1][2][3][4]. Without additional measures, the effects of aircraft fuel burn and related pollutant emissions will increasingly impact climate change, while Air Traffic Management (ATM) will grow in complexity. Consequently, current ATM systems must adapt to not only accommodate the high traffic demand, but to do so in an environmentally conscious manner while aiming to reduce emissions.

In recent years, airlines have started to be more and more subject to environmental regulations to reduce aviation emissions and to work towards a carbon-neutral industry. While airlines' main goal is to maximise profit, they are becoming increasingly interested in long-term objectives such as sustainable operations. The Emission Index (EI), given by Equation 2.1 on the left, is defined as the mass of pollutant emitted m_X (in [g]) per unit mass of fuel consumed m_{fuel} (in [kg]). The EI is constant for most pollutants (including CO_2), hence the amount of pollutant emitted is directly proportional to the amount of fuel consumed, as can be seen in Equation 2.1 on the right [5]. Note that when the EI is not constant, the relation is still monotonic [6].

$$\text{EI} = \frac{m_X}{m_{\text{fuel}}} \quad \leftrightarrow \quad m_X = \text{EI} \cdot m_{\text{fuel}} \quad (2.1)$$

Therefore, by reducing their fuel consumption, airlines would not only cut on costs but they would also reduce their impact on the climate. To facilitate fuel consumption reduction, one would require a high-detail trajectory prediction that would in turn allow for a more comprehensive fuel consumption estimation. Unfortunately, trajectory prediction engines are based on aircraft point-mass performance models that rely on unknown or uncertain inputs. These simulate the aircraft motion by relating the forces acting on the centre of mass to its acceleration. However, the differential equations relating these entities imply previous knowledge of the aircraft initial state (mass, thrust setting, velocity, position, etc.), aircraft intent (speed profile, trajectory change points), and atmospheric conditions (temperature, wind). Due to the generalised lack of knowledge regarding these parameters, many assumptions are made to estimate their values. Take-off Weight (TOW) is an example of such parameter, for which there has been much wish to obtain actual values. Obtaining accurate values of TOW and aircraft speed profiles in particular, would improve the quality of input data fed to the trajectory computation engines. Furthermore, this data would be of use to aviation authorities such as the International Civil Aviation Organisation (ICAO), the European Union Aviation Safety Agency (EASA), and EUROCONTROL (ECTL) to better understand the trends and objectives of different airlines. One of the reasons to focus on the climb and descent phases for such predictions is their enhanced influence on fuel consumption reduction.

Among the different phases of flight, take-off is the most fuel-intensive one, with 5 to 40% of fuel consumed for long to short-range flights respectively, followed by landing [7][8]. Although these phases are short in terms of time and distance compared to cruise, when evaluating the fuel consumed per distance flown, take-off and climb proportionately burn the most fuel for two main reasons [9]. The first is the engine setting at full power, to be able to accelerate to required speeds and to overcome the TOW. The second, also relevant for the descent and landing phases, is the higher density of the air at lower altitudes which requires higher thrust from the engines to overcome the higher friction [10]. Note

that the descent and landing phases are slightly less fuel-intensive (in comparison to take-off and climb) due to the lower weight of the aircraft at the end of flight, but mainly because the aircraft is descending and decelerating rather than climbing and accelerating. Another reason is that the drag and runway (RWY) friction are used to decelerate while the potential energy of the aircraft is made use of to further reduce engine power, thus making these phases less thrust-dependent. It is clear that climb and descent are influential phases when it comes to fuel reduction potentials, while being the most prone to last-minute changes introduces an additional area of improvement. The climb phase provides the required environment for the TOW calculation, while speed profiles in the climb and descent phase are still not predicted before take-off, hence the focus of this research.

The Aviation Sustainability Unit (ASU) at EUROCONTROL (ECTL) Headquarters in Brussels, the agency with who this study will be conducted, is looking to improve fuel burn and emission estimations in the climb and descent phases by generating more accurate trajectory predictions through Machine Learning (ML) data modeling.

Currently, vertical profiles are estimated solely based on Flight Plan (FPL) data. When comparing FPLs and actual trajectories (post-flight), the approximations seem to be rather accurate in the en-route part of the flight, however the accuracy diminishes for the climb and descent phases. This could be due to several operational factors such as using a different Standard Instrument Departure (SID) route, Standard Terminal Arrival Route (STAR), or RWY In Use (RIU) which does not correspond to the one in the FPL, missing trajectory information, or missing RWY assignment. Additionally, SIDs and STARS are generally given as two-dimensional indications, that is horizontal indications between waypoints with minimum and/or maximum rates of climb/descent, and some altitude constraints. However, there is no predefined vertical profile with continuous altitude indications. This means that different aircraft may fly along the same SID/STAR with different vertical profiles, which reduces the accuracy of the FPL-based trajectory predictions.

Another reason for the flawed vertical profile predictions is that the inputs required are generally only known partially. As introduced previously, besides the parameters dictated by Air Traffic Control (ATC), other missing inputs include aircraft TOW, type of engines and wing configuration, climb and descent speeds, and pilot choices i.e. use of maximum climb thrust or de-rated thrust, choice of take-off and landing flap settings (and retraction/extension altitudes). For accurate trajectory prediction, it is essential to feed the model with accurate and reliable inputs. In this way, more accurate vertical profile predictions, and consequently better fuel consumption estimations, can be generated.

The Base of Aircraft Data (BADA) team at ECTL Innovation Hub in Brétigny-sur-Orge, is currently working on assessing if some of the unknown trajectory inputs (e.g. TOW, climb/descent speeds) can be estimated from some of the known trajectory inputs (e.g. departure/landing airports, aircraft operator, route length, date of flight). A main issue, however, is that the information they want to predict is not readily-available in most datasets for the following reasons:

- Actual TOW can only be obtained from the airline, for instance in on-board Flight Data Recordings (FDRs). While they occasionally have access to such datasets, these are limited in scope i.e. one airline, few aircraft types.
- The climb and descent airspeeds are sometimes available in Mode S radar or ADS-B recordings, however most repositories do not seem to store this information (or do not give access to it). Although the ground speed is generally available, deducing the airspeed from it requires wind data, which itself is rarely available from Mode S or ADS-B sources. Furthermore, correlating trajectory data with a separate weather forecast model would add a lot of effort and uncertainties to the process.

Due to linked interests and similar research topics, there will be collaboration with both units (ASU and BADA team) throughout the MSc thesis. It is important to note that this project will focus solely on better estimating the input parameters, specifically TOW and speed profiles, to the trajectory prediction infrastructure. Trajectory prediction and fuel consumption and emission prediction are not the scope here, rather future applications of the model developed. Nevertheless, they are still crucial components to understanding the reason behind the start of this project.

3

Background Information

Before diving into the core of the problem, it is important to have a clear overview of all the theory essential to the understanding of the research topic. This chapter presents the main background information for this exact purpose. First, the organisation ECTL is briefly introduced in Section 3.1. Next, Section 3.2 details the Landing and Take-off (LTO) cycle of an aircraft, followed by the basic structure of current ICAO 2012 FPLs in Section 3.3. Finally, an overview of trajectory computation infrastructures is given in Section 3.4 and Section 3.5 explains the basics of machine learning and the most optimal method applicable to hidden operational parameters prediction, according to previous literature.

3.1. EUROCONTROL

This Master of Science (MSc) thesis is conducted within a traineeship program of one year at ECTL, short for the European Organisation for the Safety of Air Navigation. All the information stated below about the agency is taken from their official website [11], unless stated otherwise.

ECTL is a pan-European intergovernmental organisation dedicated to supporting European aviation, both civil and military, and to coordinate the harmonisation of its ATM procedures and systems. The agency was established in 1960 and currently has 41 member and two comprehensive agreement states (in total: 43 states) [12]. As stated in Chapter 2, its headquarters are located in Brussels (Belgium) but it also has three other sites, namely an Innovation Hub in Brétigny-sur-Orge (France), the Aviation Learning Centre (ALC) in Luxembourg (Luxembourg), and the Maastricht Upper Area Control Centre (MUAC) in Maastricht (The Netherlands). Even though it is not one of the European Union (EU)'s official institutions, ECTL being an European organisation, strong relations endure between the two. ECTL also collaborates with aviation industry stakeholders, including airlines, Air Navigation Service Providers (ANSPs), airports, and other international organisations such as the International Civil Aviation Organisation (ICAO) and the European Union Aviation Safety Agency (EASA).

The key functions and activities of ECTL are listed below:

- **Safety:** ECTL strives to maintain and improve air navigation safety across Europe. In order to lower the risk of accidents and incidents, they offer safety analyses, recommendations, and encourage the adoption of best practices to reduce the risk of accidents and incidents.
- **Network Management and ATM efficiency:** ECTL has been nominated Network Manager (NM) by the European Commission, to perform Air Traffic Flow Management (ATFM) across its member states. The area for which ECTL is responsible is shown below in Figure 3.1, divided in Flight Information Regions (FIRs) and Upper Flight Information Regions (UIRs).

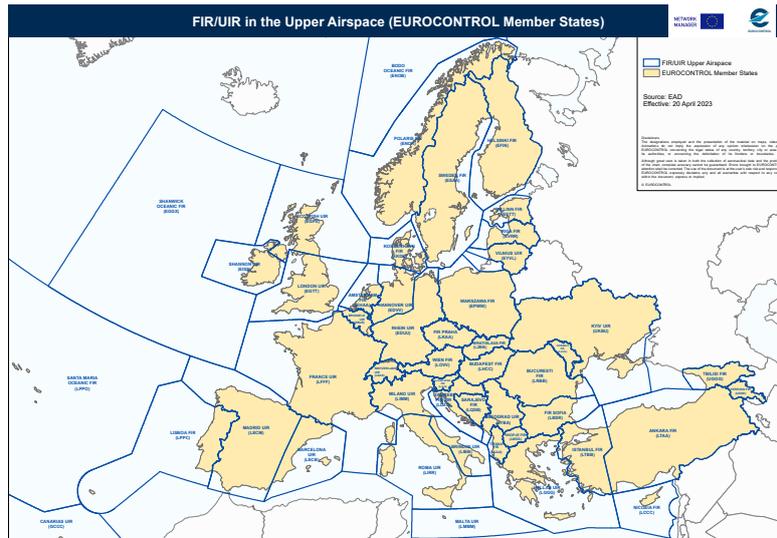


Figure 3.1: ECTL - FIR/UIR upper airspace chart 2023 [13].

The NM acts as a central unit for European ATFM that coordinates the flow of air traffic to improve the overall performance and safety of the network. It focuses on optimising ATM efficiency by reducing delays and balancing the demand with the capacity for airspaces and airports.

- **Research & Development (R&D):** ECTL conducts R&D projects, both internally and in collaboration with the Single European Sky ATM Research (SESAR) program. In fact, ECTL is one of the two founding members of the SESAR project, together with the European Commission. The goal of SESAR is "to develop the new generation of ATM systems capable of ensuring the safety and fluidity of air transport over the next 30 years" [14]. Internal R&D activities aim to improve current ATM technology and procedures, including ECTL tools through automation and the introduction of Artificial Intelligence (AI) methodologies. Note that most R&D activities are conducted at the Innovation Hub in Brétigny-sur-Orge (France), although many research projects are also carried out at MUAC and at the headquarters in Brussels.
- **Data and Information Sharing:** ECTL is committed to exchanging data and information with their member states and stakeholders (ANSPs, airports, airlines) to ensure the safety and efficiency of the ATM network. This data may be weather, traffic, restrictions, and other relevant information.
- **Training and Capacity Building:** The ALC in Luxembourg offers training for Air Traffic COntrollers (ATCOs), ATM operational staff, and other aviation professionals. These programs are for introducing, maintaining, and improving the skills of professionals involved in operational activities within the ATM system.
- **Policy and Regulation:** ECTL provides expert advice on ATM policy and regulation to the European institutions. This helps with the development of the Single European Sky (SES), which "tackles the fragmentation of European airspace and aims at improving ATM performance from the safety, capacity, cost-efficiency and environmental perspectives" [15].

Overall, ECTL is key organisation for the European air transport industry. By closely collaborating with its member states and stakeholders, it ensures the safety and efficiency of the ATM network while aiming to reduce aviation's impact on the climate.

3.2. LTO Cycle

The ICAO emissions certification standards were set to regulate smoke and gaseous emissions, namely unburned hydrocarbons (HC), carbon monoxide (CO) and nitrogen oxides (NO_x). For visible emissions, the smoke limit was set, while other limits were set for the gaseous emissions to address local air quality issues in the vicinity of airports. These use a reference LTO cycle, defined (by ICAO) as basis for the mass of gaseous emissions calculations. The LTO cycle can be visualised in Figure 3.2. In the case of

smoke, the ICAO emissions certification standards are applicable to all turbojet and turbofan engines, however in the case of gaseous emissions, they are applicable to engines with a thrust greater than 26.7 [kN] [16].

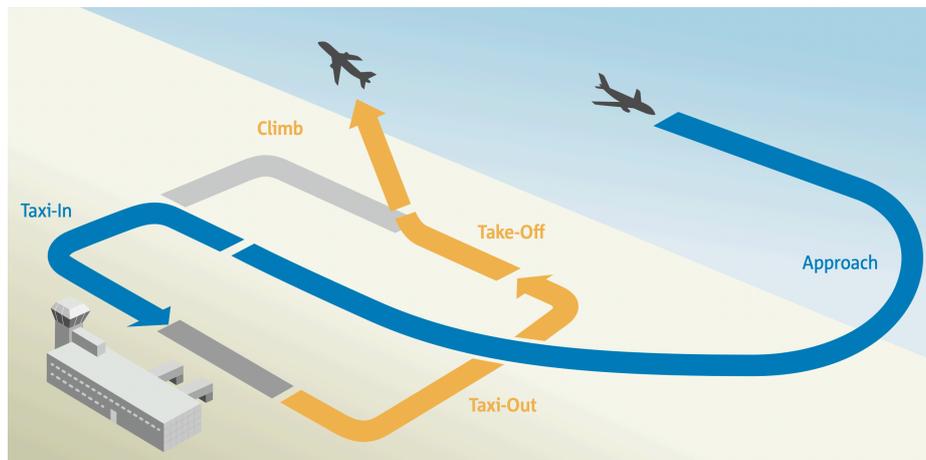


Figure 3.2: LTO cycle as defined by ICAO [16].

The LTO cycle defines four modes of engine operation, corresponding to each phase of the cycle, as shown in Table 3.1. The average duration and thrust setting of each mode are also specified.

Table 3.1: LTO cycle [16].

Mode	Thrust [%]	Time [min]
Take-off	100	0.7
Climb	85	2.2
Approach	30	4.0
Taxi	7	26

For the scope of this project, the thrust settings of the LTO cycle listed above will be used as reference. Note that different airlines and/or aircraft types may have slight variations of thrust setting as they may operate with different objectives, however, as this type of data is unknown and because these companies still have to comply with ICAO regulations, it is safe to assume the thrust settings listed in Table 3.1. Furthermore, although the climb mode is said to last circa 2.2 [min], the 85% thrust setting value will be taken for the full climb profile up to the Top of Climb (TOC). This assumption is accurate because the LTO cycle only considers operations in the vicinity of the aerodrome, hence the short duration of the LTO climb mode.

Besides the notions presented above, the climb phase in particular has a standard procedure for airspeed profile, called the Indicated Airspeed (IAS)/Mach (M) law [17]. It is divided into three main phases:

- Below 10,000 [ft]: climb is done at a constant IAS_1 , limited by ATC.
- At 10,000 [ft]: the climbing aircraft follows a slightly higher IAS_2 , limited by a Mach number M . IAS_2 is better suited than IAS_1 for optimum climb.
- Above the crossover altitude: climb is resumed at constant M . The crossover altitude is where IAS_2 and M both represent the same True Airspeed (TAS).

These phases can be distinguished in Figure 3.3, in which an example of the A320 family is given in green, with standard climb profile: $(IAS_1, IAS_2, M) = (250 \text{ [kts]}, 300 \text{ [kts]}, 0.78)$. Clearly the crossover altitude is at the crossover of $IAS_2 = 300 \text{ [kts]}$ with $M = 0.78$, where both values correspond to the same TAS.

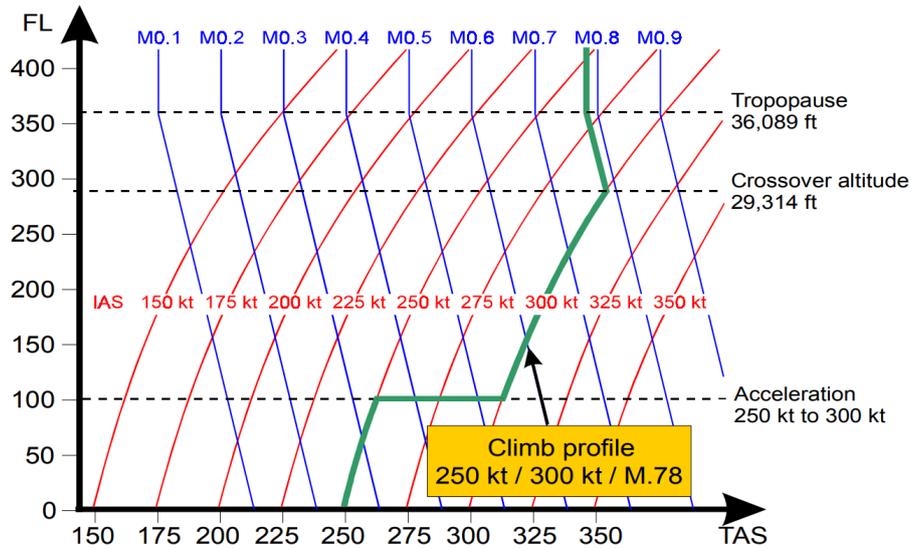


Figure 3.3: Climb Profile at given IAS/MACH Law for the A320 family [17].

Analogously, the speed profile of the aircraft during climb may be described as a set of two Calibrated Airspeeds (CAS) and one Mach speed (M). CAS is essentially IAS corrected for instrument and installation error [18]. These are followed in order from take-off to TOC: CAS₁, CAS₂, and M. As for IAS, CAS₁ is the constant speed with which the aircraft climbs to 10,000 [ft], CAS₂ is the subsequent constant speed with which the aircraft climbs to crossover altitude, and M is the Mach speed maintained by the aircraft until reaching cruise. The IAS/M or CAS/M law is used will be used as reference for the airspeed profile predictions during climb.

3.3. ICAO 2012 Flight Plan

The official definition of a FPL given by ICAO is as follows: 'A navigation FPL is a document prepared in accordance with the instructions of the operator contained in the Operations Manual and used in flight by the pilot to assist in navigation and safe operation of the aircraft.' [19]. Detailed rules regarding contents, completion, changes to, and closing of a FPL can be found in ICAO's Annex 2 [19] and in national flight information publications [20].

When referring to FPL data, this report refers to the ICAO 2012 FPLs. This FPL format was fully implemented on the 15th of November 2012 and it is the current standard in air transport operations [21]. The main parameters contained in such a FPL are presented in this section.

FPLs are made up of 19 items. The majority of the items are completed by the aircraft operators, yet some fields require the input of ATC and communication services. The items that airlines must fill in when filing a FPL are listed in Table 3.2 along with the contents corresponding to the items.

Table 3.2: FPL items completed by aircraft operators [21].

FPL item number	Contents
Item 7	Aircraft identification
Item 8	Flight rules and type of flight
Item 9	Number and type of aircraft and wake turbulence category
Item 10	Equipment
Item 13	Departure aerodrome and time
Item 15	Crusing speed, altitude/level, and route
Item 16	Destination aerodrome, total estimated elapsed time, and alternate aerodrome(s)
Item 18	Other information
Item 19	Supplementary information

Once a FPL is filed by an aircraft operator, it is received by the NM Operations Centre (NMOC) at ECTL. The latter validates, corrects (if necessary), and distributes them to the ANSPs and operational partners concerned [22]. Only after NMOC has accepted the FPL is the aircraft operator able to use it for its subsequent flight.

The information given in the FPLs is stored at ECTL, as will be later described in Section 5.3 of Chapter 5. It is important to acknowledge recent and ongoing developments concerning the Flight and Flow Information for a Collaborative Environment (FF-ICE) format, to be implemented into operations by the end of 2025. This information has been acquired from ECTL experts. The FF-ICE format was introduced to expand the information provided by aircraft operators for trajectory prediction improvement. The enriched data includes:

- 4D trajectories predictions, including the aircraft mass at each point along the trajectory
- aircraft *operational* TOW predictions
- flight-specific performance data, used to match the trajectories predicted by NM with those predicted by the aircraft operators, and further refine the trajectory predictions in the NM systems.

Although this sounds promising, it is important to note that the mass profile and TOW predictions listed above are *optional* parameters to be provided by aircraft operators. Airlines generally classify this data as sensitive, hence not many may agree to share it with ECTL. Even if they do agree, these remain estimations which could be done with a less-efficient algorithm or one that is not ML-based. Nevertheless, it is expected that the mass estimations provided by aircraft operators will have better accuracy, since they have access to their own fleet-specific data. Lastly, while the mandate for FF-ICE FPLs implementation is set to the end of 2025, it is difficult to guarantee that this will be followed by all aircraft operators.

For the above reasons, the motivation of this MSc thesis in predicting aircraft TOW and speed profiles remains valid, while FF-ICE FPLs could be considered for future applications of this study. Note that speed profiles are not available in current ICAO 2012 FPLs for climb and descent phases, therefore predicting these would add to the already-accessible information.

3.4. Trajectory Computation Infrastructure

The basics of trajectory prediction as well as the general functions of the BADA model will be used in parallel to deduct the actual TOW in the first part of this project. It is for this reason that this section focuses on the main components of trajectory computation infrastructures. These consist of three main elements, namely aircraft performance modelling, a trajectory computation engine, and weather modelling. An overview of the steps taken for trajectory prediction, including "real world" differences, is shown in Figure 3.4.

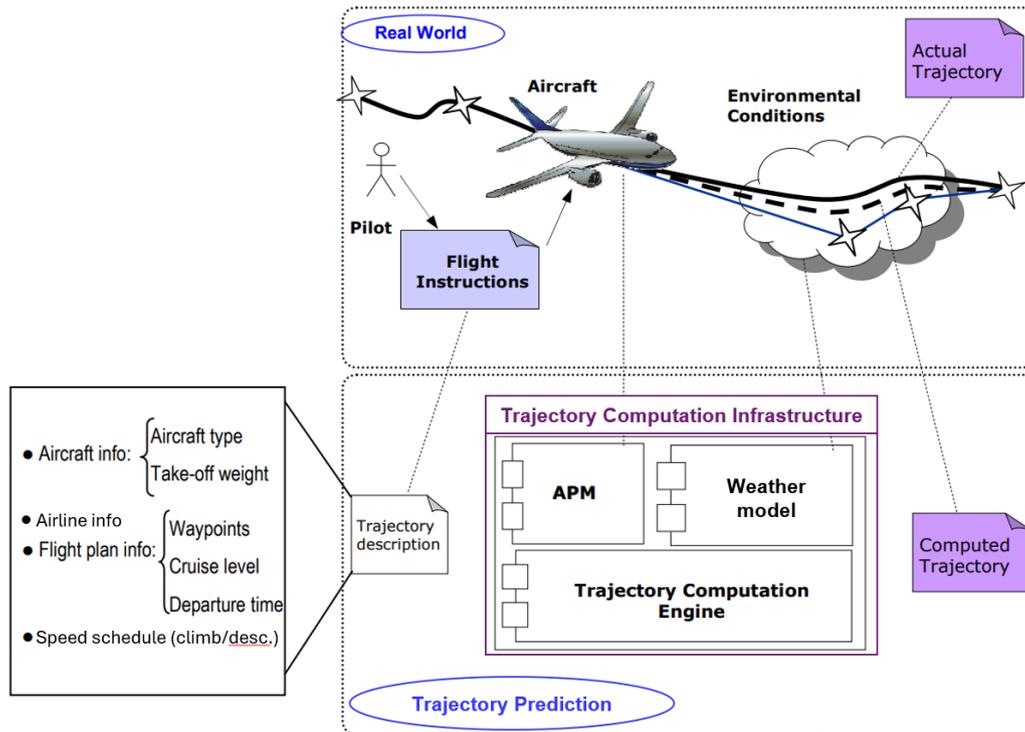


Figure 3.4: Trajectory prediction mapping: input parameters, methodology overview, and "real world" differences. EUROCONTROL.

Aircraft performance modelling is detailed in Section 3.4.1, followed by trajectory computation engines in Section 3.4.2 and weather modelling in Section 3.4.3.

3.4.1. Aircraft Performance Modelling

Aircraft Performance Models (APMs) are key tools in supporting simulation-based studies, such as this one. They allow to realistically replicate the performance of various aircraft types, including the geometric, kinematic and kinetic aspects of their behaviour over the entire operational flight envelope. The APM's role in the trajectory computation infrastructure is to provide accurate outputs (e.g. rate of climb, fuel flow) when fed with given inputs (e.g. aircraft weight, airspeed, altitude). In order to accurately reproduce or predict a trajectory, it needs to be fed with the right inputs, and used in conjunction with a trajectory computation engine and weather model, as stated previously. A few examples of operational input parameters are listed in the bottom left box in Figure 3.4.

The APM which will be used throughout this study, BADA, was developed by ECTL. BADA centralises and harmonises the aircraft performance aspect. It is the international reference for aircraft performance modelling for the purposes of trajectory prediction and simulation. The information provided in this section was taken from ECTL's official website [23], the BADA user manual [24], and the research paper written by the BADA team [25], which gives an overview of the BADA model and its functionalities.

There are two main versions of BADA, labelled as the standard and extended versions. The standard version consists in a set of pre-computed performance tables provided in the form of 'Performance Tables Files (*.PTF)', presenting only a snapshot of each aircraft's performance parameters under default conditions. The PTFs contain summary performance tables of TAS, climb/descent rates, and fuel consumption at various Flight Levels (FLs) for a specific aircraft type. The extended version of BADA is able to compute tailored values of these parameters with its full mathematical/physical model. The latter can be fed with any input values, including TOW, climb and descent speeds, and temperature. Due to the increased variety of input data, the extended BADA is more detailed and accurate in compar-

ison to its standard version which essentially contains static performance parameters. The remainder of this sub-section refers to the extended BADA version.

Besides the different versions, there are three main BADA model "families":

- **Family 3 (BADA3)** provides coverage for close to 100% of aircraft types in the European Civil Aviation Conference (ECAC) area. Its objectives are to accurately model aircraft behaviour over the nominal part of the flight envelope and to meet today's requirements for aircraft performance modelling and simulation. The normal operating range of an aircraft is illustrated in green in Figure 3.5.
- **Family 4 (BADA4)** is a newly developed model family, covering 80% of aircraft types in the ECAC area. It provides increased levels of precision (compared to BADA3) over the entire flight envelope, including the orange and red portions of the flight envelope in Figure 3.5, to enable modelling and simulation of advanced systems and future concepts.
- **Family H (BADAH)** is the new extension of BADA used for trajectory simulation and prediction of helicopter aircraft types. BADAH has only been listed for the knowledge of the author, however it is not of interest to this project because it is only applicable to helicopters.

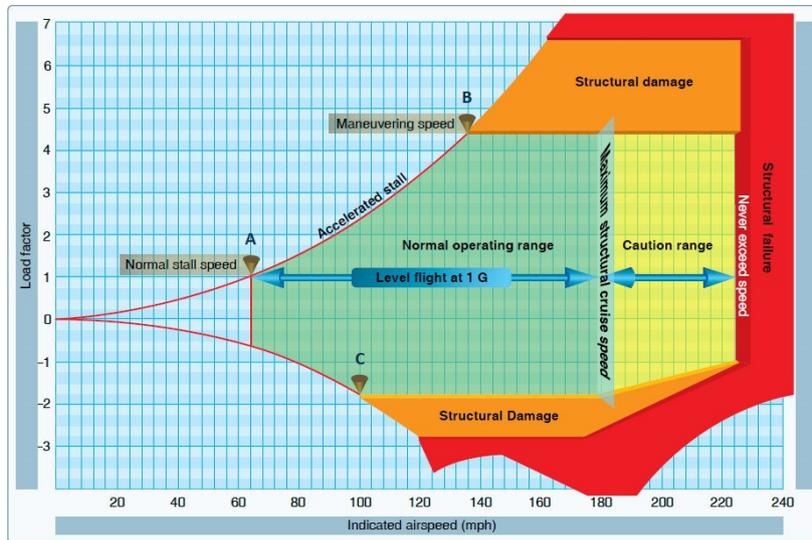


Figure 3.5: Typical flight envelope (V-n diagram) [26]. The horizontal axis lists indicated airspeed (V) and the vertical axis the load factor (n).

Finally, additional integrated tools exist in which BADA integration has been granted by ECTL. These tools are not developed by the agency and include the Federal Aviation Administration (FAA)'s Aviation Environmental Design Tool (AEDT) [27] or MITRE's Terminal Area Route Generation and Traffic Simulation (TARGETS) [28], amongst others.

The APM adopted by BADA is based on a mass-varying, kinetic approach. This approach models the aircraft as a point and considers all acting forces that cause the aircraft motion. The founding relation of BADA3 and BADA4 is based on the Total Energy Model (TEM) shown in Equation 3.1.

$$(T - D)v = mg_0\dot{h} + mv\dot{v} \quad (3.1)$$

In the latter, T is the thrust of the engines, D the aerodynamic drag, v the TAS, m the mass of the aircraft, g_0 the gravitational acceleration at sea level, h the altitude, and \dot{h} the vertical speed. The TEM equates the rate of work done by forces acting on the aircraft to the rate of increase in potential and kinetic energy. To facilitate calculations, Equation 3.1 can be rearranged as shown below to express vertical speed.

$$\dot{h} = \frac{T - D}{mg_0} \cdot v \cdot \text{ESF} \quad (3.2)$$

In Equation 3.2, the Energy Share Factor (ESF) is introduced. It is expressed as a function of TAS and altitude using Equation 3.3.

$$\text{ESF} = \left(1 + \frac{v}{g_0} \cdot \frac{dv}{dh}\right)^{-1} \quad (3.3)$$

Next, the variation of aircraft mass in time \dot{m} is accounted for using the fuel consumption model in Equation 3.4, where F is the fuel flow or fuel consumption of the aircraft.

$$\dot{m} = -F \quad (3.4)$$

Equations 3.2 and 3.4 form a system of Ordinary Differential Equations (ODEs) describing the aircraft motion. To compute this motion on a specific flight segment, one can set boundary conditions over the respective segment. The forces and variables of these equations, namely thrust T , drag D , and fuel flow F , are further modelled in the form of polynomial expressions. This is where the differences between BADA3 and BADA4 are introduced.

Note that, although BADA4 models aircraft dynamics more accurately, BADA3 will be used as APM for the scope of this project for several reasons. First, the case studies that will be analysed, later presented in Section 7.3, only cover civil aviation aircraft, specifically commercial flights transporting passengers. Furthermore, the applicability of operations is restrained to climb and descent. As both conditions lie within the nominal part of the flight envelope covered by BADA3, there is no need to consider the full flight envelope covered by BADA4. Second, BADA3 is the most complete in terms of aircraft types coverage, which is important for the future extension of the project to a wider range of aircraft types. Finally, the most valid reason for choosing BADA3 over BADA4 is the scope of this thesis. The goal is *not* to develop a more accurate point-mass model, but to better estimate the input variables (TOW and speed profiles) fed to BADA. For this purpose, BADA3 is sufficient. Consequently, the remainder of this section only details the expressions of the parameters modelled by BADA3, which will be referred to as BADA in this report.

Thrust is modelled for three flight phases, namely maximum climb, maximum cruise, and idle descent thrust levels, and for three different engine types: jet, turboprop and piston engines. As the aircraft treated in this project only concern aircraft equipped with jet engines, the other engine types are not presented in this section. However, the thrust equations of aircraft with such engines can be found in the BADA overview research paper [25].

Equation 3.5 describes the parameters on which the maximum climb thrust $T_{\max \text{ climb}}$ is dependent in BADA3. It is only a function of altitude h and corrected for temperature deviations ΔT_{ISA} from the International Standard Atmosphere (ISA).

$$T_{\max \text{ climb}} = f(h, \Delta T_{\text{ISA}}) \quad (3.5)$$

The thrust in the remaining flight phases, or any other flight phase for that matter, is modelled as a fraction of maximum climb thrust, as shown in Equation 3.6.

$$T = c \cdot T_{\max \text{ climb}} \quad (3.6)$$

In the above, c refers to the percentage of maximum climb thrust, for which the LTO cycle thrust setting c_{LTO} in Table 3.1 is taken as reference. The percentages listed in Table 3.1 are percentages of *total* thrust available, but c refers to the percentage of maximum *climb* thrust. Therefore, when computing maximum climb thrust with Equation 3.6, $c = 100\%$ which corresponds to $c_{\text{LTO}} = 85\%$ of the LTO climb mode. Analogously, when $c_{\text{LTO}} = 100\%$ during take-off, the value of the corresponding maximum climb thrust percentage is approximated to $c \approx 118\%$. Similarly, during descent $c_{\text{LTO}} = 30\%$ which corresponds to $c \approx 35\%$.

Next, using the result from Equation 3.6, the fuel flow is modelled a function of thrust and TAS as detailed in Equation 3.7.

$$F = f(T, v) \quad (3.7)$$

Finally, aerodynamic drag is calculated using Equation 3.8, where the drag coefficient C_D is expressed as a function of lift coefficient C_L , high lift devices position δ_{HL} and landing gear position δ_{LG} , as shown in Equation 3.9.

$$D = C_D \cdot \frac{1}{2} \cdot \rho \cdot v^2 \cdot S \quad (3.8)$$

$$C_D = f(C_L, \delta_{HL}, \delta_{LG}) \quad (3.9)$$

In Equation 3.8, ρ is the local air density and S is the aerodynamic reference area which depends on the aircraft type considered. For each flight phase defined in BADA, namely take-off, initial climb, climb, cruise, descent, approach and landing, there is a set aerodynamic configuration for high lift devices and landing gear positions. These are accounted for in the drag model.

Keep in mind that Equations 3.5, 3.7, and 3.9 are polynomial expressions. An example is given below in Equations 3.10 to 3.12 for the maximum climb thrust.

$$(T_{\max \text{ climb}})_{\text{ISA}} = a_1 \cdot \left(1 - \frac{h}{a_2} + a_3 \cdot h^2 \right) \quad (3.10)$$

$$T_{\max \text{ climb}} = (T_{\max \text{ climb}})_{\text{ISA}} \cdot (1 - a_4 \cdot (\Delta T_{\text{ISA}})_{\text{eff}}) \quad (3.11)$$

$$(\Delta T_{\text{ISA}})_{\text{eff}} = \Delta T_{\text{ISA}} - a_5 \quad (3.12)$$

In the above, a_{1-5} are coefficients that are specific to the aircraft type considered.

3.4.2. Trajectory Computation Engines

There are many versions of trajectory computation engines and their selection will depend on the application desired, yet one can distinguish two main categories: kinetic and kinematic modelling. While kinetic modelling focuses on forces and energy, kinematic modelling only deals with aircraft motions [29] [30]. In Table 3.3, several ECTL tools are classified according to the type of trajectory computation engine used. Although the scope of the tools may not necessarily be trajectory prediction, each tool has their individual computation engine because aircraft trajectories are useful inputs to their subsequent desired outputs. For example, IMPACT, ECTLS's 'Integrated aircraft noise and emissions modelling platform', delivers noise contour shape files and estimates of fuel burn and emissions for a wide range of pollutants, amongst other parameters [31]. To generate these, the trajectories must be known, hence the need for a trajectory computation engine.

Table 3.3: ECTL trajectory computation engines.

Kinetic models	Kinematic models
IMPACT Maastricht Upper Area Control Centre (MUAC) operational system BADA (TCL)	IFPS ETFMS (FTFM & CTFM)

Notice that BADA also has its own trajectory computation engine, named Trajectory Computation Light (TCL), utilising a kinetic approach. Essentially, the difference between models using the kinetic approach and those using the kinematic approach is reduced to what version of the BADA model is used i.e. standard or extended. The tools using kinetic models, listed in Table 3.3, except for IMPACT which has its own (new) aircraft trajectory calculator, use the extended BADA model. Using its corresponding equations from Section 3.4.1, the TCL is able to compute the tailored trajectory of an aircraft given its initial conditions, the BADA datasets, a sequence of flight intents, and weather conditions.

On the other hand, kinematic models, as those used in NM systems, namely the Initial Flight Plan Processing System (IFPS) and the Enhanced Tactical Flow Management System (ETFMS), use the standard version of BADA with PTFs. The trajectories computed with this version are static predictions, not sensitive to profile variations, and the mass of the aircraft is not taken into consideration in the calculations. It is for these reasons that, for the scope of this project, it would be more appropriate to use a trajectory computation engine based on kinetic modelling because the kinematic approach cannot take into account user-defined speed profiles and it also has limitations regarding the aircraft mass and therefore TOW. Therefore, the extended version of BADA, specifically its trajectory computation engine TCL, will be used in the form of a Python library called pyBADA [32].

3.4.3. Weather Modelling

Weather modelling is done through the use of GRIB files containing weather forecast across the globe. GRIB stands for General Regularly distributed Information in Binary form [33]. The information provided

in this section has been taken from official ECTL documentation on GRIB files processing.

GRIB files are organised by year, month, and day. There are four forecasts per day, done at 00pm, 06am, 12pm, and 18pm. These are called forecast times. For each of these hours, there are 11 predictions for the future, starting with T+06 hours and ending with T+36 hours, with increments of +03 hours. These are prediction times (Hp). Each GRIB file is associated with one prediction time and the forecast time at which it was made, so there is one GRIB file per time of forecast and prediction times. The files are named using the following format:

```
'YYYYmmdd1001metop101T+HpHp_HfHfHfHf'
```

where Hp and Hf are the prediction and forecast times described above.

Each GRIB file contains various GRIB messages, each one containing a grid of data points of a particular measure at a specific pressure level in [hPa], with the associated latitudes and longitudes grid. The available measurements are:

- Temperature
- U component of wind: the wind vector component parallel to the equator. It is positive when the wind is directed towards the East.
- V component of wind: the wind vector component parallel to the meridians. It is positive when the wind is directed towards the North.
- Geopotential height
- Relative humidity

The granularity of the data is 1.25 degrees for the latitude and longitude grid and 25 [hPa] for the pressure altitude, which corresponds to roughly 20 FLs.

Finally, to read and extract information from GRIB files there are two main Python packages available: `pygrib` [34] and `cfgrib` [35]. While ECTL generally uses the first one, `cfgrib` is a better option due to its improved decoding methods. Figure 3.6



Figure 3.6: Example output of GRIB file data extraction - wind vectors. EUROCONTROL.

3.5. Machine Learning

This section provides the essential background knowledge necessary for the implementation of ML algorithms. First, a general overview of AI is presented in Section 3.5.1 introducing all its subsets and their classes. Next, Section 3.5.2 details the ML method which will be implemented in this project, namely Gradient Boosting Decision Trees (GBDTs).

3.5.1. Artificial Intelligence

ML consists in the development of algorithms trained on datasets to build self-learning models that enable computers to make predictions, decisions, or to classify information without being explicitly programmed. Although the terms ML and AI are often used interchangeably, it is important to note that ML is a subset of AI. In fact, the word "AI" serves as an umbrella term for a number of related, but separate, sub-fields. Different stages of AI exist, as detailed in Figure 3.7. Narrow AI, also referred to as Artificial Narrow Intelligence (ANI), is dedicated to specific tasks and it is the only type of AI achievable to this day. The following stages, namely Artificial General Intelligence (AGI) and Artificial Super Intelligence (ASI), have showed little-to-no progress regarding Research & Development (R&D). AGI is a theoretical stage in which the machine would reach human intelligence, being able to reproduce human behaviour including problem solving, learning, planning, and idea generation. Finally, ASI is the most advanced stage in which the machine would theoretically surpass human intelligence [36].

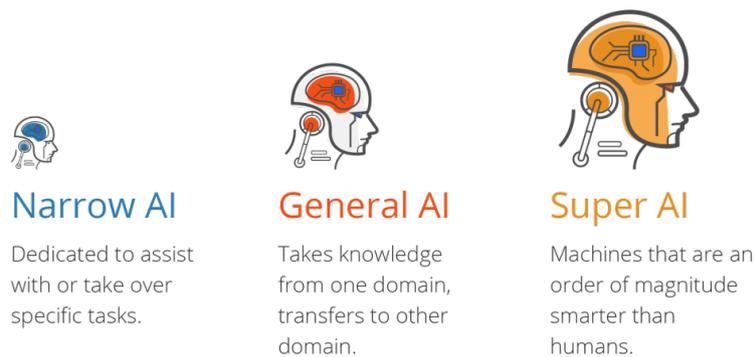


Figure 3.7: AI stages from ANI to ASI with descriptions [36].

To this day, the progress of AGI and ASI can be disregarded, thus ANI will be referred to as AI for the scope of this report. As introduced previously, ML is only a subpart of AI and all the other areas within this field are depicted in Figure 3.8, where ML can be further divided into supervised, unsupervised, and deep learning. In supervised learning, the algorithm learns from previous examples in which a given input (or set of inputs), called *input features*, are associated with a "correct" output, also called *output label*. On the other hand, in unsupervised learning algorithms, the given data is not associated with any output labels [37]. The goal here is not to give a "correct" answer but to identify patterns in the dataset provided.

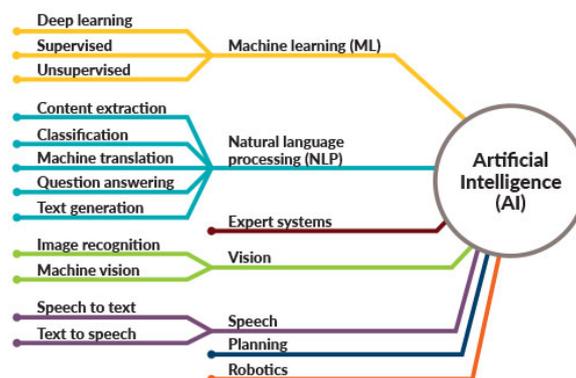


Figure 3.8: AI areas and sub-fields, detailing the three main pillars of ML [37].

Deep learning is a class of ML algorithms using Artificial Neural Networks (ANNs) to mimic the structure and functionality of the human brain [38]. What distinguishes deep learning from ML is the larger amount of data that it can process as well as the automation of feature extraction. The latter re-

moves part of the human intervention required during the learning process [39]. Note that deep learning may use both supervised and unsupervised algorithms [40]. The following list briefly defines the main groups of supervised and unsupervised learning algorithms. These definitions and classifications, as well as most of the information provided in this section, has been taken from Andrew Ng's 'Machine Learning Specialization' course on Coursera [41].

Supervised learning:

- **regression** predicts a number from infinitely many possible outputs
- **classification** predicts a category from a small number of possible outputs

Unsupervised learning:

- **clustering** groups similar data points together (from unlabeled data)
- **anomaly detection** finds unusual data points (can be useful for fraud detection in the financial system)
- **dimensionality reduction** compresses a large dataset to a much smaller dataset while losing as little information as possible

For the scope of this report, only the algorithm which will be implemented are detailed in this section, namely GBDTs. These belong to the supervised learning class. The reason for selecting this specific algorithm is later detailed in Chapter 7 along with its implementation for predicting aircraft TOW and speed profiles.

3.5.2. Gradient Boosting Decision Trees

Gradient boosting is a ML algorithm used for classification and regression problems [42]. While GBDTs are the general designation, Gradient Boosting Regression (GBR) or Gradient Boosted Regression Trees (GBRTs) are specific to regression applications, relevant to this project. The algorithm essentially relies on stacking a series of weak prediction models (predictors) or learners, namely decision trees, to make a more accurate prediction of the desired label(s) through consecutive iterations. The decision trees are added sequentially, as shown in Figure 3.9, each one correcting the previous input dataset [43].

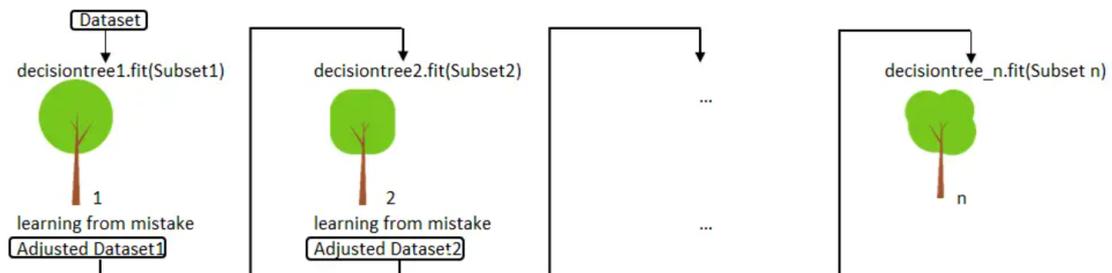


Figure 3.9: GBDT working principle [42].

An example of single decision tree is shown below in Figure 3.10, where the terminology of the relevant parameters is introduced.

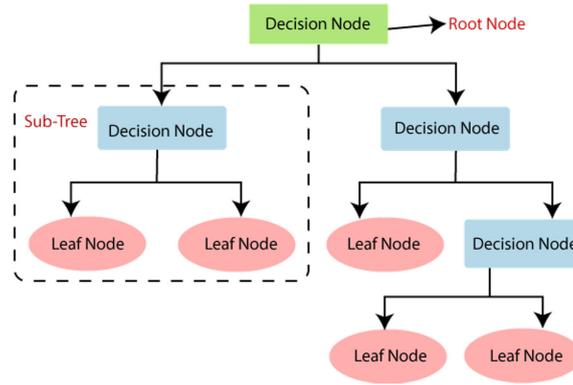


Figure 3.10: Decision tree - overview of components [42].

Decision trees are similar to flowcharts and are made up of decision nodes, leaf nodes, and branches. They are read from top to bottom and the first decision node at the top (in green in Figure 3.9) is called the root node. It partitions the input features recursively based on their value. Each subsequent decision node, including the root node, denotes an input feature of the dataset. Each branch and leaf node of the decision tree denotes a decision and its associated outcome, respectively.

A simplistic example is detailed in the following paragraphs to better illustrate the working principle of GBDTs. The example is taken from an online article [44] in which one tries to estimate the price of a house based on its age, square footage and location. The example dataset is listed in the left-most box of Table 3.4.

Table 3.4: GBR example dataset and first iteration results.

i	Age	Square footage	Location	Price	Residuals	First Decision Tree Residuals	First Output Label (Price) Predictions
1	5	1500	5	480	-208	-208	667.2
2	11	2030	12	1090	402	512	739.2
3	14	1442	6	350	-338	-313	656.7
4	8	2501	4	1310	622	512	739.2
5	12	1300	9	400	-288	-313	656.7
6	10	1789	11	500	-188	-188	669.2

The first step in this algorithm is to calculate the arithmetic average of the output label(s), in this case the price. This is summarised below in Equation 3.13, where n is the number of samples in the dataset.

$$\text{average} = \frac{\sum_{i=1}^n x_i}{n} = \frac{\sum_{i=1}^6 \text{price}_i}{6} = 688 \tag{3.13}$$

In the second step, the residuals are calculated using Equation 3.14 and the results are given in the second box of Table 3.4. Note that for the first decision tree, the predicted value is the average calculated in the previous step.

$$\text{residual} = \text{actual value} - \text{predicted value} \tag{3.14}$$

The third step is to populate the first decision tree with the computed residuals. Notice that the goal of the decision tree is to predict the residuals, not the desired label! The results are illustrated in Figure 3.11



Figure 3.11: First decision tree of the given example, with computed residuals. Own work.

When there are more residuals from the dataset than there are leaf nodes in the decision tree, there will be more than one residual inside the same leaf node. If this is the case, the arithmetic average of the residuals is computed. The latter replaces the set of residuals within one leaf and the decision tree is reduced to single-valued leaf nodes, as depicted below in Figure 3.12.

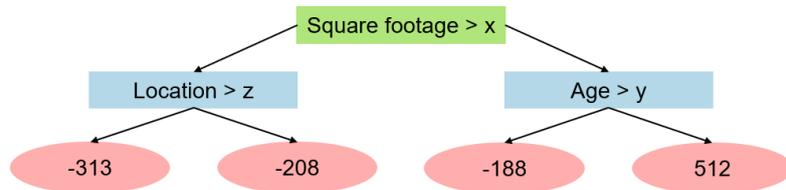


Figure 3.12: First decision tree of the given example, with averaged residuals (when applicable). Own work.

After passing through the decision tree, each data sample is assigned an updated value for its residual. These are listed in the third box of Table 3.4.

Using the decision tree’s residuals, one can compute the predicted value of the output label (price) for each data sample ($i=1$ to $i=6$). This is done as step four of the algorithm using Equation 3.15, where m is the number of iterations.

$$(\text{predicted value})_{m+1} = (\text{predicted value})_m + (\text{learning rate}) \cdot (\text{residual predicted by decision tree})_m \tag{3.15}$$

The learning rate in the above equation is a hyper-parameter introduced to prevent over-fitting the model. Previous implementations of GBR have shown that taking incremental steps towards the solution introduces bias, that is a deviation in the expected results. This achieves a lower overall variance, leading to more accurate predictions for samples outside of the training data. By introducing the learning rate, the model is forced to iterate more times and thus use more decision trees, which allows for the implementation of such an incremental approach towards the final solution. Commonly used values of learning rate lie between 0.1 to 0.3 [45]. In this example, the learning rate is equal to 0.1.

The fifth step is to compute the new residuals once again, yet this time the predicted values from Equation 3.13 actually correspond to the ones from step 4, namely those in the right-most box of Table 3.4. The new residuals are listed in the right-most box of Table 3.5. They will be used for populating the second decision tree of the model, as explained in step 3.

Table 3.5: GBR example dataset and start of second iteration.

i	Age	Square footage	Location	Price	First Output Label (Price) Predictions	New Residuals
1	5	1500	5	480	667.2	-187.2
2	11	2030	12	1090	739.2	350.8
3	14	1442	6	350	656.7	-306.7
4	8	2501	4	1310	739.2	570.8
5	12	1300	9	400	656.7	-256.7
6	10	1789	11	500	669.2	-169.2

Finally, steps three to five are repeated m times, until the number of iterations allows to reach an acceptable level for prediction error. For continuous variables, in linear regression, prediction error is typically computed using the Root Mean Squared Error (RMSE), calculated using Equation 3.16 [46].

$$\text{RMSE} = \sqrt{\frac{\sum(\hat{y}_i - y_i)^2}{n}} \quad (3.16)$$

In the above, \hat{y}_i and y_i are the predicted and observed values for the i^{th} observation, respectively, and n is the sample size. The maximum allowed RMSE will depend on the application and context of the problem. Therefore, the exact value for prediction RMSE will be determined in the upcoming steps of the project.

Once the number of iterations has reached an acceptable value for RMSE and the model is trained, the entire sequence of decision trees can be used to make a final prediction of the output label(s). The final computation for predicting the latter is given by Equation 3.17.

$$\begin{aligned} \text{prediction} &= \text{average value} \\ &+ (\text{learning rate}) \cdot (\text{residual predicted by decision tree})_1 \\ &+ (\text{learning rate}) \cdot (\text{residual predicted by decision tree})_2 \\ &+ \dots \\ &+ (\text{learning rate}) \cdot (\text{residual predicted by decision tree})_m \end{aligned} \quad (3.17)$$

Although this example was given for numeric input features, it is also possible to input other types of variables such as strings and categorical variables, when implementing the GBDTs algorithm. To do so, the dataset must be prepared into the expected format, which consists in encoding the input variables. Encoding can be done in two different ways [47]:

- **Integer encoding:** Each unique category value is assigned an integer value. This type of encoding is acceptable for variables that have a natural ordered relationship.
- **One-hot encoding:** For categorical variables that are not ordinal, this type of encoding is used. It consists in introducing a binary variable for each unique category value.

4

Literature Review

Before any research project, it is important to assess the current state of the art to understand what has already been done, what can be improved, and what is missing in terms of innovation. In doing so, this chapter details the literature review necessary for identifying the research gap that will eventually be the focus of the MSc thesis.

Section 4.1 presents the NM systems used for predicting aircraft trajectories. They are currently operational and do not make use of machine learning techniques. Next, Section 4.2 lists all relevant research projects that were done on the topic in academia, highlighting their main differences such as data used and machine learning approach. Finally, in Section 4.3, an internal project conducted at ECTL in 2021, titled *Climb'n Descent*, is introduced as basis for this MSc thesis.

4.1. Current Trajectory Prediction

There are two main NM operational systems, namely IFPS and ETFMS. Both are able to calculate the vertical profile of the aircraft based on speeds, SID/STAR assigned, waypoints, and routes specified in the FPL. IFPS provides a less detailed pre-tactical estimation since it does not consider weather parameters, while ETFMS gives more accurate estimations by considering weather effects [48][49]. For this reason, IFPS data will not be directly used, but rather the ETFMS data which is in fact deducted from IFPS and FPLs.

ETFMS has two kinds of vertical profile approximations, namely a pre-tactical one given by the Filed Tactical Flight Model (FTFM) [50], and one that is continuously updated throughout the flight with actual position outputs, given by the Current Tactical Flight Model (CTFM) [51]. The latter is an update of the first. The CTFM relies mostly on radar Correlated Position Reports (CPRs), but when these are not available, for example when an aircraft is flying over the Atlantic Ocean, ADS-B reports are used as data source. For this project, the CTFM-generated profile will be discarded and ADS-B data will be taken instead for the flown trajectory data. Furthermore, the FTFM-predicted trajectory parameters, sometimes called ETFMS-predicted trajectory in this report, will serve as input features to the ML model that will be developed. This is further explained in Chapters 5 and 7.

Note that both IFPS and ETFMS (FTFM) only provide estimations of the vertical profiles based on the FPL information described in Section 3.3. However, as stated previously, the real situation is much different due to operational uncertainties, which are not taken into consideration when computing these trajectories. Therefore, a main cause of uncertainty for trajectory prediction lies in the difference between the planned and actual operations, specifically in the operational parameters fed to these models i.e. TOW and airspeed profile.

4.2. Academic Research

The research papers presented in this subsection are used as reference to narrow down the objectives and innovation aspects of the MSc thesis. As stated previously, the vertical profile largely depends on internal variables (e.g. TOW), but it is also subject to tactical intervention from ATC, which in turn depends on other traffic, weather conditions, time of day, etc. Therefore, accurately predicting the climb

and descent profiles is subject to many operational uncertainties, which vary from airline to airline. It is for this reason that, to reduce error margins, the input parameters fed to the selected trajectory computation engine should be precisely estimated. Actual TOW and speed profiles are parameters of interest to ECTL and other aviation stakeholders that do not possess such data. These values can be used for various purposes of which fuel burn calculations and take-off performance, besides trajectory prediction. However, the research done on the topic is not as extensive as the research done on trajectory prediction itself. Furthermore, between the two parameters, there has been more effort in predicting aircraft TOW rather than speed profiles during climb and descent. Note that while trajectory prediction improvement may be an application of the results of this project, as trajectory prediction engines are highly influenced by the operational inputs fed to them, it is not the ultimate goal of this MSc thesis.

In terms of previous work, George A. Vouros et al. [52] developed a data-driven methodology using mechanistic and ML models to estimate several aircraft hidden parameters, specifically selected cost index (CI) and payload mass (PL). Initially, trajectories were enriched with FTFM FPL data, thus taking into account airlines' preferences. DYNAMO, a trajectory prediction engine developed by Ramón Dalmau-Codina et al. [53], was used to obtain "target" FPLs that are enriched with provided input variables in combination with the unknown PL and CI. These include weather data (from GRIB files), aircraft type, origin-destination (OD) pair, airspace structure, initial trajectory, and route charges. DYNAMO is very flexible regarding its input data and it can operate in two different ways: optimisation and prediction modes. The optimisation mode is used to generate the "target" FPLs, while the prediction mode is used as a mechanistic model to predict the Key Performance Indicators (KPIs) of interest i.e. fuel consumption, flown distance, and gate-to-gate time. These are used in series: first, $\text{DYNAMO}_{\text{optimisation}}$ generates "target" FPLs which are then used as training trajectories for the ML model. Then, the hidden variables estimated by the ML model are used as input to the $\text{DYNAMO}_{\text{prediction}}$ to estimate the KPIs for each given trajectory. Therefore, this study goes further than just predicting the PL and CI: it uses these predictions to further estimate KPIs of interest. The goal is, with these KPIs, to be able to estimate the impact of new ATM concepts and technologies on the entire system.

Different ML methods were used for predictions, hence different models were built and compared using the Mean Absolute Error (MAE) of the results as reference. Between lasso regression, neural networks, support vector regression, kernel ridge regression, and gradient boosting trees, the latter achieves the best balance for estimating both PL and CI. An important gap to note from this paper is, although it does estimate two hidden parameters, the TOW is not assessed. Nevertheless, the obtained results and the assessment of each ML method give useful indications on how to approach the problem. Furthermore, in terms of flight phases, this paper does not have a particular focus as it computes the values based on the entirety of the trajectory, including climb, cruise, and descent. Therefore, there may be particular differences in regards to flight phases i.e. some ML methods may work better for climb and descent than they do for the overall trajectory which is dominated by cruise.

Alligier and Gianazza [54] focused on improving aircraft climb prediction by better estimating operational factors, specifically the TOW and speed profile during climb. Note that the speed profile is given by a set of two CAS and one M summarised in a tuple, as previously described in Section 3.2. The approach taken for the ML implementation is a stochastic gradient boosting tree algorithm. This method was selected for its stronger performance in prediction compared to the similar regression trees method, which can be considered a "downgraded" version of gradient boosting. Both methods are known to be insensitive to input monotonic transformations, hence the trade-off.

ADS-B data was taken from The OpenSky Network while The Global Forecast System (GFS)'s GRIB files were used for weather data [55]. Furthermore, two approaches are taken, namely one for predicting TOW and $(\text{CAS}_1, \text{CAS}_2, M)$ right before take-off and another that considers a 10 minutes prediction horizon for real-time estimations. The latter uses *known* data acquired *during* the flight and needs at least nine past points of the considered aircraft for computations. However predictions are made only using the BADA physical model instead of the Gradient Boosting Machine (GBM) technique. Since the goal is to be able to predict the operational parameters *before* take-off, only the results obtained from the first approach (i.e. predictions before take-off) will be considered for the scope of this project. It is useful to notice that an important lacking aspect of this study is the omission of FPL data, which leads to neglecting airline preferences. Including this dataset would allow to improve predictions regarding both operational parameters and vertical profiles, especially for estimations done prior

to take-off. Finally, the author published the ML code on their GitHub page, making it accessible for potential verification and/or validation [56].

Alligier et al. [57] proposed a new approach to improve aircraft climb prediction using ML, where the aircraft mass is considered to be the response variable of a prediction model that is learned from a set of example trajectories. However, the actual mass data is unavailable. To overcome this, the authors propose adjusting a modeled mass for each example trajectory to ensure the modeled power aligns as closely as possible with the observed energy rate on future points. Essentially, the authors suggest replacing the missing actual mass with an adjusted mass that best fits the energy rate on the examples, assuming a maximum climb thrust setting. This adjusted mass becomes the output variable of the prediction model. The approach taken by this paper regarding reverse engineering the trajectory data will be used for this MSc thesis. Note that, while they do have access to some FPL data, they do not have access to the entirety of the information that would enable them to reproduce the vertical profile given by FTFM i.e. no access to the planned route. The FPL data they had access to included requested FL, requested speed, distance between airports, aircraft operator, and departure and destination airports.

A study done at MIT took a different approach, using statistical modelling instead of ML to estimate TOW. Chati and Balakrishnan [58] implemented Gaussian Process Regression (GPR), a non-parametric probabilistic method, to build the regression model. Instead of providing a specific result for the TOW, the complete predictive distribution was given. The latter's uncertainty estimates measure the cumulative impact of components that are not included in the TOW modeling, as well as the random fluctuations in aircraft operation. Note that the main source of data for these predictions was the aircraft take-off ground roll. Therefore, the focus of the study is on this particular flight phase and trajectory variables were extracted from it. The operational flight data used are obtained from the Flight Data Recorders (FDRs) of a major airline. As they had access to such data, a statistical approach was taken instead of ML methods. Furthermore, while they do acknowledge that ML techniques have previously been used for TOW modelling, they consider that the accuracy of the TOW estimates given by these models cannot be computed, due to the unavailability of ground truth data. Indeed, as previously mentioned, ML-based modelling for TOW predictions have used trajectory prediction accuracy for evaluation purposes. Additionally, ADS-B data would not be suited due to the proximity to the ground of the flight phase considered.

While there is a limitation in regards to the flight phase, this paper is comparable this MSc thesis project in other ways, such as potential future applications. After predicting the TOW, they have used the results to estimate aircraft fuel flow rates, since these are highly dependent on TOW. This is similar to what could be achieved with the results of the ML model that will be developed. IMPACT, the ECTL tool introduced in Section 3.4.2, can be used to estimate fuel consumption and emissions associated with the trajectories predicted using better input values for TOW and speed profiles. The difference is that a GPR model was developed to estimate fuel flow rate using the aircraft TOW as a predictor variable, while in this case, an already-developed tool will be used for such predictions. The study showed promising results, with nearly 50% smaller error for TOW compared to models in the Aircraft Noise and Performance (ANP) database. However, due to their limited access to flight data i.e. one single airline, the results and method are not applicable to other airlines or flight phases.

Xinyu He et al. [59] made use of Quick Access Recorder (QAR) data to estimate the initial-climb aircraft mass. A QAR is an airborne flight recorder designed to provide quick and easy access to raw flight data [60]. It is installed and used by airlines in their fleet to enable routine monitoring of aircraft systems and flight crew performance [61]. The idea of this study is to bridge the gap between the model-based and data-driven methods, hence there are two main steps in their approach. First, a physical model is used to select parameters from the QAR dataset and formulate the aircraft mass as an implicit function of these parameters. Second, the Multi Layer Perceptron Neural Network (MLPNN), a ML technique able to approximate any function, is used for approximating the set function. In this way, the results benefit from the advantages of both model-based and data-driven approaches. Furthermore, an important benefit of using this method is that it does not rely on any aircraft-specific parameters, thus it is universally applicable to all aircraft types. However, since the data used is airline-specific, the results may not reflect other airlines' operational parameters, nevertheless, the method could still be applied provided there is access to multiple QAR datasets.

Lastly, Sun et al. [62]-[63], produced a series of research papers, done in sequence, with the goal to continue initial research ([62]) and improve the prediction accuracy of their results. Using ADS-B data and physical kinetic models, Sun et al. [62] proposed two least-squares-approximation-based analytical methods to estimate the TOW. The first method is based on the kinetic model for the lift-off phase, while the second studies the aircraft motion on the runway at each sample moment and estimates the mass recursively.

Next, Sun et al. [64] used flight data from a complete trajectory and calculated aircraft mass at different flight phases using different methods, including fuel-flow models. Subsequently, these mass calculations are combined with the prior knowledge of the initial aircraft-mass probability distribution to yield the maximum a posteriori TOW estimation using the Bayesian inference method.

For the next step, Sun et al. [65] developed a method that is able to estimate the mass of any flight. This paper reports the investigation of the variations previously observed by Sun et al. [64] owing to dependent factors, such as prior distribution, thrust, and wind. Additionally, to validate the proposed estimation process, the results were compared against data recorded during 50 test flights, with TU Delft's Cessna Citation II laboratory aircraft. The validation outcomes revealed a mean absolute error of 4.3% in the predicted mass with respect to the true aircraft mass (among all flights).

Finally, Sun et al. [63] included wind and temperature data from the enhanced Mode-S dataset as additional observations alongside ADS-B data. They proposed a stochastic recursive Bayesian approach that employed a regularised sample importance re-sampling particle filter in conjunction with a non-linear state space model. This approach could eliminate noise from observations and determined an acceptable noise limit to obtain an accurate aircraft mass estimate.

Note that, although this sequence of research projects were focused on estimating aircraft TOW, the approaches taken and developed were not ML-based. While they may be useful in terms of data used, assumptions made, and kinetic modelling, the statistical methods applied are beyond the scope of this MSc thesis.

4.3. Climb'n Descent Project - EUROCONTROL

While the previous sections listed essential research within aircraft TOW and speed profiles prediction methodologies, this section will detail a project focusing on the climb and descent phases of flight. *Climb'n Descent* was an internal project conducted in 2021 at ECTL's headquarters in Brussels. The goal was to improve the accuracy of 4D trajectories (both FTFM and CTFM) predicted by the current ETFMS by using ML, hence the title of the executive report '*AI/ML-based Augmented 4D Trajectory*' [66]. The focus remains on the climb and descent phases, as for this project, and is limited to flights taking off from or landing at an airport located in the NM area. This is the area shown in Figure 3.1 where NM is directly responsible for ATFM.

The reason for presenting this project is that it was the initial topic idea for this MSc thesis. After starting to work at ECTL and having had several meetings with various experts, this project was discovered, and because there was not much room for improvement, the current topic was suggested by the BADA team from the Innovation Hub in Brétigny-sur-Orge. This shifted the focus of the project to operational parameters prediction, rather than the prediction of the entire climb and descent profiles. Essentially, the estimation of TOW and speed profiles is another approach to eventually improving the trajectory predictions, however this will be a potential application of the model results, it is not the scope of this MSc Thesis.

The data considered for this project is detailed in Section 4.3.1, followed by the ML approach taken in Section 4.3.2, and the results in Section 4.3.3. Note that all the information presented is taken from the *draft* executive report [66], as the project had to be suspended due to higher priority of other ongoing projects at ECTL. Therefore, it is only available internally.

4.3.1. Data Considered

ML models are highly dependent on the data they are fed with when it comes to the accuracy of the predictions. The amount, variety, and detail of the data set will affect the results of the model.

The ML model developed in *Climb'n Descent* was fed with extensive amounts of data, precisely all flights in the NM area from the past three years (2018 -2020) were considered. More than 40 features,

of which ETFMS Flight Data (EFD) messages, Meteorological Information (MET), and airline data, were used to train the ML model. The three main datasets are:

- CTFM flown trajectories for actual flown profiles
- FPL and FTFM predicted trajectories for planned profiles
- GRIB files for weather input

Note that for the actual flown profiles, the CTFM trajectory was taken as reference, which mainly consists of CPRs from radar reports. These are sampled every 30 seconds, which may be too large for the dynamicity of the flight phases considered, which suggests a point of improvement in regards to data considered.

The inputs fed to ETFMS for trajectory computation have a significant influence on the predicted profiles. To reduce uncertainty and improve the reliability of the FTFM-predicted profiles, there has been collaboration with various airlines who have provided *actual* values of aircraft TOW for up to 30% of the flights in the dataset considered.

Permutation importance was used to rank the features from least to most influential in regards to the learning performance of the model [67]. A table ranking the parameters that had the highest impact on the predictions is shown below in Table 4.1.

Table 4.1: Candidate features ranked by importance for the climb and descent phases [66].

Climb		Descent	
Weight	Feature	Weight	Feature
494468.7164 ± 269.1501	PERF_CAT_LOWER_FL	392129.5391 ± 248.7002	PERF_CAT_LOWER_FL
217568.8688 ± 138.2701	FTFM_CLIMB_RATE	211356.3405 ± 95.6282	FTFM_DESC_RATE
138494.9605 ± 44.0213	FTFM_MAX_FL	133131.4453 ± 68.8156	FTFM_DESC_FIRST_PT_LAT
114020.7645 ± 86.3738	FLT_DEP_AD	85637.1216 ± 64.1071	FTFM_DESC_LAST_PT_PT_LAT
109271.3590 ± 243.7783	FLT_DEP_AD_LAT	85262.9041 ± 138.5218	FLT_FTFM_ADES_LAT
105701.0231 ± 96.9098	FTFM_CLIMB_FIRST_PT_LAT	80916.0368 ± 71.9405	FLT_FTFM_ADES
95154.7142 ± 86.0832	ICAO_ACFT_TY_ID	72740.5408 ± 34.9251	FTFM_DESC_FIRST_PT_LNG
86846.6291 ± 88.8068	FTFM_CLIMB_FIRST_PT_LNG	70372.2655 ± 109.2796	FTFM_DESC_LAST_PT_LNG
86710.6489 ± 193.9731	FLT_DEP_AD_LNG	69247.5777 ± 83.0451	FLT_FTFM_ADES_LNG
23296.1818 ± 26.1849	FTFM_CLIMB_DURATION	43342.9997 ± 56.8700	FTFM_MAX_FL
21731.4291 ± 59.1714	AO_ICAO_ID	37916.0572 ± 130.2117	FTFM_DESC_DURATION
20337.5237 ± 73.7881	FTFM_CLIMB_FIRST_PT	32727.9660 ± 55.2942	FTFM_DESC_LAST_PT
18971.2889 ± 22.4656	FLT_FTFM_ADES_LAT	12746.5049 ± 19.2558	ETA_DAYOFYEAR
18136.2638 ± 26.9874	FLT_FTFM_ADES_LNG	11355.1165 ± 65.0552	AIRAC_CYCL
18026.4043 ± 22.2186	FTFM_DESC_LAST_PT_PT_LAT	9524.1099 ± 37.4795	ICAO_ACFT_TY_ID
16417.4972 ± 20.0458	FTFM_DESC_LAST_PT_LNG	6437.3164 ± 30.2539	AO_ICAO_ID
15343.8757 ± 44.8245	ETA_DAYOFYEAR	5731.4322 ± 19.5940	FLT_REG_MARKING
15176.5899 ± 32.8208	FLT_REG_MARKING	5658.8823 ± 21.7385	FTFM_CLIMB_FIRST_PT_LAT
15034.2075 ± 24.5128	FTFM_CLIMB_LAST_PT_LNG	5400.5508 ± 40.4232	FTFM_CLIMB_LAST_PT_LNG
14964.0634 ± 29.0470	FTFM_CLIMB_LAST_PT_LAT	5119.9972 ± 15.9033	FTFM_CLIMB_FIRST_PT_LNG
5277.5141 ± 34.1406	AIRAC_CYCL	4980.9815 ± 21.1130	FTFM_CLIMB_LAST_PT_LAT
5241.9545 ± 21.2503	FLT_FTFM_ADES	4700.7469 ± 24.7470	ETA_HOUR
4887.8957 ± 17.6464	FTFM_DESC_FIRST_PT_LNG	4670.4889 ± 45.3331	FTFM_DESC_FIRST_PT
3690.1746 ± 25.7551	FTFM_DESC_FIRST_PT_LAT	4651.6086 ± 44.6057	FLT_DEP_AD_LAT
3012.4545 ± 14.8094	TMP	3753.9084 ± 4.9501	FTFM_CLIMB_DURATION
2183.6777 ± 16.3065	FTFM_DESC_DURATION	3006.2726 ± 18.9401	ETOT_HOUR
1564.0903 ± 14.7829	ETOT_HOUR	2245.8780 ± 20.6452	FLT_DEP_AD_LNG
1345.6206 ± 13.0203	ETA_HOUR	1997.4443 ± 10.4342	FLT_DEP_AD
800.5184 ± 9.6858	ETOT_DAYOFWEEK	1193.9671 ± 10.3647	TMP
753.4219 ± 10.7042	FTFM_CLIMB_LAST_PT	471.1080 ± 6.1305	FTFM_CLIMB_FIRST_PT
721.3021 ± 10.8727	FTFM_DESC_LAST_PT	459.9846 ± 13.7360	ETOT_DAYOFWEEK
637.9216 ± 12.4657	UGRD	292.5480 ± 8.8095	ETA_DAYOFWEEK
610.3421 ± 8.2791	ETA_DAYOFWEEK	231.8482 ± 4.1430	FLT_UID
294.7066 ± 8.4218	VGRD	159.9437 ± 6.6919	VGRD
280.6895 ± 3.4741	WIND_SPD	159.6583 ± 4.7820	UGRD
224.4980 ± 2.5128	FTFM_DESC_FIRST_PT	137.9027 ± 3.8524	FTFM_CLIMB_LAST_PT
94.7180 ± 3.4824	FLT_UID	104.0301 ± 4.4942	WIND_SPD
30.0619 ± 2.5353	WIND_DIR_RELAT_TO_ACFT	62.3850 ± 6.1863	WIND_DIR_RELAT_TO_ACFT

4.3.2. Machine Learning Approach

A Deep Neural Network (DNN) was taken for the ML algorithm. This choice was based on industry-recognised technology and its structure is illustrated in Section 4.3.2. Multiple architectures were tested during hyper-parameter tuning and the most successful one is listed in Figure 4.3.2. Hyper-parameter tuning was a recurrent activity all along the learning process management and the model training. It is the enabler

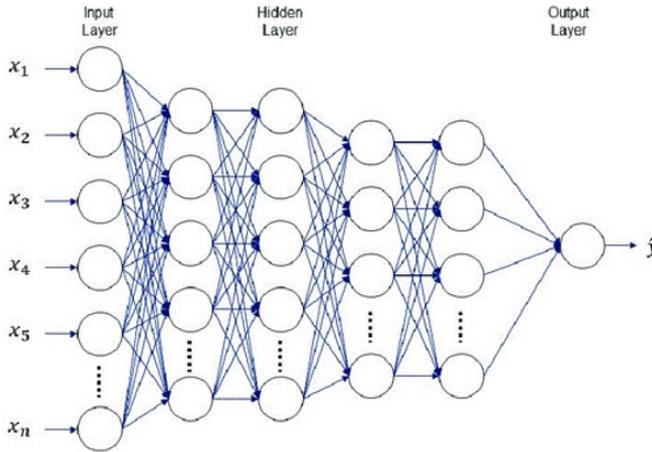


Figure 4.1: DNN structure [66].

Layer number	Number of neurons
1	512
2	512
3	256
4	256
5	128
6	64

Table 4.2: Internal architecture of the DNN [66].

Initially, the development was done using the Standard Rectified Linear Unit (SReLU) activation function, but a noticeable improvement in performance was noticed when adopting the Parametric Rectified Linear Unit (PReLU). This activation function (PReLU) was adopted for a number of its advantages in DNNs, particularly in avoiding the vanishing gradients problem, as was the case with SReLU, but also the avoidance of the dying neuron problem.

Finally, several loss function strategies were studied during the learning and training process. After multiple try-outs, it was decided to use the RMSE which appeared to give the best results on the test set.

4.3.3. Verification & Results

The most appropriate way to assess the performance of the DNN was to analyse its impact on the network situation. This analysis was done using the PREdiction QUALity (PREQUAL) tool, which has been developed to benefit from features available on the NM systems, of which ETFMS. These features allow a user to replay traffic scenarios for a set period of time, based on all the events received by the NM Systems.

The test objective is to verify that the new ML component (DNN) produces more accurate predictions than the current ETFMS baseline. For the remainder of this section, the ETFMS software implementing the DNN is referred to as "ML" and the current ETFMS baseline is named "BL".

The test methodology was applied on two days:

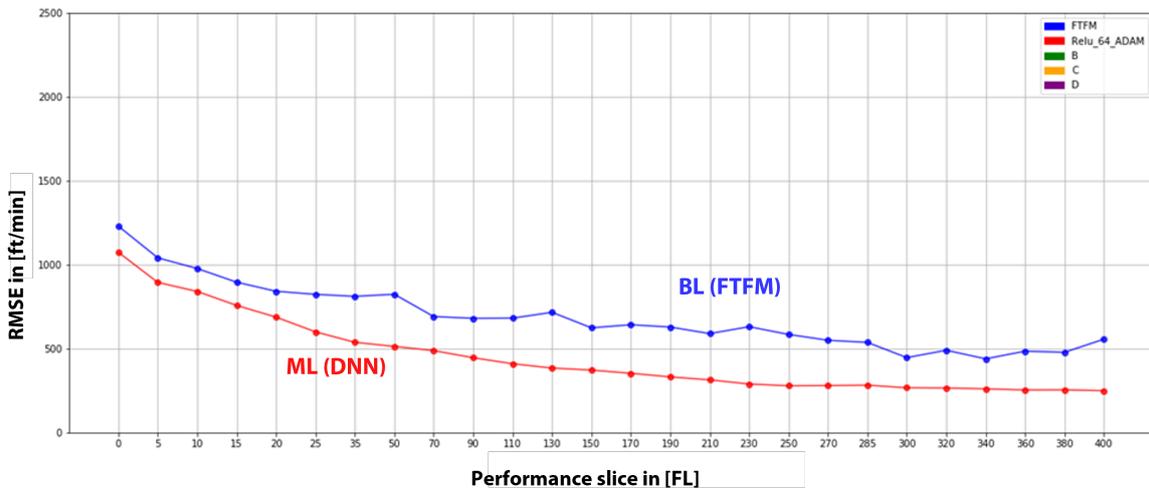
- One cold day: 14-Feb-2020. This day is also known as the day with the highest traffic for that year, just before the Covid-19 pandemic.
- One hot day: 10-Aug-2020.

As the temperature has an impact on barometric altitude, improvement or degradation of ML were likely to be more obvious. The improvements at network level are shown in Table 4.3.

Table 4.3: Improvements of the DNN at network level [66].

	14/02/2020		10/08/2020	
	Average	Maximum	Average	Maximum
BL	544,428	3,226,165	283,051	1,860,046
ML	514,225	2,880,420	265,889	1,655,747
Improvements	5.54 %	10.71 %	6.06 %	10.98 %

In the above, the average and the maximum deviations in FLs with respect to the actual flown trajectory (CTFM) were computed. The last row shows the percentages of improvement achieved with ML compared to BL, which is quite significant. Furthermore, Figure 4.2 plots the improvement of the DNN-modelled climb rate (ML) compared to the climb rate in the FTFM (BL). This was done in terms of RMSE, as stated previously, and the CTFM was taken as reference.

**Figure 4.2:** MSE of the climb rate in [ft/min], comparison of BL with ML [66].

From these results, it is clear that the ML approach achieved notable improvements in regards to trajectory prediction capabilities. Nevertheless, there are some potential points of improvement which could be useful for this MSc thesis as well, despite the different objectives.

First, as already explained, the DNN algorithm was trained with CTFM-predicted trajectories consisting of CPRs. For future steps and improvements, it has been recommended to train the model with ADS-B data. Not only does the latter have a much smaller sampling rate, circa five seconds per ADS-B report compared to 30 seconds for CPRs, ADS-B also has better coverage. This is further explained in Chapter 5. Next, one could focus on the input features fed to the trajectory computation engine and try to improve their estimations. This is mostly intended for TOW and climb/descent speed profiles in particular, as they have the largest impact. Lastly, when FF-ICE FPLs will become available, the ML model could be trained with those. This is expected to bring significant improvements to the computed predictions due to the increased amount of detail and information provided in such FPLs.

5

Data Available

This chapter is dedicated to the data made available for this project, detailing the different datasets. ECTL stores its data in a cloud storage (database) named "Datalake". A general outline of its contents is depicted in Figure 5.1. Each section of this chapter details a different dataset to which access has been granted. Only the information relevant to the ML model to-be-developed is treated, although in reality the respective dataset may contain significantly more parameters than those presented.

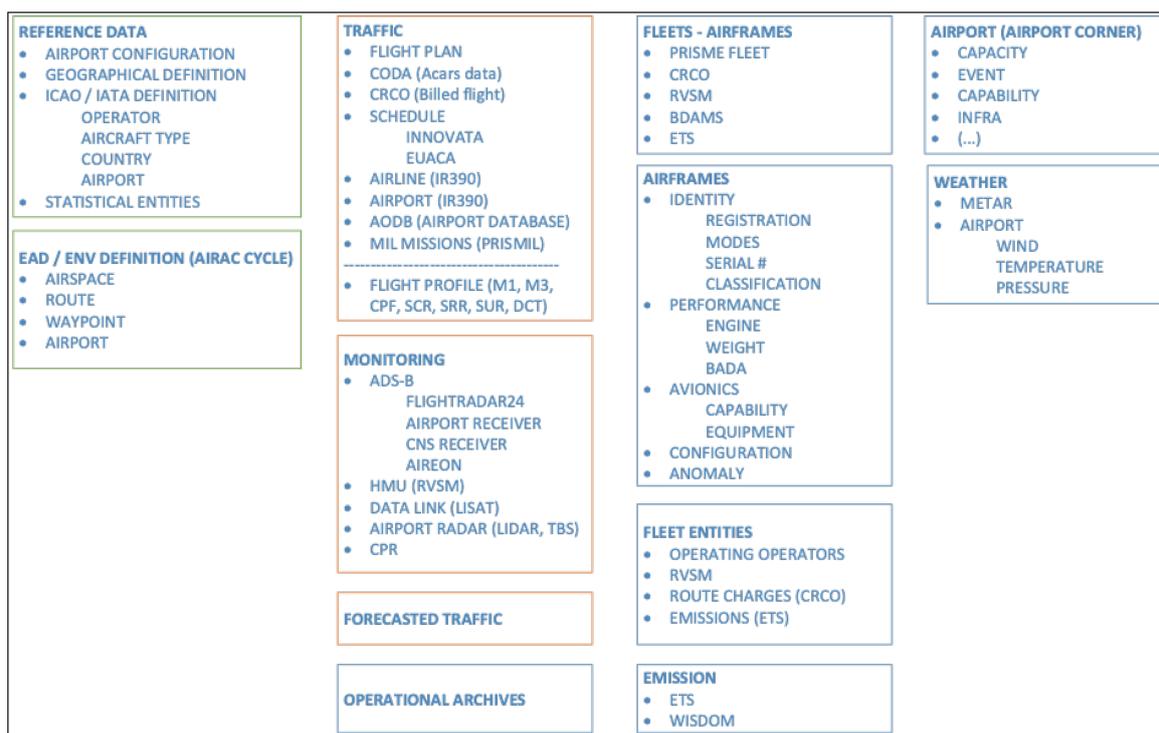


Figure 5.1: Datalake: ECTL's database cloud storage. EUROCONTROL.

The FlightRadar24 (FR24) dataset is detailed in Section 5.1 followed by Aireon in Section 5.2. Next the FPL data, including predicted trajectories from the FTFM (ETFMS) is explained in Section 5.3, followed by weather data in Section 5.4. Note that two ADS-B datasets are considered, namely FR24 and Aireon, due to their different contents, as well as variations in the reliability and availability of the data provided.

5.1. FlightRadar24

FR24 is, for the most part, a ground-based ADS-B data source, although very recently additional specialised space-based tracking data has been introduced for their professional customers [68]. Unfortunately, as this was introduced mid-2023, it is not sufficient data to train a ML model. It is for this reason, amongst others, that Aireon ADS-B data, presented in Section 5.2, will be used as space-based ADS-B source. It is important to note that, compared to Aireon, FR24 provides better accuracy at lower altitudes and in the vicinity of airports.

There are several reasons for selecting ADS-B data instead of radar data. First, the data rate at which CPRs are measured is circa 30 seconds, which would introduce too much uncertainty in the computations, especially for the climb and descent phases. For comparison, FR24 ADS-B reports are sampled every five seconds. Another reason for using ADS-B is the adaptability and smaller size of their ground stations in comparison to radar towers. This allows for more flexibility in regards to their physical placement, enabling better visibility regardless of the terrain or other obstacles [69].

The parameters available from the FR24 dataset, used to train the model, are listed in Table 5.1. Each ADS-B report, at one timestamp, is provided with all these values.

Table 5.1: ADS-B data from FR24.

Name	Data Type	Unit	Description
flightid	string	-	Flight ID in hexadecimal
aircraftid	string	-	24-Bit ICAO Address in hexadecimal
latitude	double	deg	Latitude
longitude	double	deg	Longitude
track	int	deg	True track (over ground)
altitude	bigint	ft	Height above sea level
speed	int	kts	Ground speed
type	string	-	ICAO aircraft designator (type), mapped from address.
registration	string	-	Aircraft registration, mapped from address.
lastupdate	timestamp	s	Time (UTC/GMT) of position since midnight 1 Jan 1970 UTC/GMT
origin	string	-	IATA airport code for origin
destination	string	-	IATA airport code for destination
flight	string	-	Flight number
onground	int	-	Indication of flight status: 1 = on ground 0 = in flight
vspeed	bigint	ft/min	Vertical speed (rate of climb/descent)
callsign	string	-	Callsign: up to 8 characters, as sent from aircraft transponder

5.2. Aireon

The Aireon dataset contains ECTL-owned space-based ADS-B data provided by Aireon. The names of the variables, listed in Table 5.2, are associated with item numbers corresponding to the ones in ECTL's 'Specification for Surveillance Data Exchange ASTERIX' [70].

While FR24 is more accurate around airports, Aireon has much more detailed information and better accuracy. To add up to the benefits of using ADS-B instead of radar data, a crucial point for space-based ADS-B in particular, is that it enables tracking of aircraft in regions that require oceanic coverage [71]. This introduces more opportunities for global coverage. Aireon space-based ADS-B reports are transmitted to ECTL up to six hours before entering or after exiting the NM area (Figure 3.1), with a sampling rate of up to eight seconds. These allow for improved traffic prediction accuracy.

As FR24 is more reliable in the vicinity of airports and because Aireon ADS-B reports are limited to a range of 6 hours outside of the NM area, FR24 data will be used for the take-off part of the LTO cycle as well as for long-haul flights taking off or landing more than six hours away from the NM area.

Table 5.2: ADS-B data from Aireon.

Name	Data Type	Unit	Description
a_040_atp	Whole Number	-	Address Type: 0 = 24-Bit ICAO address 1 = Duplicate address 2 = Surface vehicle address 3 = Anonymous address 4-7 = Reserved for future use
a_040_arc	Whole Number	-	Altitude Reporting Capability: 0 = 25 ft 1 = 100 ft 2 = unknown 3 = invalid
a_071_timeapposition	Decimal Number	s	Time of Applicability for Position: Time (UTC) of applicability of the reported position, in the form of elapsed time since last midnight.
a_073_timerecposition	Decimal Number	s	Time of Message Reception for Position: Time (UTC) of reception of the latest position squitter in the ground station, in the form of elapsed time since last midnight.
a_074_timerecpositionhp	Decimal Number	1/s	Time of Message Reception for Position - High Precision: Time at which the latest ADS-B position information was received by the ground station, expressed as a fraction of the second of the UTC time.
a_075_timerecvelocity	Decimal Number	s	Time of Message Reception for Velocity: Time (UTC) of reception of the latest velocity squitter in the ground station, in the form of elapsed time since last midnight.
a_077_timetransmission	Decimal Number	s	Time of Report Transmission: Time (UTC) of the transmission of the ASTERIX category 021 report, in the form of elapsed time since last midnight.
a_080_aircraftaddress	Text	-	Target Address: 24-Bit ICAO address in hexadecimal
a_131_latitude	Decimal Number	deg	High resolution position latitude
a_131_longitude	Decimal Number	deg	High resolution position longitude
a_140_geometricaltitude	Decimal Number	ft	Geometric Height: Minimum height from a plane tangent to the Earth's ellipsoid, defined by WGS-84.
a_145_flightlevel	Decimal Number	-	Flight Level
a_146_sas	Whole Number	-	Selected Altitude Availability: 0 = no source information provided 1 = source information provided

Table 5.2 continued from previous page

a_146_altitude	Whole Number	ft	Selected Altitude: Selected altitude as provided by the avionics and corresponding one of the following: - MCP/FCU selected altitude (the ATC-cleared altitude entered by the flight crew into the avionics) - FMS selected altitude.
a_155_barometricverticalrate	Decimal Number	ft/min	Barometric Vertical Rate
a_157_geometricverticalrate	Decimal Number	ft/min	Geometric Vertical Rate
a_160_airspeed	Decimal Number	kts	Ground Speed
a_160_trackangle	Decimal Number	deg	Track angle, clockwise reference to "True North".
a_170_callsign	Text	-	Target (aircraft or vehicle) identification in 8 characters, as reported by the target.
a_295_aos_age	Decimal Number	s	Aircraft Operational Status Age
a_295_gh_age	Decimal Number	s	Geometric Height Age
a_295_fl_age	Decimal Number	s	Flight Level Age
a_295_isa_age	Decimal Number	s	Selected Altitude Age
a_295_bvr_age	Decimal Number	s	Barometric Vertical Rate Age
a_295_gv_age	Decimal Number	s	Ground Vector Age, applicable to ground speed and track angle.
a_295_ti2_age	Decimal Number	s	Target Identification Age
ar_bps_bps	Decimal Number	hPa	Barometric Setting
ar_selh_selh	Whole Number	deg	Selected Heading
reception_time	Decimal Number	s	Reception time of the ADS-B report.
reception_date	Text	-	Reception date of the ADS-B report.

5.3. Flight Plans

FPL data is taken from several sources, namely ICAO 2012 FPLs, information derived from it with the ETFMS, and the predicted trajectory from the FTFM. As previously stated, the CTFM trajectory will not be considered for this project since ADS-B data is taken instead.

This data is stored under one dataset in the datalake named "gold_aru_flight_live". Raw FPL data, together with FPL-derived data from ETFMS, are grouped in one table of this dataset, called "flt". The parameters taken from this table are listed in Table 5.3. Note that only the last version of the FPL will be considered, with most recent changes and updates.

Table 5.3: FPL-derived data, either raw FPL data or information computed by ETFMS (based on standard FPL data).

Name	Data Type	Unit	Description
flt_lobt	Date & Time	-	Last estimated off-block date and time (from latest FPL version), as stored by ETFMS.
flt_est_flight_dur	int	min	Estimated duration of a flight.
flt_rvr_val	int	m	Minimum runway visibility range for a flight to land. 0 = closed for all traffic 9999 = free for all traffic Value ranges from 1 to 9998.

Table 5.3 continued from previous page

flt_natflt	Text	-	Indicates whether this flight flies over the North Atlantic Ocean. Y = yes N = no
flt_reg_marking	Text	-	Markings of registration of the aircraft, 18th ICAO field (reg).
flt_uid	Numeric	-	Unique identifier of a flight in the ARCHIVE to Unix (ARU) system.
flt_acft_id	Text	-	Aircraft identification; it may be the registration marking of the aircraft, or the ICAO identification of the aircraft operator followed by the flight identifier, or any other identification string (flttyp); it is not necessarily the callsign.
acft_id_iata	Text	-	Aircraft identification, containing Aircraft Operating Agency (AOA) in IATA (2 or 3 letters), flight number (1 to 4 numbers), and optional suffix (0 or 1 letter), eventually padded with spaces to fit within 8 characters.
flttyp	Text	-	Type of flight. S = Scheduled Air Service N = Non-scheduled Air Transport Operation G = General Aviation M = Military X = Other
ao_icao_id	Text	-	Identifies an aircraft operating agency, which is a person or an organisation or a company engaged or bidding to engage an aircraft operation.
flt_dep_ad	Text	-	ICAO identification of the Aerodrome of Departure (ADEP). In case of: AFIL = Air-Filed FPL, or ZZZZ = unknown, ICAO 2012 strictly states that the aerodrome name or primary fix with location (degrees and minutes, ddmmNdddmmE format preferred) be entered in field 18 ("other information") of the FPL, preceded by a "DEP/" tag.
flt_ftfm_ades	Text	-	ICAO identification of the Aerodrome of Destination (ADES). In case of: ZZZZ = unknown, ICAO 2012 strictly states that the aerodrome name or primary fix with location (degrees and minutes, ddmmNdddmmE format preferred) be entered in field 18 ("other information") of the FPL, preceded by a "DEST/" tag.
icao_acft_ty_id	Text	-	ICAO identification of the type of aircraft.
flt_etot	Date & Time	-	Estimated take-off time. It is the timeOver of the first segment of the FTFM point profile for this flight.

Table 5.3 continued from previous page

flt_eta	Date & Time	-	Estimated time of arrival from the last segment of the FTFM point profile for this flight.
flt_f_rte_len	int	nm	Length of a route.
flt_e_taxitime	int	s	Estimated taxi time.
flt_e_acft_type	Text	-	Estimated ICAO identification of the type of aircraft.
flt_arcaddr	Text	-	24-Bit ICAO address in hexadecimal. It is the unique identification of the aircraft's frame.
flt_f_rte	Text	-	Complete ICAO field 15 information comprising of initial requested speed and flight level and route. Contains corrected route information sent from IFPS to addresses outside of CFMU. It is an array of 1 to N characters (alphabetic, digit, special characters) at the source but truncated to 2000 characters in the NM Data Warehouse (DWH).
airac_cycl	Numeric	-	AIRAC cycle to which belongs information about the flight.

The FTFM-estimated profile data is stored in another table of the "gold_aru_flight_live" dataset, called "pt_prof". The information from this table can be seen in Table 5.4. Note that the first parameter will be used to filter out the data points which were not generated with the FTFM. To do so, the following constraint will be set: flt_model_ty = 1.

Table 5.4: FTFM computed trajectory parameters, based on FPL and FPL-derived information from ETFMS.

Name	Data Type	Unit	Description
flt_model_ty	int	-	Classification of the flight model: 1 = FTFM 2 = RTFM 3 = CTFM
pt_prof_seq_no	Numeric	-	Rank of the point into the sequence. All points (segments and vectors) are included in this sequence. Generated by the DWH.
pt_prof_seg_seq_no	Numeric	-	Rank of the point into the segment sequence. Only segments are included in the sequence (vectors will not be included and will have a null value). Generated by the DWH.
pt_prof_pt_ty	Text	-	Type of the element of the point profile. F = First segment P = En-route segment on a Point G = En-route segment on a Geographical position linked to the FPL S = En-route segment on a geographical position not linked to the FPL, but added to have a better approximation of a long DCT segment L = Last segment V = Vector

Table 5.4 continued from previous page

rte_ty	Text	-	Classification of a route. SR = Standard Route AR = Air Route AP = Arrival Procedure DP = Departure Procedure TR = TOS Route ST = Standard path NR = No Route (DCT) OT = OTS PT = Polar Track GR = Generated Route UR = User Route
pt_prof_pt_id	Text	-	Identification of the point or the aerodrome to which the point is associated. This information is only given for points of type F, P, and L.
pt_prof_rte_id	Text	-	Identification of the route with which the point is associated. This information is not given for points of type N or V, nor for routes of type NR.
pt_prof_time_ov	Date & Time	-	Date and time flying over the point, as given by one of the flight models. For elements of type 'F', this is the take-off date and time; for elements of type 'L', this is the date and time of arrival.
pt_prof_fl	int	-	FL of the flight at this point. The ADEP/ADES FL is now filled with the highest FL it reaches during its whole trajectory, according to a given flight model.
pt_prof_lat	Text	-	Latitude, given as part of the location of the point. Only given for points of type G and S.
pt_prof_long	Text	-	Longitude, given as part of the location of the point. Only given for points of type G and S.
flt_model_lobt	Date & Time	-	Last estimated off-block date and time, as stored by ETFMS.
flt_model_flt_uid	Numeric	-	Unique identifier of a flight in the ARU system.
pt_prof_covered_distance	int	nm	Distance from ADEP to point.
pt_prof_trend_before	Text	-	Trend before the profile element "=" : indicates that flight is cruising "/" : indicated that flight is climbing "\ " : indicates that flight is descending " " : empty value
pt_prof_trend_after	Text	-	Trend after the profile element "=" : indicates that flight is cruising "/" : indicates that flight is climbing "\ " : indicates that flight is descending " " : empty value

When manipulating the data, it is crucial to match the ADS-B data with the data from or derived from the FPL. The parameters which link the different datasets are listed in Table 5.5. These will be used for data matching between the datasets described so far.

Table 5.5: Matching ADS-B data with FPL data.

ADS-B		FPL	
FR24	Aireon	FPL ("flt")	FTFM ("pt_prof")
aircraftid	a_080_aircraftaddress	flt_arcaddr	-
callsign	a_170_callsign	-	-
registration	-	flt_reg_marking	-
earliest 'lastupdate'	earliest 'reception_time'	flt_etot	-
latest 'lastupdate'	latest 'reception_time'	flt_eta	-
-	-	flt_uid	flt_model_flt_uid
-	-	flt_lobt	flt_model_lobt

5.4. Weather & Wind Forecast

The weather data is produced by the Satellite Distribution System (SADIS) who delivers World Area Forecast System (WAFS) data. SADIS is developed and operated by the MET Office on behalf of the ICAO. All ICAO member states can have access to the SADIS data, by implementing a connection to the SADIS File Transfer Protocol (FTP). A connection was already present at ECTL and data has been gathered since 2015.

There are three main types of sources for the SADIS weather data:

- **Meteorological Aerodrome Report (METAR):** contains current weather conditions at the time that the report is prepared. It is updated every 30 minutes and valid for the respective 30 minutes.
- **Terminal Aerodrome Forecast (TAF):** contains forecasted weather conditions for a given area around the airport. It is updated four times a day and is valid for 30 hours [72].
- **GRIB files:** contain wind vector predictions at specific pressure levels, amongst other weather parameters. Section 3.4.3 detailed the files' content as well as how to process the data.

The METAR and TAF will be used for the take-off part of the LTO cycle, as they are only applicable to low altitudes and in the vicinity of the aerodrome of departure/destination. Specifically, they are valid in a radius between 8 to 16 [km], as shown in Figure 5.2. The altitude up to which they are applicable depends on the height of the ConTRol zone (CTR) around the respective aerodrome, which can be found in its Electronic Aeronautical Information Publication (EAIP). These tend to be in the same order of magnitude, for example at Schiphol Airport it is up to 3000 [ft] [73].

METAR and TAF information is stored in a dataset named 'gold_metop_pbiafd'. The parameters of this dataset are listed in Table 5.6, where 'text_info' contains the full report. For completeness, the latter will be decoded using an open source Python library 'metaf2xml' [74].

Table 5.6: SADIS meteorological dataset in the vicinity of aerodromes.

Name	Data Type	Unit	Description
airport	Text	-	ICAO identification of the aerodrome.
event_type	Text	-	Terminal Area (Aerodrome) Forecast (TAF) or METeorological Aerodrome or Aeronautical Report (METAR).
text_info	Text	-	Full report (TAF or METAR).
time_of_issuance	Date & Time	-	Date and time at which report was issued.
valid_from	Date & Time	-	Date and time from which report is valid.
valid_to	Date & Time	-	Date and time up to which report is valid.

Besides the data above, 'gold_metop_pbiafd' also contains different meteorological events, extracted from the METAR and TAF reports. These include:

- | | | | |
|----------------|----------------------------------|-------------------------|--------------------|
| • patches | • drizzle | • snow | • sand |
| • blowing | • hail | • unknown precipitation | • volcanic ash |
| • low drifting | • small hail and/or snow pellets | • mist | • dust storm |
| • freezing | • ice crystals | • widespread dust | • funnel clouds |
| • shallow | • ice pellets | • fog | • dust sand whirls |
| • partial | • rain | • smoke | • squalls |
| • showers | • snow grains | • haze | • sandstorm |
| • thunderstorm | | | • cumulonimbus |

Their occurrence is represented by '1', while their absence by '0' i.e. the events are essentially binary variables.

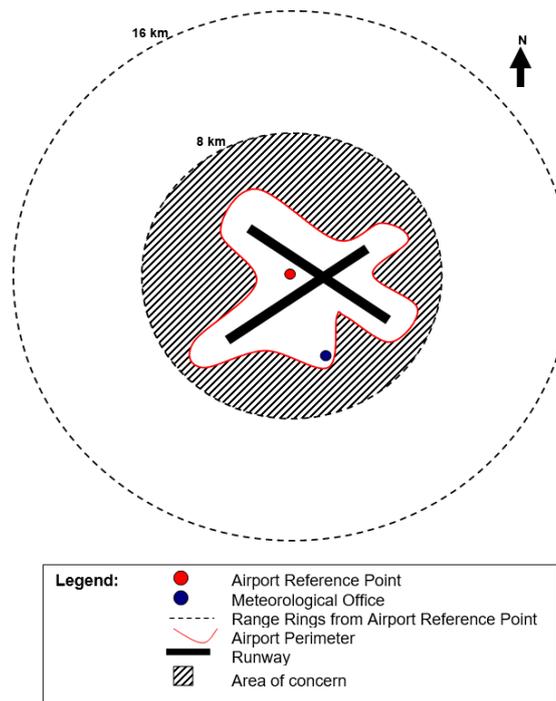


Figure 5.2: Illustration of aerodrome area around which the METAR and TAF are valid [75].

Beyond the areas depicted in Figure 5.2, GRIB files will be used for the wind predictions at different FLs. Once again, the approach described in Section 3.4.3 will be followed to extract the desired data.

6

Research Gap

Looking at previous research presented in Section 4.2 and the *Climb'n Descent* project done at ECTL in Section 4.3, several research gaps can be identified. These are described in the following sections.

6.1. Data Considered

In regards to data considered in previous studies, there is a lack of access to FPL data. Although Alligier et al. [57] did have access to some FPL data, they did not have access to the whole dataset. The FPL parameters they were able to gather were: requested flight level, requested speed, distance between airports, aircraft operator, and departing and arrival airports. As these are limited in scope, the airline preferences could not be reflected within the computed predictions. Fortunately, this gap will be filled by using the FPL and FPL-deduced data from NM, listed in Tables 5.3 and 5.4.

6.2. Prediction Time Frame

The time frame used for predictions in the previously presented research papers is not consistent and sometimes unclear. While Alligier and Gianazza [54] predict the operational parameters before take-off and Sun et al. [63] estimate the aircraft mass within 30 seconds once the aircraft is airborne, there is no specified time frame for the other studies. The MSc thesis will aim to estimate hidden values, namely TOW and speed profiles, within a time horizon of three to four hours before departure/arrival, depending on the time at which the last FPL update has been done. This is reflected in the title, which highlights the *dynamic* nature of the climb and descent profiles. Note that pre-tactical predictions are not excluded and the developed model may also be used to predict vertical profiles up to one day before operations, provided this time frame corresponds to the one of the last filed or changed FPL.

6.3. Results Applicability

Another important aspect to consider is that improving the accuracy of fuel consumption and emissions prediction has not yet been attempted via the estimation of operational parameters. Although these parameters are highly related and fuel flow is closely dependent on aircraft mass and vice versa, there has been no research correlating both predictions, to the knowledge of the author. While this is *not* the scope of this project, the estimated operational parameters could be used subsequently for the potential improvement of both trajectory prediction and corresponding fuel consumption and emissions. This is briefly detailed in Section 6.3

7

Research Proposal

Following the research gap and all the background information presented previously, the research proposal is formulated in this chapter. First, the research objective and research questions are presented in Section 7.1, followed by the methodology taken to achieve them in Section 7.2. The latter makes use of the data available from Chapter 5. Next, the case studies are briefly described in Section 7.3 along with the verification and validation strategy in Section 7.4. Lastly, an overview of the project planning is given in Section 7.5, including the work breakdown and Gantt chart.

7.1. Research Questions and Objectives

The objective of the Msc thesis is to:

Develop a ML tool to predict aircraft TOW and speed profiles during climb and descent.

To achieve this research objective, the *main* research question that must be answered throughout the project is:

How to develop a ML method that correctly estimates aircraft TOW and speed profiles during climb and descent?

In addition to the main research question, more sub-questions (sometimes with sub-sub-questions) have been formulated. These help with the structure of the research journey and with the organisation of the work packages that will be followed.

1. Which ML methods are more efficient at providing accurate predictions of aircraft TOW and speed profiles during climb and descent?
 - (a) What ML methods are used for aircraft TOW and speed profiles prediction?
 - (b) What ML methods are used for other aircraft operational parameters prediction?
2. How can ML be used to predict aircraft TOW and speed profiles without having direct access to such historical data?
 - (a) What flight trajectory data is needed to extract aircraft TOW and speed profiles during climb and descent?
 - (b) How can aircraft TOW and speed profiles be obtained through reverse engineering flight trajectories?
3. How can FPL data and FTFM-predicted trajectories improve the estimations of aircraft TOW and speed profiles during climb and descent, when used as input features to the ML model?
4. What input features fed to the ML model have the most influence on the aircraft TOW and speed profiles predictions?

Answering the sub-questions, in the order listed above, will allow to answer the main research question with a structured work flow. In this way, the research objective can be effectively achieved.

7.2. Methodology

The methodology followed to answer the main research question and to comply with the research objective is detailed in this section. There are two main phases in this project, namely trajectory augmentation and the implementation of the ML algorithm selected. Sub-sections 7.2.1 and 7.2.2 describe these two work packages. Lastly, an overview of potential applications of the ML model results is given in Section 7.2.3. The future steps regarding usage of results are of particular interest to ECTL.

7.2.1. Trajectory Augmentation

The data augmentation process, illustrated in Figure 7.1, consists in estimating the TOW and speed profiles by reverse-engineering a flown trajectory. To do so, the BADA3 model with its system of ODEs adopted by its TCL is used. Specifically, as described in Section 3.4.1, Equations 3.2 and 3.4 are rearranged to obtain the aircraft initial mass and therefore TOW.

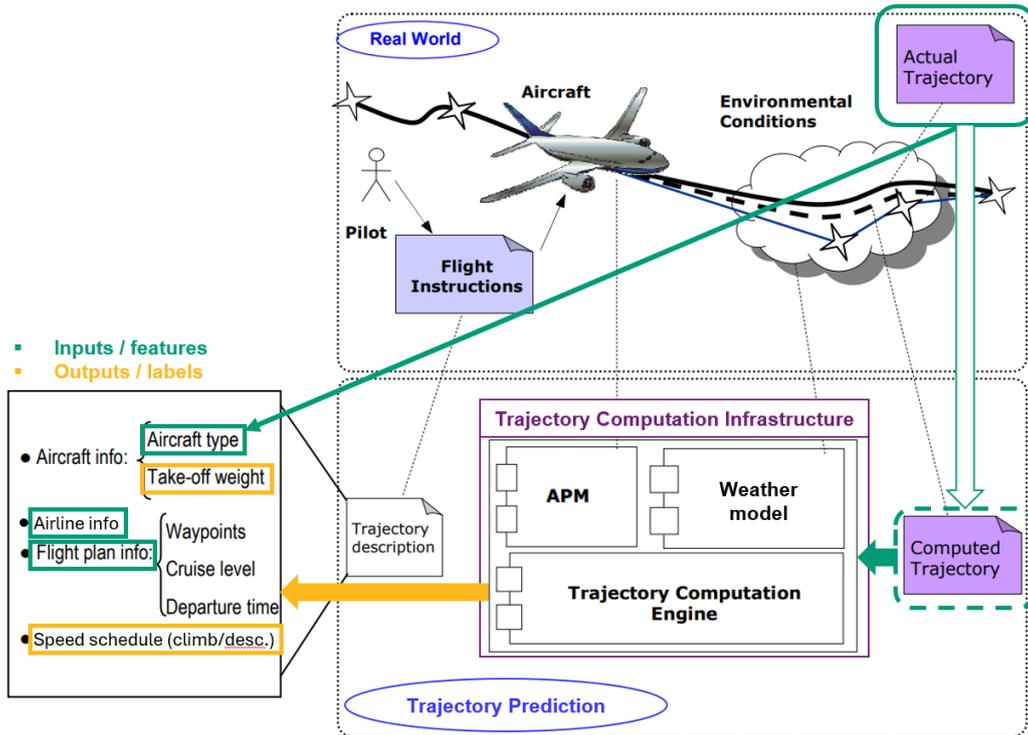


Figure 7.1: Reverse engineering methodology for deducing unknown TOW and speed profiles. EUROCONTROL.

When rearranging Equation 3.2, the aircraft mass can be computed as follows:

$$m = \frac{T - D}{g_0} \cdot \frac{1}{h} \cdot v \cdot ESF \quad (7.1)$$

Equation 7.1 is used to calculate the aircraft mass at a specific moment in time over the climb or descent phase. All the required parameters on the right-hand side are either known from ADS-B data or modelled using BADA's TCL equations.

The thrust T is computed using Equation 3.6, with the adapted percentage of maximum climb thrust c and the maximum climb thrust $T_{\max \text{ climb}}$ from Equation 3.5. Recall that $T_{\max \text{ climb}}$ is a function of altitude and temperature correction from the ISA. The altitude is known from both FR24 and Aireon datasets in Tables 5.1 and 5.2 respectively.

Aerodynamic drag D is calculated using Equation 3.8, where the density ρ is based on its reference altitude from Table 5.1 and/or Table 5.2, the TAS v is computed from the ground speed given by both ADS-B datasets, and S is a constant value stored in BADA for each aircraft type. To compute the TAS,

Equation 7.2 is used, where v_{ground} is the ground speed and v_{wind} is the wind speed taken from the GRIB files [76].

$$v = v_{\text{ground}} - v_{\text{wind}} \quad (7.2)$$

Finally, to compute the drag coefficient C_D , Equation 3.9 is used, in which all required inputs are stored in BADA. These will vary depending on the aircraft type and flight phase, as described in Section 3.4.1. When taking the LTO cycle as reference, the clean configuration will be used for high lift devices and landing gear positions after take-off has been completed. Therefore during climb, δ_{HL} and δ_{LG} are equal to zero.

Next, the rate of climb or descent \dot{h} is taken directly from the ADS-B reports in Tables 5.1 and 5.2, and the ESF is computed with Equation 3.3, where the rate of change of TAS with altitude dv/dh can be modelled from consecutive position reports.

Once the aircraft mass at a set moment in time is known, Equation 3.4 will be used to deduct the initial mass at take-off and consequently the TOW. In this equation, the fuel flow F is modelled as shown in Equation 3.7, where T and v are computed as explained in the previous paragraphs. To avoid large errors in TOW results, the time intervals taken should not exceed a margin of three ADS-B reports, and preferably, Equation 3.4 should be applied recursively to two consecutive timestamps until reaching take-off conditions.

This trajectory augmentation process is repeated for all the trajectories with which the ML will be trained. It is crucial to have an associated TOW to each flown trajectory, its corresponding FPL, and the associated FTFM-predicted trajectory for the second step of the methodology.

7.2.2. Machine Learning Model Development

Once a reasonably sized dataset of augmented trajectories is built, the next step is to develop and train a ML model to predict the desired labels (TOW and climb/descent speed profiles) from known features (e.g. aircraft type, operator, Aerodrome of Departure (ADEP)/Aerodrome of Destination (ADES), etc.).

Regarding feature selection, all the parameters listed in Table 5.3, Table 5.4, Table 5.6, and the wind speed forecasted from the GRIB files detailed in Section 3.4.3 will be used. This is depicted in the flowchart of Figure 7.2, in which the methodology is summarised.

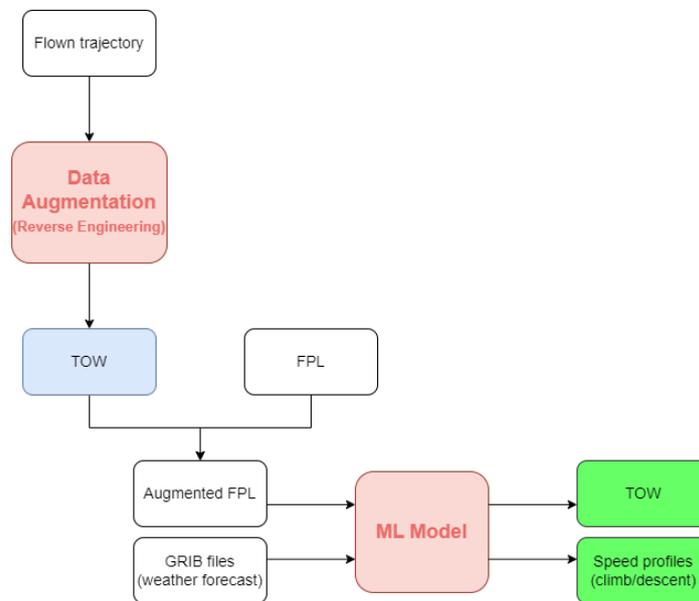


Figure 7.2: Data flow within the project - methodology overview. Own work.

The selected features are essentially FPL or FPL-deduced data and weather data. In the first part of the methodology, represented by the vertical data flow in Figure 7.2, when the TOW is deduced

from the flown trajectory, it is also associated with the respective FPL of the flight. In this way, the trajectory augmentation process creates a dataset of augmented trajectories and their corresponding FPLs. For the ML model input, the only value of interest from the augmented flown trajectories is the TOW, therefore, the latter is linked with the flight's accompanying FPL to generate an "augmented FPL". This notion refers to the dataset of FPL data, FTFM-predicted trajectory, and their associated TOW.

Due to the abundance of input features, permutation importance will be used to rank the decisiveness in obtaining the output labels. This approach was also taken in the *Climb'n Descent* project. Permutation importance is an inspection technique for ML models trained with tabular data. It is defined as the decrease in a model's score when a single feature value is randomly shuffled [67]. This allows to break the formed relation between the feature and the desired label and the model's score indicates its dependence on the feature.

Regarding the ML method, George A. Vouros et al. [52] found GBDTs to give the lowest MAE for CI and PL predictions, followed by neural networks. In Gradient Boosting, new decision trees are added to the model to correct the residual error of the existing model [77]. Each decision tree is created using a greedy search procedure to select split points that best minimise an objective function. A variation of the Gradient Boosting algorithm, namely Stochastic Gradient Boosting, is used by Alligier and Gianazza [54]. In Stochastic Gradient Boosting, at each iteration, a sub-sample of the training data is drawn at random (without replacement) from the full training dataset [78]. The randomly selected subsample is then used, instead of the full sample, to fit the base learner. This same benefit can be used to reduce the correlation between the trees in the sequence in gradient boosting models. Alligier et al. [57] also found that the Stochastic Gradient Boosting Tree algorithm achieved the lowest error. Based on the ML methods applied in the literature, GBDTs seem to be the most effective and least error-inducing algorithm. As a result, it will be the main approach for the ML model. The working principle behind this method was detailed in Section 3.5, along with implementation details for this project.

Although this other algorithm family was not treated in Section 3.5, Xinyu He et al. [59] implemented a MLPNN, a feed-forward artificial neural network, which demonstrated an improved prediction accuracy of the aircraft mass, lower errors, and better generalisation capability of the proposed method compared to state-of-the-art regression models. While less research suggested to use neural networks for the prediction of aircraft operational parameters, this approach could also be explored, provided sufficient timing.

7.2.3. Potential Applications of MSc Thesis Results

Following the main methodology of the MSc Thesis, it is also interesting to highlight the potential applications of the obtained results. A summarised flowchart is given in Figure 7.3, in which ECTL's particular interest is illustrated.

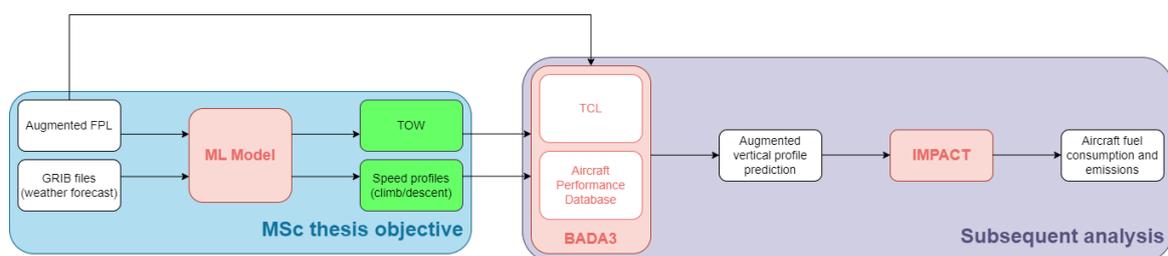


Figure 7.3: Potential application of MSc Thesis results for sustainability - data flow overview. Own work.

The outcomes of the above architecture are divided in two main blocks: first, those answering the main research question of the MSc thesis on the left, and second, those that will serve for subsequent analysis of the MSc thesis results on the right. Regarding the first block, the expected results are the correct prediction of TOW and speed profiles, highlighted in green. Next, the subsequent analysis' outcomes, in which ECTL is interested, are a set of augmented vertical profile predictions and their corresponding fuel consumption and emissions estimations. It is important to highlight that the ML algorithm (gradient boosted trees and neural networks) are applied solely for predicting TOW and

speed profiles of respective aircraft. The BADA3 trajectory computation infrastructure may then feed with these operational parameters for more accurate information with respect to existing FPLs. Once the augmented trajectories are computed, they can be used as input for IMPACT to obtain better aircraft fuel consumption and emissions predictions.

7.3. Case Studies

This section briefly details the case studies considered in the MSc thesis. In terms of timeline, the past three years of data will be considered each time.

To introduce variety and to extend the capabilities of the built model, it is essential to train it with different sets of data. For this reason, a set of three airports have been selected, at which different types of operating airlines prevail:

1. Amsterdam Airport Schiphol: legacy carrier hub
2. Brussels South Charleroi Airport: mainly low-cost carriers operating (mostly Ryanair)
3. Düsseldorf Airport: cargo carriers and operations

At the start, only the first case study will be used to train the model. In doing so, only flights coming from or going to Amsterdam Airport Schiphol will be considered. Note that from the entire case study dataset, 80% of the data will be used for training the model, while the remaining 20% is left for verification purposes.

Once the model is calibrated for this case study, there are two options of continuation. First, one can train the same algorithm with the next case studies, namely Brussels South Charleroi Airport and Düsseldorf Airport, to encompass all types of aircraft carriers' patterns for TOW and speed profiles. The other option is to train three separate models for the three case studies. In this case, these would be airline type-tailored models for legacy, low-cost, and cargo carriers. Both options will be explored to determine the least error-inducing one for predicting TOW and speed profiles.

Aircraft types are also an important aspect of the flights in the dataset. To comply with the majority of the traffic in NM, the 15 most flown aircraft types will be considered, accounting for almost 70% of traffic in the ECAC area in 2022. The ranking is done based on the number of flights operated. As these vary every year, three rankings will be used for the last three years of data. These are listed below in Tables 7.1 to 7.3 from 2020 to 2022 rankings, respectively. Notice that the top five most-operated aircraft are equivalent for all three years considered.

Table 7.1: Aircraft coverage report 2020 ranking, ECAC area.

Rank	ICAO ID	Aircraft full name	Percentage of traffic [%]	Cumulative traffic [%]
1	B738	BOEING 737-800	16.782715	16.782715
2	A320	AIRBUS A-320	12.838578	29.621293
3	A319	AIRBUS A-319	5.752930	35.374223
4	A20N	AIRBUS A-320neo	3.935992	39.310215
5	A321	AIRBUS A-321	3.458093	42.768308
6	E190	EMBRAER ERJ-190-100	2.234077	45.002384
7	B77W	BOEING 777-300ER	2.156667	47.159051
8	DH8D	DE HAVILLAND CANADA DHC-8-400 Dash 8	2.148522	49.307573
9	AT76	ATR ATR-72-600	1.606272	50.913845
10	AT75	ATR ATR-72-500	1.586052	52.499896
11	B789	BOEING 787-9 Dreamliner	1.417391	53.917287
12	B734	BOEING 737-400	1.401644	55.318930
13	B737	BOEING 737-700	1.357212	56.676142
14	A21N	AIRBUS A-321neo	1.261347	57.937489
15	E195	EMBRAER 195	1.205300	59.142790

Table 7.2: Aircraft coverage report 2021 ranking, ECAC area.

Rank	ICAO ID	Aircraft full name	Percentage of traffic [%]	Cumulative traffic [%]
1	B738	BOEING 737-800	17.683149	17.683149
2	A320	AIRBUS A-320	12.225744	29.908893
3	A319	AIRBUS A-319	5.082916	34.991809
4	A20N	AIRBUS A-320neo	4.994396	39.986205
5	A321	AIRBUS A-321	3.122625	43.108831
6	A21N	AIRBUS A-321neo	2.437878	45.546709
7	B77W	BOEING 777-300ER	2.126162	47.672871
8	E190	EMBRAER ERJ-190-100	2.125713	49.798584
9	AT76	ATR ATR-72-600	1.771957	51.570541
10	B789	BOEING 787-9 Dreamliner	1.593108	53.163649
11	AT75	ATR ATR-72-500	1.380213	54.543861
12	A333	AIRBUS A-330-300	1.340687	55.884548
13	DH8D	DE HAVILLAND CANADA DHC-8-400 Dash 8	1.340463	57.225011
14	E195	EMBRAER 195	1.235938	58.460949
15	B734	BOEING 737-400	1.225508	59.686457

Table 7.3: Aircraft coverage report 2022 ranking, ECAC area.

Rank	ICAO ID	Aircraft full name	Percentage of traffic [%]	Cumulative traffic [%]
1	B738	BOEING 737-800	18.629006	18.629006
2	A320	AIRBUS A-320	15.365513	33.994519
3	A319	AIRBUS A-319	5.598492	39.593010
4	A20N	AIRBUS A-320neo	5.007694	44.600705
5	A321	AIRBUS A-321	4.029483	48.630187
6	A21N	AIRBUS A-321neo	3.223739	51.853927
7	B38M	BOEING 737-8	2.930594	54.784520
8	AT76	ATR ATR-72-600	2.298983	57.083503
9	E190	EMBRAER ERJ-190-100	2.281057	59.364560
10	B77W	BOEING 777-300ER	1.772847	61.137407
11	B789	BOEING 787-9 Dreamliner	1.474979	62.612385
12	A333	AIRBUS A-330-300	1.401738	64.014123
13	E195	EMBRAER 195	1.357802	65.371926
14	CRJ9	CANADAIR CL-600 Regional Jet CRJ-900	1.219544	66.591470
15	AT75	ATR ATR-72-500	1.093464	67.684935

7.4. Verification & Validation

Once the model is functional and provides reasonable results, verification and validation processes need to be undertaken. For verification, besides the remaining 20% of the case study dataset, Alligier and Gianazza [54] published their ML code for predicting TOW and speed profiles on GitHub [56]. A limitation to note about this model is that it is only applicable to the climb phase.

For Validation, the actual TOW values used in the *Climb'n Descent* project will be used. Although these account for circa 30% of flights in the NM area over a period of three years, it is still valuable data for validation purposes. Another validation source will be given by an ASU colleague working part-time as a pilot for Iberia, the Spanish flag-carrier airline. He will provide values of aircraft operational weight for his flights over the past two years. This data is taken from FPLs given by Iberia. Starting October 2023 until April 2024, actual TOW data displayed by the Flight Management System (FMS) will also be gathered.

7.5. Work Breakdown

As the technical aspects of the project methodology have been detailed, it is time to introduce the logistical approach. The MSc thesis project is planned according to a five-day working week with eight hours of work per day. Regarding holidays, a total of 24 leave days are included in the contract, excluding Belgian national holidays which amount to 15 extra days. The leave days will be grouped into five separate weeks of holidays spread across the contractual year. A Gantt chart, presented in Figure A.2 in Appendix A, was constructed to have a general overview of the project milestones, tasks, and holidays. For that the general *Sustainable Air Transport (SAT)* MSc thesis structure, shown in Figure A.1, was followed. A workflow diagram is also presented in Figure 7.4 below, with each block corresponding to a particular work package.

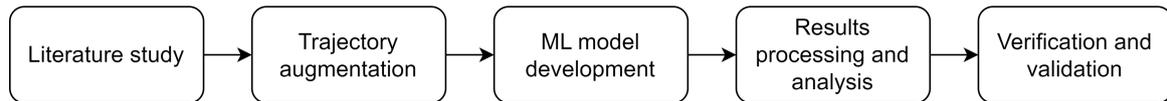
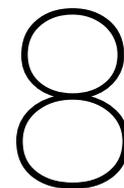


Figure 7.4: Workflow diagram of MSc thesis project. Own work.

The work packages' names are straightforward and clearly correspond to the data flow previously described in Figure 7.2. The first step is to approach the problem from an academic perspective, with the goal of acquiring theoretical knowledge on the matter, understanding current state-of-the-art ML methods used, and getting acquainted to how the approach to the problem advanced throughout the years. All this information, including past internal ECTL projects on the subject, is summarised in this literature study report. The next two blocks of the workflow diagram reflect the methodology described in Section 7.2, from the trajectory augmentation process based on reverse engineering flown trajectories, to the development of the ML model and the implementation of GBDTs. Finally, the last two blocks are concerned with post-computational analyses of results and verification and validation processes, explained in Section 7.4.

The current knowledge of the author is also considered in the planning. Although they have successfully developed a number of Python models individually, the experience in regards to ML modeling is limited. Furthermore, a transition period of two weeks was taken into account to allow the author to get acquainted with ECTL's developers' environment and to understand the structure of the data and tools that will be used. Finally, three weeks of margin are accounted in the planning, allowing for potential delays and model iterations.



Conclusion

The complexity in predicting aircraft take-off weight is brought up by the lack of data for its value. This excludes the possibility of training machine learning algorithms with raw data, making the approach to the problem less straightforward. While speed is a known parameter, measured with on-board instruments, accurately predicting the speed profiles during climb and descent is more challenging than doing so for cruise flight. This is due to the dynamic nature of the vertical profiles during these phases and the operational uncertainty brought up by flight plan deviations.

The goal of this project is to develop a machine learning tool to predict aircraft take-off weight and speed profiles during climb and descent. To do so, a two-step approach is taken, which consists in a data augmentation process followed by the development of a machine learning model. The latter is comprised of a gradient boosting decision trees algorithm, which has been found to be the most effective at predicting such parameters, based on previous literature. The data augmentation process consists in reverse engineering the take-off weight from a set of actual flown trajectories, and generating a sufficiently large dataset with such augmented trajectories for training the ML model with the required input features. These include flight plan data, their associated predicted trajectories and deduced take-off weights, as well as weather data. Once the model is functional, future potential applications may be explored.

As this MSc Thesis is done in collaboration with the Aviation Sustainability Unit at EUROCONTROL, they are interested in potential fuel savings improvement. Trajectory computation infrastructures rely on the input data they are fed with. Thus by improving the estimations of aircraft take-off weight and speed profiles during climb and descent, possibly more accurate trajectory predictions could be obtained for these phases. Once these predictions are computed, they can be used to estimate the associated fuel consumption and emissions. Therefore, improving the estimation of unknown operational parameters could indirectly refine the prediction of fuel used during climb and descent. This could potentially increase fuel savings, thus touching upon sustainability aspects.

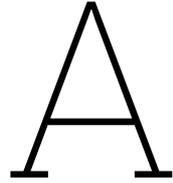
References

- [1] EUROCONTROL. *EUROCONTROL Forecast Update 2022-2024*. June 2022. URL: <https://www.eurocontrol.int/publication/eurocontrol-forecast-update-2022-2024>.
- [2] Airbus. *Global Market Forecast 2022-2041*. 2022. URL: <https://www.airbus.com/en/products-services/commercial-aircraft/market/global-market-forecast>.
- [3] EUROCONTROL. *EUROCONTROL Aviation Outlook 2050*. Apr. 2022. URL: <https://www.eurocontrol.int/publication/eurocontrol-aviation-outlook-2050>.
- [4] ICAO. *ICAO forecasts complete and sustainable recovery and growth of air passenger demand in 2023*. Feb. 2023. URL: <https://www.icao.int/Newsroom/Pages/ICAO-forecasts-complete-and-sustainable-recovery-and-growth-of-air-passenger-demand-in-2023.aspx>.
- [5] Gabriel D. Roy. *Advances in Chemical Propulsion*. 1st ed. Boca Raton, 2001. DOI: <https://doi.org/10.1201/9781420040685>.
- [6] Federal Aviation Administration. *Aviation Environmental Design Tool (AEDT) Version 3e Technical Manual*. May 2022. URL: https://aedt.faa.gov/Documents/AEDT3e_TechManual.pdf.
- [7] aviationfile. *What Stage of a Flight Consumes the Most Fuel?* Mar. 2022. URL: <https://www.aviationfile.com/what-stage-of-a-flight-consumes-the-most-fuel/#:~:text=If%5C%20we%5C%20look%5C%20at%5C%20the%5C%20the%5C%20longest%5C%20time%5C%20and%5C%20distance>.
- [8] Fiona Young-Brown. *Fuel Burn Rates for Private Aircraft*. Sept. 2015. URL: <https://www.sheparchive.com/aircraft/fuel-burn-private-aircraft.html#:~:text=Take%5C%20off%5C%20and%5C%20landing%5C%20%5C%2D%5C%20take%5C%20contributing%5C%20to%5C%20the%5C%20overall%5C%20weight>.
- [9] aviationfile. *What Stage of a Flight Consumes the Most Fuel ?* 2022. URL: <https://www.aviationfile.com/what-stage-of-a-flight-consumes-the-most-fuel/#:~:text=If%5C%20we%5C%20look%5C%20at%5C%20the%5C%20total%5C%20amount%5C%20of%5C%20the%5C%20longest%5C%20time%5C%20and%5C%20distance.%5C%20Sections%5C%20of%5C%20Flight>.
- [10] Thrust Equation. July 2022. URL: <https://www1.grc.nasa.gov/beginners-guide-to-aeronautics/thrust-force/>.
- [11] EUROCONTROL. *EUROCONTROL - Home Page*. 2023. URL: <https://www.eurocontrol.int/about-us>.
- [12] ICAO. *The Postal History of ICAO - EUROCONTROL*. 2023. URL: <https://applications.icao.int/postalhistory/eurocontrol.htm>.
- [13] EUROCONTROL. *Flight information region (FIR/UIR) charts - 2023*. 2023. URL: <https://www.eurocontrol.int/publication/flight-information-region-firuir-charts-2023>.
- [14] SKYbrary. *SESAR*. 2023. URL: <https://skybrary.aero/articles/sesar>.
- [15] European Commission. *Single European Sky*. 2023. URL: https://transport.ec.europa.eu/transport-modes/air/single-european-sky_en.
- [16] European Environment Agency and European Union Aviation Safety Agency. *European aviation environmental report 2016*. Publications Office of the European Union, 2016. DOI: [doi/10.2822/385503](https://doi.org/10.2822/385503).
- [17] Airbus. *Getting to Grips with Aircraft Performance*. Feb. 2002. URL: https://aedt.faa.gov/Documents/AEDT3e_TechManual.pdf.
- [18] Pilot Institute. *Why One-Hot Encode Data in Machine Learning?* 2021. URL: <https://pilotinstitute.com/airspeed-types/>.

- [19] International Civil Aviation Organization (ICAO). *Rules of the Air*. July 2005. URL: https://www.icao.int/Meetings/anconf12/Document%5C%20Archive/an02_cons%5C%5B1%5C%5D.pdf.
- [20] Skybrary. *Flight Plan*. 2023. URL: <https://skybrary.aero/articles/flight-plan#:~:text=%28ICAO%20Annex%20%3A%20Rules%20of%20the%20Air%29%20A,in%20navigation%20and%20safe%20operation%20of%20the%20aircraft..>
- [21] Skybrary. *Flight Plan Completion*. 2023. URL: <https://skybrary.aero/articles/flight-plan-completion>.
- [22] EUROCONTROL. *Flight plan filing and management*. 2023. URL: <https://www.eurocontrol.int/service/flight-plan-filing-and-management>.
- [23] EUROCONTROL. *Base of aircraft data*. 2023. URL: <https://www.eurocontrol.int/model/bada>.
- [24] EUROCONTROL. *User Manual for the Base of Aircraft Data (BADA) Revision 3.8*. Apr. 2010. URL: https://www.eurocontrol.int/sites/default/files/library/007_BADA_User_Manual.pdf.
- [25] Damir Poles Angela Nuic and Vincent Mouillet. “BADA: An advanced aircraft performance model for present and future ATM systems”. In: (). DOI: <https://doi.org/10.1002/acs.1176>.
- [26] UAV Navigation. *Flight Envelope*. 2023. URL: <https://www.uavnavigation.com/support/kb/general/general-system-info/flight-envelope>.
- [27] Federal Aviation Administration. *Aviation Environmental Design Tool*. 2023. URL: <https://aedt.faa.gov/>.
- [28] MITRE. *Terminal Area Route Generation and Traffic Simulation*. 2023. URL: <https://www.mitre.org/our-impact/intellectual-property/terminal-area-route-generation-and-traffic-simulation-targets#:~:text=TARGETS%20is%20a%20terminal%20area%20navigation%20%28RNAV%29%20procedure,operational%20assessment%20of%20air%20traffic%20procedures%20and%20airspace..>
- [29] Amy Dusto. *Kinetics vs Kinematics: What's the Difference and Why it Matters*. 2020. URL: <https://sciencing.com/kinetics-vs-kinematics-whats-the-difference-why-it-matters-13720229.html>.
- [30] Junzi Sun, Joost Ellerbroek, and Jacco M. Hoekstra. “WRAP: An open-source kinematic aircraft performance model”. In: *Transportation Research Part C: Emerging Technologies* 98 (2019), pp. 118–138. ISSN: 0968-090X. DOI: <https://doi.org/10.1016/j.trc.2018.11.009>. URL: <https://www.sciencedirect.com/science/article/pii/S0968090X18306089>.
- [31] EUROCONTROL. *Integrated aircraft noise and emissions modelling platform*. 2023. URL: <https://www.eurocontrol.int/platform/integrated-aircraft-noise-and-emissions-modelling-platform>.
- [32] Ramon Dalmau, Marc Melgosa, and Xavier Prats. “pyBADA: Easy BADA Integration in Python for Rapid Prototyping”. In: (2018). URL: <https://upcommons.upc.edu/bitstream/handle/2117/127485/SESAR+Innovation+Days+2018.pdf;jsessionid=A7A7381EF8955302E0CD038352527B56?sequence=1>.
- [33] ECMWF. *What are GRIB files and how can I read them*. 2023. URL: <https://confluence.ecmwf.int/display/CKB/What+are+GRIB+files+and+how+can+I+read+them>.
- [34] Jeff Whitaker. *pygrib 2.1.4*. Sept. 2021. URL: <https://pypi.org/project/pygrib/>.
- [35] Alessandro Amici and Iain Russell. *cfgrib 0.9.10.4*. May 2023. URL: <https://pypi.org/project/cfgrib/>.
- [36] APROsoftware. *The Difference Between Machine Learning and Deep Learning*. 2023. URL: <https://apro-software.com/machine-learning-and-deep-learning/>.
- [37] International Society of Automation. *Artificial Intelligence and Machine Learning: Enhancing Human Effort with Intelligent Systems*. 2022. URL: <https://www.isa.org/intech-home/2022/august-2022/features/artificial-intelligence-and-machine-learning-enhan>.

- [38] Avijeet Biswal. *Top 10 Deep Learning Algorithms You Should Know in 2023*. 2023. URL: https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm#what_is_deep_learning.
- [39] IBM Data and AI Team. *AI vs. Machine Learning vs. Deep Learning vs. Neural Networks: What's the difference?* 2023. URL: <https://www.ibm.com/blog/ai-vs-machine-learning-vs-deep-learning-vs-neural-networks/>.
- [40] Angelo Sorbello. *Is Deep Learning Supervised or Unsupervised?* 2023. URL: <https://linkde.lta.com/is-deep-learning-supervised-or-unsupervised/#:~:text=Deep%20learning%20uses%20both%20supervised%20and%20unsupervised%20learning,is%20used%20for%20finding%20patterns%20in%20unlabeled%20data.>
- [41] Andrew Ng. *Machine Learning Specialization*. 2023. URL: <https://www.coursera.org/specializations/machine-learning-introduction?action=enroll>.
- [42] Gaurav. *An Introduction to Gradient Boosting Decision Trees*. 2023. URL: [https://www.machinelearningplus.com/machine-learning/an-introduction-to-gradient-boosting-decision-trees/#Learning-rate-and-n_estimators-\(Hyperparameters\)](https://www.machinelearningplus.com/machine-learning/an-introduction-to-gradient-boosting-decision-trees/#Learning-rate-and-n_estimators-(Hyperparameters)).
- [43] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, & TensorFlow*. 3rd ed. California, United States of America: O'Reilly Media, 2022.
- [44] Cory Maklin. *Gradient Boosting Decision Tree Algorithm Explained*. 2019. URL: <https://towardsdatascience.com/machine-learning-part-18-boosting-algorithms-gradient-boosting-in-python-ef5ae6965be4>.
- [45] Aratrika Pal. *Gradient Boosting Trees for Classification: A Beginner's Guide*. 2020. URL: <https://medium.com/swlh/gradient-boosting-trees-for-classification-a-beginners-guide-596b594a14ea>.
- [46] Zach. *What is Prediction Error in Statistics? (Definition Examples)*. 2022. URL: <https://www.statology.org/prediction-error-statistics/>.
- [47] Jason Brownlee. *Why One-Hot Encode Data in Machine Learning?* 2020. URL: <https://machinelearningmastery.com/why-one-hot-encode-data-in-machine-learning/>.
- [48] EUROCONTROL. *IFPS Users Manual*. Apr. 2023. URL: <https://www.eurocontrol.int/publication/ifps-users-manual>.
- [49] EUROCONTROL. *Enhanced tactical flow management system*. 2023. URL: <https://www.eurocontrol.int/system/enhanced-tactical-flow-management-system>.
- [50] EUROCONTROL. *FTFM - Filed Tactical Flight Model*. 2023. URL: <https://ansperformance.eu/acronym/ftfm/>.
- [51] EUROCONTROL. *CTFM - Current Tactical Flight Model*. 2023. URL: <https://ansperformance.eu/acronym/ctfm/>.
- [52] George A. Vouros et al. "Data-driven estimation of flights' hidden parameters". In: (2022). URL: <https://upcommons.upc.edu/handle/2117/386838?show=full>.
- [53] Ramon Dalmau-Codina et al. "A Fast and Flexible Aircraft Trajectory Predictor and Optimiser for ATM Research Applications". In: June 2018.
- [54] Richard Alligier and David Gianazza. "Learning aircraft operational factors to improve aircraft climb prediction: A large scale multi-airport study". In: *Transportation research. Part C, Emerging technologies* 96 (2018), pp. 72–95. DOI: 10.1016/j.trc.2018.08.012. URL: <https://enac.hal.science/hal-01878615>.
- [55] OpenSky. *Publication Data*. 2018. URL: <https://opensky-network.org/datasets/publication-data/>.
- [56] Richard Alligier. *Learning Aircraft Operational Factors to Improve Aircraft Climb Prediction: A Large Scale Multi-Airport Study*. 2020. URL: <https://github.com/richardalligier/trc2018>.
- [57] Richard Alligier, David Gianazza, and Nicolas Durand. "Machine Learning and Mass Estimation Methods for Ground-Based Aircraft Climb Prediction". In: *IEEE Transactions on Intelligent Transportation Systems* 16.6 (2015), pp. 3138–3149. DOI: 10.1109/TITS.2015.2437452.

- [58] Yashovardhan S. Chati and Hamsa Balakrishnan. "Statistical Modeling of Aircraft Takeoff Weight". In: (2017). URL: <https://dspace.mit.edu/handle/1721.1/115247>.
- [59] Xinyu He et al. "Data-driven Method for Estimating Aircraft Mass from Quick Access Recorder using Aircraft Dynamics and Multilayer Perceptron Neural Network". In: *ArXiv abs/2012.05907* (2020). URL: <https://api.semanticscholar.org/CorpusID:228376122>.
- [60] Federal Aviation Administration. *Advisory Circular*. 2004. URL: <https://web.archive.org/web/20190712095745/http://www.ihst.org/portals/54/Attachment%20AC120-82.pdf>.
- [61] flightrecorder.com. *Quick Access Recorder (QAR)*. 2023. URL: <http://www.flightrecorder.com/qar.html>.
- [62] Junzi Sun, Joost Ellerbroek, and Jacco Hoekstra. "Modeling and Inferring Aircraft Takeoff Mass from Runway ADS-B Data". In: June 2016. URL: https://www.researchgate.net/publication/305403975_Modeling_and_Inferring_Aircraft_Takeoff_Mass_from_Runway_ADS-B_Data.
- [63] Junzi Sun et al. "Particle filter for aircraft mass estimation and uncertainty modeling". In: *Transportation Research Part C: Emerging Technologies* (2019). URL: <https://api.semanticscholar.org/CorpusID:197454207>.
- [64] Junzi Sun, Joost Ellerbroek, and Jacco Hoekstra. "Bayesian Inference of Aircraft Initial Mass". In: June 2017. URL: https://www.researchgate.net/publication/317600026_Bayesian_Inference_of_Aircraft_Initial_Mass.
- [65] Junzi Sun, Joost Ellerbroek, and Jacco M. Hoekstra. "Aircraft initial mass estimation using Bayesian inference method". In: *Transportation Research Part C: Emerging Technologies* 90 (2018), pp. 59–73. ISSN: 0968-090X. DOI: <https://doi.org/10.1016/j.trc.2018.02.022>. URL: <https://www.sciencedirect.com/science/article/pii/S0968090X18302626>.
- [66] F. Triboulet (EUROCONTROL). *AI/ML-based Augmented Trajectory. Executive Report*. Apr. 2021.
- [67] Leo Breiman. "Random Forests". In: (2001). DOI: <https://doi.org/10.1023/A:1010933404324>.
- [68] Ian Petchenik. *Satellite-based ADS-B data now available to all Flightradar24 users*. May 2023. URL: <https://www.flightradar24.com/blog/satellite-based-ads-b-data-now-available-to-all-flightradar24-users/>.
- [69] Federal Aviation Administration. *Ins and Outs*. 2023. URL: https://www.faa.gov/air_traffic/technology/equipadsb/capabilities/ins_outs#:~:text=ADS%2DB%20ground%20stations%20are,the%20terrain%20or%20other%20obstacles..
- [70] EUROCONTROL. *EUROCONTROL Specification for Surveillance Data Exchange ASTERIX - Part 12 Category 021: ADS-B Target Reports*. 2.6. Sept. 2021. URL: <https://www.eurocontrol.int/sites/default/files/2021-12/asterix-adsbtr-cat021-part12-v2-6.pdf>.
- [71] EUROCONTROL. *EUROCONTROL's Deployment of Aireon's Space-Based ADS-B for Flow Management*. 2023. URL: <https://www.eurocontrol.int/sites/default/files/2021-06/ads-b-qna.pdf>.
- [72] Aviation Weather Centre. *TAF Overview*. 2023. URL: <https://www.aviationweather.gov/taf/help>.
- [73] LVNL. *Schiphol EAIP*. 2023. URL: <https://eaip.lvn1.nl/web/2023-04-06-AIRAC/html/eAIP/EH-AD-2.EHAM-en-GB.html>.
- [74] meta2xml. *meta2xml*. 2023. URL: <https://sourceforge.net/projects/meta2xml/>.
- [75] ICAO (Bryan Boase). *Geographical Area for which the TAF and METAR are defined*. Sept. 2008.
- [76] NASA. *Relative Velocities*. May 2021. URL: <https://www.grc.nasa.gov/www/k-12/airplane/move.html>.
- [77] Jason Brownlee. *A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning*. 2016. URL: <https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/>.
- [78] Jason Brownlee. *Stochastic Gradient Boosting with XGBoost and scikit-learn in Python*. 2016. URL: <https://machinelearningmastery.com/stochastic-gradient-boosting-xgboost-scikit-learn-python/>.



Project Planning

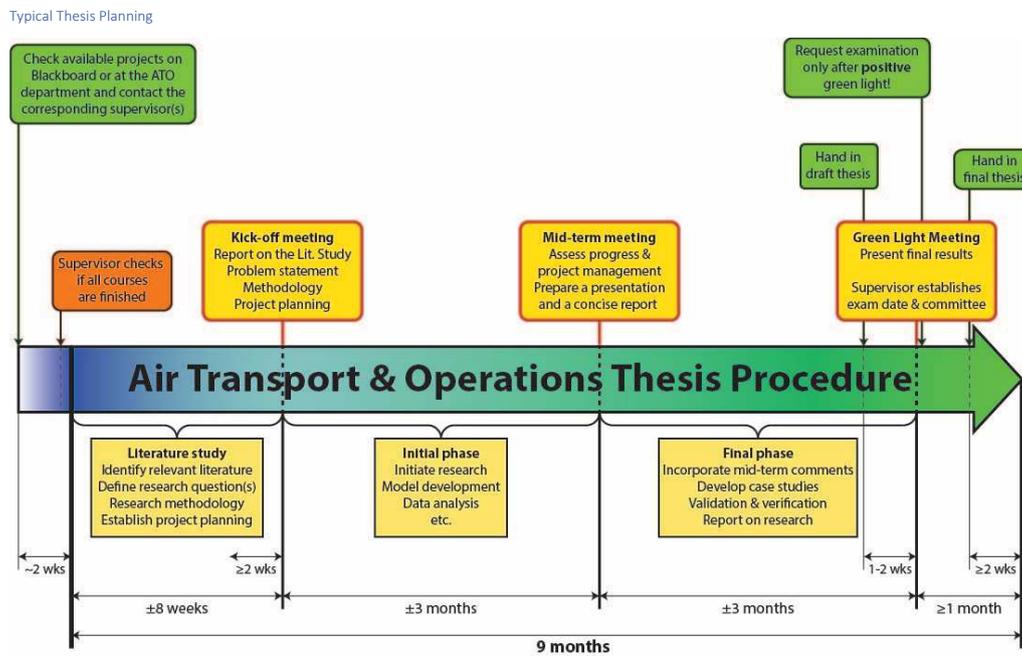


Figure A.1: SAT MSc thesis planning. TU Delft.

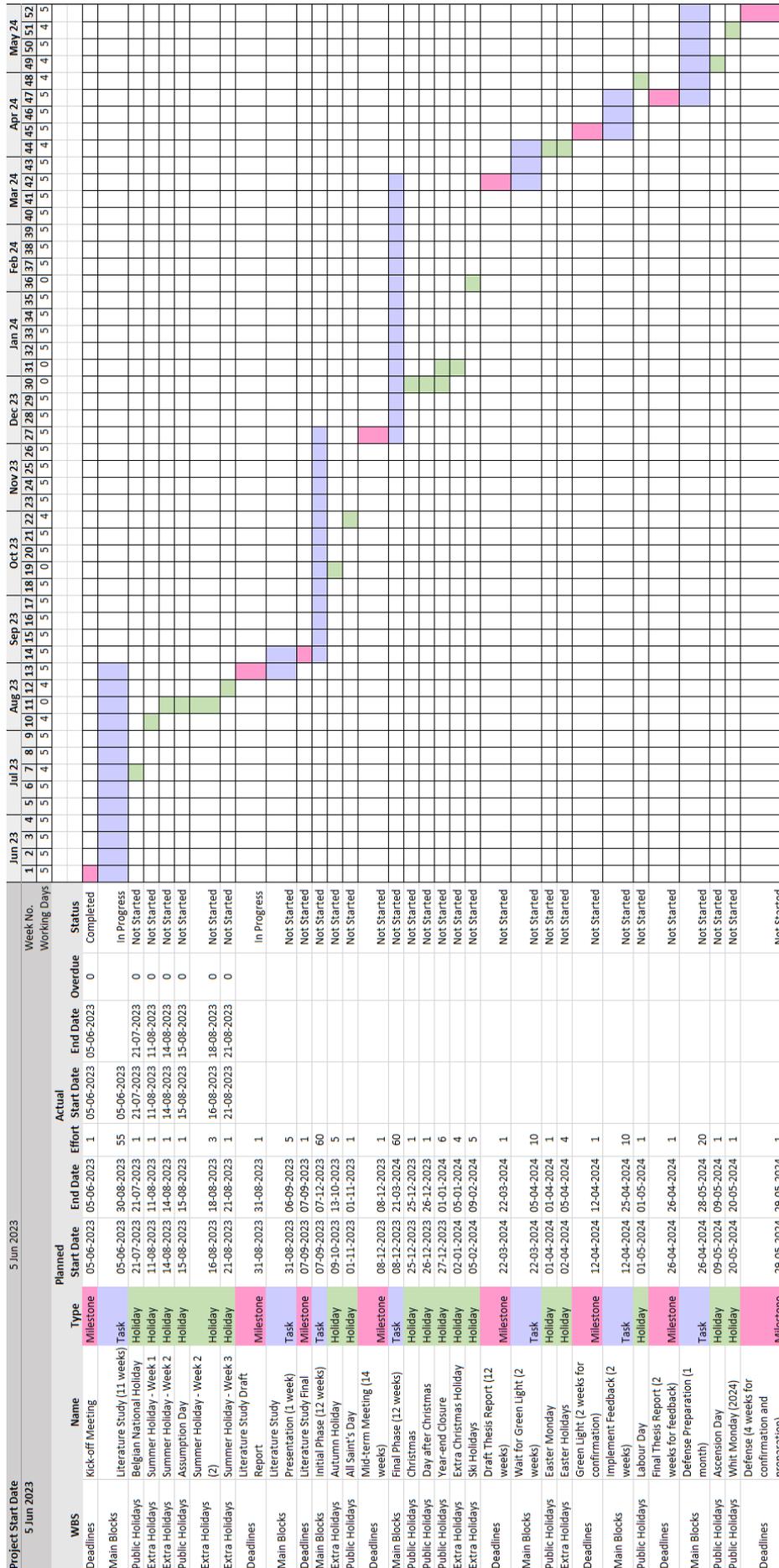


Figure A.2: Weekly Gantt chart. Own work.