

Faculty of Sciences and Technology
Department of Informatics Engineering

Condition-Based Maintenance for Diagnosis and Prognosis in Aircraft Systems

Application in the Air Bleed System of Boeing 747

Daniel Romário Gomes Azevedo

Dissertation in the context of the Master's in Informatics Engineering, Specialization in
Intelligent Systems advised by Prof. Alberto Cardoso and co-advised
by Prof. Bernardete Ribeiro and presented to the
Faculty of Sciences and Technology / Department of Informatics Engineering.

September 2019



UNIVERSIDADE DE
COIMBRA

This page is intentionally left blank.

Acknowledgements

I would like to thank ReMAP and the University of Coimbra, for giving me the opportunity of having my Thesis enrolled in the ReMAP project. Particularly, I want to acknowledge the people involved in the WP5, for all the input and feedback provided to this work.

Also, I would like to express my gratitude to my advisor, Prof. Alberto Cardoso and co-advisor Prof. Bernardete Ribeiro, for all the guidance and help granted during these last two semesters, as well as, the remaining professors from the University of Coimbra contributing to the ReMAP project.

Finally, I want to thank to my family, friends and colleagues for all the support provided during my Master's degree.

This research is supported by European Union's Horizon 2020 programme under the ReMAP project, grant No 769288.



This page is intentionally left blank.

Abstract

Aircraft maintenance is an important subject matter in the aircraft field and, as more useful information is gathered and processed, improvement in this area is valuable to the aircraft industry. In this particular topic, Condition-Based Maintenance (CBM) can be useful, as it can help predict when a failure will occur based on the component's condition. Using CBM, a Prognostics and Health Management (PHM) approach can be built with the objective of identifying the degradation behavior of the aircraft equipment, and anticipating possible system failures by predicting the future degradation evolution.

Therefore, this work aims to develop a PHM methodology that, based on the sensors data, is capable of diagnosing the health condition of an aircraft system, reflected by the Health Indicator (HI) value, and predict its future behavior, resulting in the computation of the Remaining Useful Life (RUL). The PHM approach is applied distinctively to three different aircraft systems, depending on the system data characteristics, different machine learning techniques are applied. An exploratory work is performed on two systems, the Turbofan engine system and the Brakes system, with the goal of testing different techniques for the diagnosis and prognosis of the systems condition.

The main contribution of this Thesis results from the work applied in the Air Bleed system of Boeing 747, which contains a significant complexity embedded. In this particular system, a new data driven technique for the HI computation is proposed, which is based on the analysis of time domain features, namely the mean and standard deviation, of the raw sensors data.

The scope and aim of the PHM development fits within the scope of the H2020 ReMAP project. Furthermore, some of the systems sensors data, namely the Brakes and the Air Bleed data were provided by the ReMAP project.

Regarding the obtained results, particularly in the Air Bleed system, these are interesting and very promising. The formulation used for the computation of the HI was positively received by the airlines' engineers involved in ReMAP, due to its innovative content. Furthermore, it is expected that this approach can be applied to other aircraft systems and, in a near future, be integrated in the maintenance plan, as a valid contribution for the execution and planning of aircraft maintenance routines.

Keywords

Aircraft Maintenance, Artificial Intelligence, Condition Based Maintenance, Health Indicator, Machine Learning, Prognostics and Health Management, Remaining Useful Life

Resumo

A manutenção de aviões é um tema importante no campo aeronáutico e, à medida que mais informação útil é recolhida e processada, melhoramentos nesta área são valiosos para a indústria da aviação. Neste tópico em particular, CBM pode ser útil, já que pode ajudar a prever quando é que uma falha vai ocorrer baseando-se na condição de saúde dos componentes de avião. Usando CBM, uma abordagem PHM pode ser elaborada com o objetivo de identificar o comportamento de degradação de um componente do avião, e antecipar a ocorrência de falhas através da previsão da futura degradação das peças.

Deste modo, este trabalho pretende desenvolver uma metodologia PHM que, com base nos dados de sensores, consiga diagnosticar a condição de saúde de um sistema de avião, refletido no valor de HI, e prever o seu comportamento futuro, resultando na computação da RUL. A abordagem PHM é aplicada distintamente em 3 sistemas de avião diferentes, dependendo das características dos respetivos dados, diferentes técnicas de *Machine Learning* são aplicadas. Trabalho exploratório é realizado em 2 dos sistemas, o sistema do motor do *Turbofan* e o sistema de Travões, com o objetivo de testar diferentes técnicas para o diagnóstico e prognóstico da condição de saúde dos sistemas.

A principal contribuição desta Tese resulta do trabalho aplicado sobre o sistema *Air Bleed* do Boeing 747, que possui uma complexidade significativa. Neste sistema em particular, uma nova técnica para a computação do HI é proposta, baseando-se na análise de *features* no domínio do tempo (média e o desvio padrão) dos dados em bruto dos sensores do *Air Bleed*.

O âmbito e objetivo do desenvolvimento do sistema PHM insere-se no âmbito do projeto ReMAP H2020. Como tal, os dados de certos sistemas de avião, nomeadamente dos Travões e do *Air Bleed*, foram disponibilizados no contexto do ReMAP.

Relativamente aos resultados obtidos, em particular no sistema *Air Bleed*, estes são interessantes e promissores. A formulação utilizada para a computação do HI foi recebida positivamente pelos engenheiros de aviação envolvidos no ReMAP, devido ao seu teor inovativo. É expectável que esta abordagem possa ser aplicada a outros sistemas de avião e que, no futuro, possa ser aplicada em cenários reais, como um contributo válido para a execução e planeamento das tarefas de manutenção em aviões.

Palavras-Chave

Manutenção em Aviões, Inteligência Artificial, Manutenção baseada no Estado de Condição, Indicador de Estado de Condição, Aprendizagem Inteligente, Prognóstico e Gestão do Estado de Condição, Vida Útil Remanescente

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 1 |
| 1.1 | Context of the Study | 1 |
| 1.2 | Problem Statement | 2 |
| 1.3 | Established Goals | 5 |
| 1.4 | Approach | 6 |
| 1.5 | Contributions and Achievements | 7 |
| 1.6 | Document Structure | 8 |
| 2 | Background | 9 |
| 2.1 | Health Indicator (HI) | 9 |
| 2.2 | Remaining Useful Life (RUL) | 9 |
| 2.3 | Prognostics and Health Management (PHM) | 10 |
| 2.4 | ATA Codes (ATA) | 10 |
| 2.5 | Flight Cycle and Flight Phase | 10 |
| 2.6 | Flight Deck Events (FDE) | 11 |
| 2.7 | Removals | 11 |
| 3 | State of the art | 13 |
| 3.1 | Current Aircraft Maintenance practices | 13 |
| 3.2 | Feature Selection | 15 |
| 3.2.1 | Filter methods | 15 |
| 3.2.2 | Wrapper methods | 16 |
| 3.2.3 | Embedded methods | 18 |
| 3.3 | Feature Reduction | 20 |
| 3.3.1 | PCA technique | 20 |
| 3.3.2 | LDA technique | 21 |
| 3.3.3 | SOM technique | 21 |
| 3.4 | Health Indicator (HI) extraction | 22 |
| 3.4.1 | Physical Health Indicator (PHI) | 23 |
| 3.4.2 | Virtual Health Indicator (VHI) | 23 |
| 3.4.3 | HI Computation | 23 |
| 3.5 | RUL computation | 30 |
| 3.5.1 | Engines | 30 |
| 3.5.2 | Power systems | 31 |
| 3.5.3 | Structural Components | 31 |
| 3.6 | Prognostics and Health Management | 32 |
| 3.6.1 | Model-based methods | 32 |
| 3.6.2 | Data-driven methods | 33 |
| 3.6.3 | Hybrid Prognostic approach | 47 |
| 3.6.4 | Types of learning | 47 |
| 3.7 | Performance Metrics | 48 |

| | |
|---|------------|
| 4 Proposed PHM Approach | 49 |
| 4.1 Datasets | 49 |
| 4.2 Data preprocessing | 50 |
| 4.2.1 Outliers detection and accommodation | 50 |
| 4.2.2 Flight Categorization | 51 |
| 4.3 Feature selection | 51 |
| 4.4 Health Indicator (HI) computation | 52 |
| 4.5 RUL computation | 53 |
| 5 Exploratory Case Studies | 55 |
| 5.1 Turbofan Engine System | 55 |
| 5.1.1 Dataset Structure | 56 |
| 5.1.2 Preprocessing of Turbofan Engine data | 58 |
| 5.1.3 Feature Selection | 59 |
| 5.1.4 HI computation | 60 |
| 5.1.5 RUL prediction | 61 |
| 5.1.6 Results obtained | 64 |
| 5.2 Brakes System | 66 |
| 5.2.1 Dataset Structure | 66 |
| 5.2.2 Preprocessing of Brakes data | 68 |
| 5.2.3 Feature Selection | 68 |
| 5.2.4 HI computation | 68 |
| 5.2.5 RUL prediction | 69 |
| 6 Diagnosis and Prognosis on the Air Bleed System | 71 |
| 6.1 Air Bleed data analytics | 71 |
| 6.1.1 Dataset structure | 72 |
| 6.1.2 Preprocessing of Air Bleed data | 77 |
| 6.1.3 Feature Selection | 87 |
| 6.2 Diagnosis of the system health condition | 88 |
| 6.2.1 HI Formulation | 89 |
| 6.2.2 Results of the HI algorithm | 96 |
| 6.3 Prognosis of the system health condition | 101 |
| 6.3.1 Identification of the Removal Intervals | 102 |
| 6.3.2 Computation of the RUL interval | 103 |
| 6.3.3 Results of the RUL algorithm | 105 |
| 6.4 Conclusions regarding the Diagnosis and Prognosis | 108 |
| 7 Conclusion | 111 |
| A Contribution and involvement in the ReMAP | 119 |
| B RUL approaches applied in the Turbofan Engine dataset | 121 |
| B.1 Similarity based Prognostics Approach | 121 |
| B.1.1 Training | 122 |
| B.1.2 Testing | 123 |
| B.2 Neural Network Based Approach | 125 |
| B.2.1 Data preprocessing | 125 |
| B.2.2 Training | 127 |
| B.2.3 Testing | 128 |
| B.3 Extrapolation based Prognostics Approach | 128 |
| B.3.1 Training | 128 |

| | |
|--|------------|
| B.3.2 Testing | 128 |
| C Sensors values of a case scenario considered normal | 131 |
| References | 135 |

This page is intentionally left blank.

Acronyms

- AE** Acoustic Emission. 22, 25–27
- AI** Artificial Intelligence. 35
- ANN** Artificial Neural Network. 21
- AR** Autoregressive. 27
- ARC** Abnormal Runway Contact. 4
- BMU** Best Matching Unit. 22
- C-MAPSS** Commercial Modular Aero-Propulsion System Simulation. 55
- CBM** Condition-Based Maintenance. v, vii, xv, 6, 7, 13–15, 32, 49, 111, 119
- CFIT** Controlled Flight Into Terrain. 4
- CNN** Convolutional Neural Networks. 29
- COG** Center of Gravity. 42
- DFT** Discrete Fourier Transform. 25
- EMD** Empirical Mode Decomposition. 27
- EoL** End of Life. 129
- FBM** Failure Based Maintenance. 14
- FDE** Flight Deck Events. 11, 72, 74, 91–93, 96–98, 100–103, 108, 109, 131
- FF** Feedforward. 36
- FFT** Fast Fourier Transform. 25, 26
- GA** Genetic Algorithm. 17
- HI** Health Indicator. v, vii, xv–xvii, 7, 9, 13, 15, 22–25, 27–30, 49, 51–53, 60–63, 65, 68, 69, 71, 74, 88–90, 92–103, 108, 109, 111–115, 119, 122–124, 127–129
- HMM** Hidden Markov Model. xv, 42, 44, 45
- LDA** Linear Discriminant Analysis. 20, 21
- LOC-I** Loss of Control in Flight. 4
- MLP** Multilayer Perceptron. xvi, 61, 65, 125, 127, 128
- MOM** Mean Of Maxima. 42

-
- NFF** No Fault Found. 11
- NMF** Non-Negative Matrix Factorization. 27
- NN** Neural Network. xvi, 25, 34–36, 63
- OSH** Optimally Separating Hyperplane. 38, 39
- PCA** Principal Component Analysis. 20, 23
- PHI** Physical Health Indicator. 22, 23, 28, 52
- PHM** Prognostics and Health Management. v, vii, 5–7, 10, 13, 32, 47–49, 53, 71, 108, 111, 112, 114
- PM** Preventive Maintenance. 14
- PSO** Particle Swarm Optimization. 17
- RBF** Radial Basis Function. 40
- RBFN** Radial Basis Function Network. 26, 36
- RCM** Reliability Centered Maintenance. 14
- RE** Runway Excursion. 4
- ReMAP** Real-time Condition-based Maintenance for Adaptive Aircraft Maintenance Planning. 6, 7
- RMS** Root Mean Square. 23–25, 27, 28, 42
- RMSE** Root Mean Squared Error. 64
- RNN** Recurrent Neural Network. 36
- RUL** Remaining Useful Life. v, vii, xvi, xvii, 6, 7, 9, 10, 13, 15, 22, 23, 25, 26, 28–30, 32, 33, 47–49, 53, 55–59, 61–71, 101–109, 111–114, 119, 121–125, 128, 129
- SBS** Sequential Backward Selection. 18
- SCF** System/Component Failure or Malfunction. 4
- SFB** Sensitive Frequency Band. 25, 26
- SFFS** Sequential Floating Forward Selection. xv, 18
- SFS** Sequential Feature Selection. 18
- SOM** Self Organizing Map. 20, 21, 23, 27
- STFT** Short Time Fourier Transform. 26, 27
- SVM** Support Vector Machine. 35, 38–40
- TPM** Total Productive Maintenance. 14
- VHI** Virtual Health Indicator. 22, 23, 28, 52
- WT** Wavelet Transform. 26, 27

List of Figures

| | | |
|------|---|----|
| 1.1 | Percentage of hull losses by accident category 1998-2017 (from <i>A Statistical Analysis of Commercial Aviation Accidents 1958-2017</i> [9]) | 4 |
| 2.1 | Fault behavior (from <i>Prognostics, The Science of Prediction</i> [19]) | 10 |
| 3.1 | Wrapper method approach (from <i>Wrappers for feature subset selection</i> [42] | 17 |
| 3.2 | Sequential Floating Forward Selection (SFFS) operation chart (from <i>A survey on feature selection methods</i> [35] | 18 |
| 3.3 | Principal Components (from <i>Basics and Examples of Principal Component Analysis (PCA)</i> [57] | 20 |
| 3.4 | Linear Discriminant (from <i>Discriminant Analysis</i> [59]) | 21 |
| 3.5 | Self-Organizing Mapping (from <i>Using Complex Network Topologies and Self-Organizing Maps for Time Series Prediction</i> [63] | 22 |
| 3.6 | Health Indicator (HI) calculation (from <i>Anomaly Detection and Fault Prognosis for Bearings</i> | 27 |
| 3.7 | HI calculation (from <i>Machinery health indicator construction based on convolutional neural networks considering trend burr</i> | 29 |
| 3.8 | Model-based flowchart applied in Condition-Based Maintenance (CBM) context (from <i>Current status of machine prognostics in condition-based maintenance: a review</i> [123]) | 32 |
| 3.9 | Discrete Kalman Filter cycle (from <i>An Introduction to the Kalman Filter</i> [132]) | 34 |
| 3.10 | example of a Feedforward Neural Network (from <i>Diagnostic rule extraction from trained feedforward neural networks</i> [135]) | 36 |
| 3.11 | example of a Radial Basis Function Network (from <i>Radial basis function networks for classifying process faults</i> [137]) | 37 |
| 3.12 | example of a Recurrent Layer Network (from <i>Computational modeling of machining systems</i> [139]) | 37 |
| 3.13 | example of a Decision Tree (from <i>Induction of decision trees</i> [141]) | 38 |
| 3.14 | Pseudo code of <i>createBranch()</i> function (from <i>Machine Learning in Action</i> [143]) | 38 |
| 3.15 | example of a SVM hyperplane (from <i>Recognition of Emotions from Human Activity Using STIP Feature</i> [145]) | 39 |
| 3.16 | Fuzzy Logic System example (adapted from <i>Fuzzy Neural Networks for Real Time Control Applications</i> [150]) | 41 |
| 3.17 | Temperature Memberships (from <i>Temperature Control using Fuzzy Logic</i> [151]) | 41 |
| 3.18 | Defuzzification method (from <i>Artificial Intelligence - Fuzzy Logic Systems</i> [152]) | 42 |
| 3.19 | Hidden Markov Model (HMM) diagram (from <i>An Introduction to Hidden Markov Models and Bayesian Networks</i> [158]) | 44 |

| | |
|--|----|
| 3.20 State transition diagram | 45 |
| 3.21 example of a Bayesian Network (from <i>Machine Learning - An Algorithmic Perspective</i> [146]) | 46 |
| 4.1 Generic Remaining Useful Life (RUL) pipeline | 49 |
| 4.2 General flowchart of the outliers detection and accommodation | 51 |
| 5.1 Simplified model of C-MAPSS (from <i>Damage Propagation Modeling for Aircraft Engine Prognostics</i> [165]) | 56 |
| 5.2 Operational settings relation | 58 |
| 5.3 Relevant Sensor (from <i>A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems</i> [17]) | 60 |
| 5.4 Irrelevant Sensor (from <i>A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems</i> [17]) | 60 |
| 5.5 Multilayer Perceptron (MLP) Architecture | 61 |
| 5.6 HI curves produced for the first 10 training units | 62 |
| 5.7 Assessment of the most similar training trajectories with the test trajectory 13 | 62 |
| 5.8 RUL estimation using a Neural Network (NN) approach for Test Unit 13 | 63 |
| 5.9 RUL estimation using an Extrapolation approach for the Test Unit 13 | 64 |
| 5.10 Table with the best 10 results until 2014 (from <i>Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets</i> [163]) | 66 |
| 5.11 Brakes dataset sample | 67 |
| 5.12 Removals dataset sample | 67 |
| 5.13 Degradation trajectories of the 8 brakes positions for aircraft PH-013 | 69 |
| 5.14 RUL prognostic | 70 |
| 6.1 Aggregated phases Organization | 77 |
| 6.2 Aggregated phases division in one flight | 78 |
| 6.3 Outlier detection in sensor H53294016 | 79 |
| 6.4 Outlier detection in sensor H83395790 | 79 |
| 6.5 Pseudo code of outliers detection and accommodation algorithm | 80 |
| 6.6 Original data | 81 |
| 6.7 Categorization using partner's flight_id | 81 |
| 6.8 Data inconsistencies found in the column H51394154 | 82 |
| 6.9 Evolution of the phase number of a particular flight | 82 |
| 6.10 H20790882 - Phase Number | 83 |
| 6.11 H38450254 - Sensor 1 | 83 |
| 6.12 H20790882 - Phase Number | 84 |
| 6.13 H20790882 - Phase Number | 85 |
| 6.14 Pseudo Code of the flight labeling algorithm | 85 |
| 6.15 Original Flights Labeling | 86 |
| 6.16 Flights Labeling after filtering | 86 |
| 6.17 Correlation heatmap of the 20 features of aircraft 04388298 | 87 |
| 6.18 Correlation of the first feature of each system | 88 |
| 6.19 Correlation of the third feature of each system | 88 |
| 6.20 Correlation heat map of the 5 features of system 1 | 89 |
| 6.21 General representation of the developed algorithm | 91 |
| 6.22 Representation of the <i>std_threshold</i> meaning | 92 |
| 6.23 Case scenario with an ideal behaviour of HI | 93 |
| 6.24 HI evolution in a real case scenario | 93 |
| 6.25 HI evolution in each of the 5 sensors from the Air Bleed system | 95 |

| | |
|---|-----|
| 6.26 HI evolution in each of the 5 sensors from the Air Bleed system | 95 |
| 6.27 HI of Time Interval 1 | 97 |
| 6.28 HI of Time Interval 2 | 98 |
| 6.29 HI of Time Interval 3 | 99 |
| 6.30 HI of Time Interval 4 | 100 |
| 6.31 Removal Interval example | 103 |
| 6.32 Removal Interval 1 | 104 |
| 6.33 RUL prediction | 105 |
| 6.34 RUL prediction | 106 |
| 6.35 RUL estimated for Removal Interval 3 | 107 |
| 6.36 RUL estimated for Removal Interval 1 | 108 |
| | |
| 7.1 Proposed Work Planning | 112 |
| 7.2 Actual Work Performed | 113 |
| | |
| B.1 Steps taken in each stage (from <i>A Similarity-Based Prognostics Approach for Engineered Systems</i>) | 121 |
| B.2 HI time series obtained for training unit 1 | 123 |
| B.3 HI curves produced for the first 10 training units | 124 |
| B.4 Smoothing performed on Sensor 6 of training unit 1 | 126 |
| B.5 Evolution of the Run Indicators for the 6 Operational Regimes | 126 |
| B.6 HI sequence obtained for training unit 1 | 127 |
| | |
| C.1 Sensor values of Sensor 1 - column H35950744 | 131 |
| C.2 Sensor values of Sensor 2 - column H24264101 | 132 |
| C.3 Sensor values of Sensor 3 - column H94813608 | 132 |
| C.4 Sensor values of Sensor 4 - column H92021026 | 133 |
| C.5 Sensor values of Sensor 5 - column H68446583 | 133 |

This page is intentionally left blank.

List of Tables

| | | |
|-----|--|-----|
| 3.1 | Kernel Formulas | 40 |
| 3.2 | Fuzzy Rules | 42 |
| 5.1 | Datasets Structure and Details | 57 |
| 5.2 | Operational Settings | 58 |
| 5.3 | Sensors description | 59 |
| 5.4 | RMSE Error | 64 |
| 5.5 | Score Error | 64 |
| 6.1 | Sensors Description | 73 |
| 6.2 | Removal Intervals | 103 |
| 7.1 | Impact & Probability Analysis | 113 |

This page is intentionally left blank.

Nomenclature

In Diagnosis of Air Bleed System

| | |
|------------------------------|---|
| $\alpha_{1,j,k}$ | Coefficient that reflects the deviation in the sensors values of the aggregated phase j of flight k |
| $\alpha_{2,j,k}$ | Coefficient that reflects the variance in the sensors values of the aggregated phase j of flight k |
| $\mu_{p_j}^{s_k}$ | Average of the values of sensor k , regarding the aggregated phase j |
| $\sigma_{p_j}^{s_k}$ | Average of the standard deviation of sensor k , regarding the aggregated phase j |
| <i>Aggregated_Phase_mean</i> | Mean of the specific flight, flight phase and sensor being analyzed |
| <i>Aggregated_Phase_std</i> | Standard deviation of the specific flight, flight phase and sensor being analyzed |
| $duration_{j,k}$ | Duration, in flight hours, of the aggregated phase j of flight k |
| n | Number of flights |
| $std_threshold$ | Value that determines the degradation boundaries |

In Prognosis of Air Bleed System

| | |
|--------------|---|
| $T_{i,k}$ | HI series of flight k regarding the Training trajectory i |
| T_i | HI series of Training trajectory i |
| $T_{Test,k}$ | HI series of flight k regarding the Test trajectory |
| T_{Test} | HI series of the Test trajectory |
| w_k | Weight associated to flight k |

This page is intentionally left blank.

Chapter 1

Introduction

This report aims to present and describe the work developed in the context of the Dissertation enrolled in the Master's in Informatics Engineering presented in the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra.

In this introductory Chapter, the context of this study is presented, as well as, an explanation of the identified problem and its impact in the current aircraft industry.

Then, the approach proposed in this work for improving aircraft maintenance is described, as a consequence of the enrollment of this work in the H2020 ReMAP project [1].

Also, the contributions and achievements produced by this work are detailed and the structure of this document is specified.

1.1 Context of the Study

Over time, the complexity embedded in the equipment and systems increases in such a way that these become more susceptible to failure. These failures can have various impacts, depending on the system affected and on the cruciality of the failure. These failures can be more problematic in critical systems, and in particular, in aircraft systems.

In an aircraft, a failure raises problems in regard to reliability, availability and safety. Therefore, significant effort is required in aircraft maintenance in order to certify that the necessary conditions for aircraft operations are achieved. Aircraft maintenance is expected to prevent failures to happen, guaranteeing and improving its safety and its reliability.

Furthermore, in 1994, Tom Edwards defined the following four points, as the objectives of an effective maintenance program [2] [3].

- Ensure that the reliability and safety imparted by its design, to an aircraft are sustained, through maintenance activity.
- Restore the levels of safety and reliability when deterioration occurs.
- When inherent reliability is not adequate, obtain information for the design modification.
- Achieve the above points at the lowest cost possible.

As stated by Edwards, a relevant issue regarding aircraft maintenance is the associated costs. These costs include the value of the spare parts in inventory, that might be needed in the future, the impact of capital related to the equipment downtime and the regulatory compliance, as well as costs related to unplanned activities like parts re-purchasing and overtime costs [4].

In order to understand the impact of unplanned activities of maintenance in aircraft, some statistical facts are presented [5]:

1. In case a large aircraft flight, like Boeing 747, is canceled, it can cost the airline US \$140,000.
2. If the aircraft takes off but, due to some technical failure, the aircraft needs to land to be repaired it can cost up to US \$150,000 per hour to the airline.
3. A delay at the gate can have a cost of US \$17,000 per hour.
4. In overall, each year, US \$75-100 million are wasted in errors per airline.

It is a fact that degradation of the aircraft components happens over time and the goal of aircraft maintenance is to prevent the components reaching a state where they are likely to fail but optimizing the components' life at reduced costs.

Improvements in maintenance can reduce the occurrence and the impact of several problems like flight delays or cancellations, gate returns, personnel injuries, in-flight shut downs and maintenance rework. This will lead to huge economical savings and will also have a positive impact on the reputation and competitiveness of the airline. One way to accomplish this is to provide more useful information regarding the state of the aircraft systems in order to perform more accurate diagnostics and predictions about the aircraft condition.

1.2 Problem Statement

Presently, aircraft maintenance is highly regulated by certified authorities like the EASA (European Union Aviation Safety Agency) in Europe and the FAA (Federal Aviation Administration) in the United States of America, and have multiple rules and laws that define the way that maintenance is planned and performed.

Besides the strict regulations and the investments in this area, aircraft maintenance can be still improved.

Planning and executing maintenance in a complex system, as an aircraft, is a challenging task, due to the different levels of management that need to be synchronized. This includes the management of the inventory, where the availability of the different materials, like the spare parts, need to be in accordance with the planning and scheduling of the aircraft inspections, which, in turn, need the correct allocation of resources like space, time and people so that the performed interventions, like the assembly and disassembly of aircraft components, occur as smoothly as possible and without interfering or delaying parallel or future actions.

Besides the complexity in integrating the inventory management with the inspections planning, operations and resources scheduling, there are other problems associated with the planning and execution of aircraft maintenance, namely [6]:

- The big dimension of the maintenance projects, which involves a large number of activities and different maintenance levels.
- Huge quantity of aircraft components and materials to manage regarding different aircraft and maintenance scheduling.
- Extensive amount of data and information retrieved from systems, that need to be analysed and stored.
- Large percentage of unplanned maintenance. The unplanned maintenance corresponds to up to 50% of the work involved in maintenance, depending on the work policies followed.

Current problem areas

In what concerns current practices regarding aircraft maintenance, as reported by Premaratne Samaranayake [6], the inventory management, which includes the spare parts and materials maintenance, along with unplanned activities and the planning of maintenance actions, constitute some of the problem areas in airlines in the present days.

The inventory management plays an important role in the airlines finances, as it represents a significant investment. A study presented by the same author based on a internal report on investigation of an airline, indicates that the purchase of aircraft components and spare parts is mainly made based on its historical usage and doesn't reflect the future need during planned inspections. Due to the costs acquiring and maintaining the spare parts in inventory, some airline tend to keep their inventory to a minimum, reducing costs. Nevertheless, these shortages might be costly in case they need to re-purchase the same parts due to necessity, which brings additional costs and delays to the airlines.

The unplanned maintenance work can be also be critical and important in the maintenance routines. The fact that 50% of the developed work corresponds to unplanned actions, means that all the thought and planned scheduling regarding maintenance only affects half of the work to be executed, as the other 50% of the work is condition dependent and is executed according to the problems identified in inspections during lay-up.

With respect to the planning, scheduling of inspections and interventions on the aircraft, the following aspects should be taken in consideration [6]:

- Uncertainty of components condition. The condition state of the aircraft components is unpredictable and thus uncertainty is present when performing the lay up inspections.
- Variance of the maintenance duration. Although it can be an established plan, the maintenance, including the assembly and disassembly, is performed according to the components' condition.
- Dependency between maintenance tasks. Some tasks need to be performed in a logical order, in order to solve a specific problem.
- Different maintenance actions. Depending on the aircraft, there might be specific maintenance configurations, which affect the actions to be performed.

Due to the importance of aircraft maintenance, studies and analysis have been performed regarding this field, by the airline companies, among others.

According to a study carried out by Boeing with respect to aircraft maintenance [7], the maintenance issues are responsible for 20% - 30% of engine in-flight shutdowns and can cost up to US \$500,000 per shutdown. Also, from 2013 to 2017, IATA concluded that aircraft malfunction had a contribution of 29% to the aircraft accidents [8].

As stated by Airbus [9], regarding hull losses (when an aircraft is destroyed or damaged beyond economical repair), the System/Component Failure or Malfunction (SCF) is responsible for 10% - 15% of the accidents that occurred between 1998 and 2017, being the second major cause according to Figure 1.1.

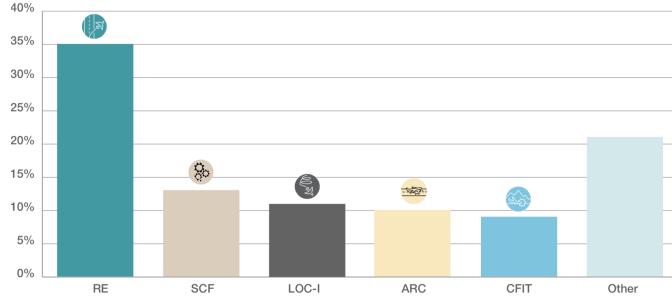
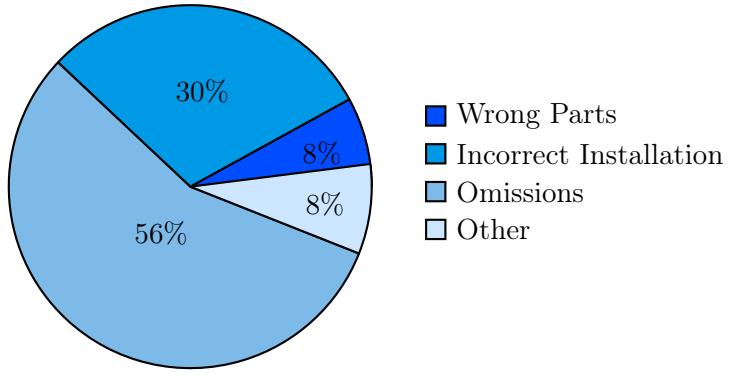


Figure 1.1: Percentage of hull losses by accident category 1998-2017 (from *A Statistical Analysis of Commercial Aviation Accidents 1958-2017* [9])

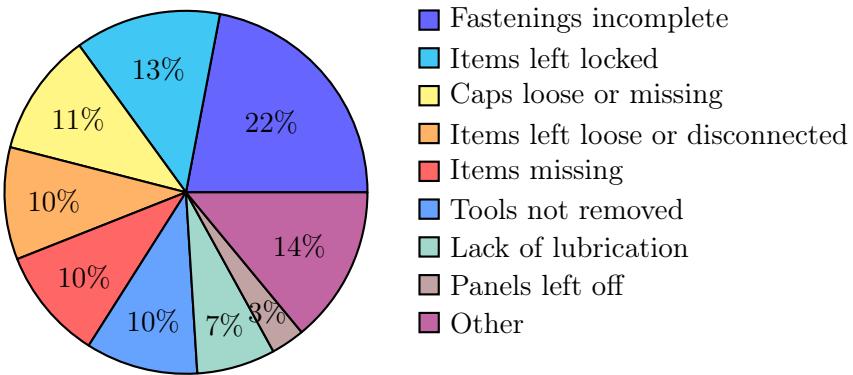
The accident categories, illustrated in Figure 1.1 are described by Airbus [9] as the following:

- **Runway Excursion (RE):** A lateral or longitudinal deviation that exceeds the surface of the track
- **System/Component Failure or Malfunction (SCF):** Failure or malfunction of an aircraft system or component that leads to an accident. SCF includes the powerplant, software and database systems.
- **Loss of Control in Flight (LOC-I):** Loss of control of the aircraft during flight, not mainly due to SCF.
- **Abnormal Runway Contact (ARC):** Hard landing that leads to an accident. The cause is not primarily due to SCF.
- **Controlled Flight Into Terrain (CFIT):** Collision during flight with some obstacles, like terrain or water, without the indication of loss of control.

Another study performed by Boeing [10], targets the causes of 122 accident occurrences, between 1989-1991, that involved human errors. These were distributed according to following graph.



In particular, the omissions percentage, is divided in the following categories [11]:



Aircraft maintenance is a critical aspect of the aircraft as a small error can lead to dramatic consequences. Therefore, aircraft maintenance should be well planned and executed. With the access of more useful and valuable information regarding the condition of the aircraft components, better diagnostics and planning can be made in aircraft maintenance, turning flights safer, more reliable and more viable regarding the costs associated.

1.3 Established Goals

The goals regarding this work are defined in alignment with the ReMAP project goals, specific of the work package involved (WP5).

The established goals are the following:

- Perform preprocessing of raw data in order to detect possible noisy data and outliers.
- Intelligent sensors data analysis in order to diagnose the system condition, using real data retrieved from airplanes.
- Develop Prognostics and Health Management (PHM) methodologies for aircraft systems that allow the prognosis of aircraft components' degradation behavior.

In detail, the aim is to analyse the raw data retrieved from the aircraft sensors specific for the component or subsystem in analysis and perform preprocessing techniques for removing eventual noise, like outliers, embedded in the data. In addition, algorithms for the

diagnostic of the system health condition must be developed in order to detect the presence of degradation or failure in the considered system. Finally, using the health condition information, machine learning methods should be used for the prediction of the system future condition. By succeeding these goals, a complete PHM methodology is developed for the diagnosis and prediction of aircraft subsystems or components condition.

1.4 Approach

A promising approach for improving the aircraft maintenance, either in terms of security and costs optimization, is the Condition-Based Maintenance (CBM). The CBM aims to focus on the system condition for planning and performing the aircraft maintenance routines, instead of fixed time intervals.

Moreover, the CBM concept and principles meet the scope of the Real-time Condition-based Maintenance for Adaptive Aircraft Maintenance Planning (ReMAP) project [1]. The H2020 ReMAP project is an European project that seeks to develop a open-source solution, the Integrated Fleet Health Management (IFHM), for aircraft maintenance where fixed-interval inspections are replaced by adaptive condition-based interventions.

Thus, this Thesis project plays a part of the ReMAP project. In this way, this work attends to develop PHM techniques, following a CBM approach, capable of diagnose the aircraft components' condition and predicting possible failures through the estimation of RULs, in order to better plan and execute aircraft maintenance. Furthermore, the implemented methods are trained and tested using data retrieved from aircraft sensors that were provided within the ReMAP context, and intelligent data analysis is performed in order to improve the methods performance, this may include data preprocessing, feature (sensors) selection and data normalization.

Therefore, and in alignment with the ReMAP project proposal regarding this specific Work Package, the implemented methodologies are focused on 2 different aspects:

1. **Diagnostic:** The Diagnostic task encompass techniques for assessing the condition of the aircraft parts of particular subsystems.
2. **Prognostic:** The Prognostic task uses the output from the diagnostic stage and comprises techniques for the prediction of the system condition in the future. The predictions are expressed by the Remaining Useful Life (RUL) of the system.

In the ReMAP, 13 different aircraft systems were identified and will be studied in the project context, namely: Cabin Air Conditioning and Temperature Control System; Cabin Pressurization and Control Systems; Engine Anti-Ice; Power Electronics Cooling systems; Nitrogen Generation System; Common Motor Start Controller; Variable Frequency Starter/Generator; Buss Power Control Unit – Generator Control Unit; Auxiliary Power Unit; Fans; Integrated Cooling System; Wheels & Brakes; and Air Bleed System.

Nevertheless, this work focus on the analysis of two specific aircraft systems within the ReMAP scope, although at different depths. First, the Brakes system, is used as a exploratory case study due to its reduced complexity and small amount of sensors data. Then, the Air Bleed system is studied as the core dataset for the training and testing of the developed approaches for the diagnostic and prognostic of the system condition due to its significant size and complexity.

Furthermore, a third system is analyzed, as an initial and exploratory work, which corresponds to the Turbofan engine, extracted from the NASA Prognostics Data Repository.

With this approach, it is expected that more useful information regarding the aircraft sub-systems condition is obtained and thus can contribute to the improvement of the efficiency of the maintenance tasks.

The resulting information regarding the diagnostic and prognostic of the systems' condition will not replace the maintenance team responsibility in assessing and diagnosing the aircraft components conditions, it will complement it. By providing more useful information regarding the systems' condition, the maintenance team will have more arguments and conditions for a more correct and accurate planning and execution of the maintenance tasks. This will lead to the possible optimization of the aircraft parts' useful life and reduce costs by avoiding fruitless interventions, while assuring the correct operation of aircraft components.

1.5 Contributions and Achievements

The outcome and conclusions retrieved from this work result in different contributions and achievements, with respect to distinctive contexts, that are going to be presented next.

In an initial stage of this work, and as a consequence of the developed exploratory work in the case studies, in particular with the NASA's dataset, a web tool for the visualization of the State of the Art methodologies applied over the dataset for the RUL prediction was created. The outcome from this work, from the methodologies point of view, resulted in the publishing of a paper entitled "*Online Simulation of Methods to Predict the Remaining Useful Lifetime of Aircraft Components*" [12] presented in the exp.at'19 Conference. In addition, the paper "*Web-based tool for predicting the Remaining Useful Lifetime of Aircraft Components*" [13] was written and published with the goal of presenting and explaining the web interface, from the graphical point of view. This web interface operation was demonstrated in a Demo session, also in exp.at'19 Conference. The web interface can be assessed in the *planum.dei.uc.pt*. Also a paper regarding the web tool developed is being written for submission in the iJOE Special Issue [14].

Still during the experimental work performed over the NASA dataset, the writing of the technical report entitled "*Benchmark of methods applied to PHM08 Challenge Dataset*" [15] helped, in an initial stage, in performing the analysis of the existing machine learning methods for the RUL prediction.

Regarding this work's role in the ReMAP project, it is enclosed in the Work Package 5 (WP5) of the ReMAP project, thus the main contributions converge with the WP5 goals.

The result of this work is a PHM methodology, based on a data driven approach, that diagnoses the system condition (expressed by the HI) and predicts the future condition, through the RUL computation.

The ReMAP proposes the implementation of a complete solution for the aircraft maintenance, following the CBM standards. Hence, the contribution from this work is expected to be integrated in a more complete and robust platform and serve as input for the execution of other important tasks planned in the ReMAP proposal, as, for example, the creation of a decision-based model for the management of maintenance scheduling tasks. More details regarding the significance and involvement of this work in the ReMAP project can be accessed in Appendix A.

1.6 Document Structure

This document is structured in 7 Chapters.

Chapter 2 raises and explains specific concepts associated with the aircraft field, that are valuable for the understanding of this work.

Chapter 3 corresponds to the description and exposition of the State of the Art regarding the tasks/steps that were used in this work. In the end, performance metrics applied to the implemented techniques are specified.

In **Chapter 4** the approach followed in this work is presented and described, by specifying and detailing the different steps that comprise the overall approach.

In **Chapter 5** the performed experimental work is described. In particular, two case studies are presented. These were carried out as exploratory work regarding the diagnostic and prognostic of aircraft components condition.

Chapter 6 details the work developed in the core dataset, the Air Bleed System. From the preprocessing to the obtained results, the multiple steps performed are described, and their results are discussed.

In **Chapter 7** a summary of the work performed is elaborated, as well as a discussion of the planning deviations and a risks analysis.

Chapter 2

Background

This Chapter 2 aims to present and describe certain core concepts associated with my work and with the aircraft field overall. Due to the fact that most of these concepts are not familiar and have a specific meaning in the aircraft context, the explanation of their meaning was found valuable and beneficial for the understanding of this work.

2.1 Health Indicator (HI)

Physical components, like aircraft parts or components, deteriorate over time due to environment conditions, wastage, failures, damage, etc. This increasing degradation has an impact on the system operation and thus the evolution of system degradation should be monitored in order to measure its effects on the system operation.

A metric used for measuring and monitoring the system condition is the Health Indicator (HI). The Health Indicator (HI) corresponds to a value that reflects the system health condition [16] and it can have different representations, like a percentage [17] or an absolute value of a specific variable [18]. Normally it is computed from the data or signals extracted from the system sensors, as they reflect the system condition.

2.2 Remaining Useful Life (RUL)

The Remaining Useful Life (RUL) is defined as the life span starting at the current time to the end of the useful life of a certain component. This term can be applied in different contexts, and in this case it is applied to aircraft. More specifically, it is used to measure the time from a point where a failure had occurred to the time where that failure evolves to the extent that it provokes the total failure, and thus the malfunctioning of that aircraft component or subsystem. Figure 2.1 presents a general graph of a fault evolution. As can be seen, the main goal is to predict the RUL, which is the time interval from the present time (t_p) to the end of life (EoL), defined by a specific *failure threshold*. This prediction can be done using different approaches summarized in Section 3.6.

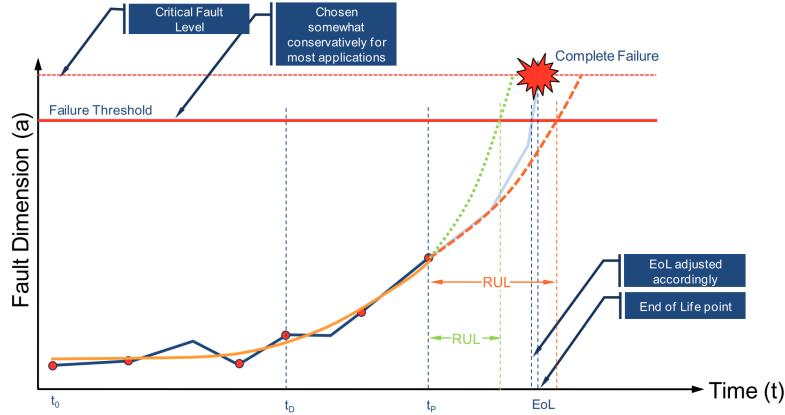


Figure 2.1: Fault behavior (from *Prognostics, The Science of Prediction* [19])

2.3 Prognostics and Health Management (PHM)

In order to make predictions about the components RUL, PHM methods should be considered. Prognostics and Health Management (PHM) systems are primarily focused on the estimation of a component's or subsystem's RUL through the evaluation of the system's current health state [20].

A PHM system is expected to detect possible faults in a subsystem, isolate them and predict their impact on the system through the monitoring of the fault growth. This way a PHM system can prevent a subsystem from total failure, increasing his reliability [20].

2.4 ATA Codes (ATA)

ATA stands for Air Transport Association and corresponds to a coding system designed with the objective of facilitating the categorization and identification of the different aircraft systems (for example, in the maintenance operations). The coding system is referred to as the ATA 100 System Codes.

These ATA codes compose a wide documentation that encloses different aspects of the aircraft flights. They identify the major aircraft systems like Air Conditioning, Electrical Power and Ice & Rain Protection, as well as other important specifications in the aircraft field like Dimensions & Areas, Periodic Informations and Comunications. These codes are accessible online [21].

For example, code 12-30 is used to refer Unscheduled Servicing included in the Servicing Routine Maintenance and code 32-40 indicates the Wheels & Brakes systems comprised in the Landing Gear of the aircraft.

2.5 Flight Cycle and Flight Phase

Another two important concepts for this work is the notion of flight cycle and flight phase.

Flight cycle corresponds to the time elapsed between the engine's start and the shutdown, thus encompasses one complete engine running [22]. Thus, generally, and with respect to commercial flights, the number of flight cycles of an airplane corresponds to the number of performed flights by that specific airplane.

The concept of flight cycles is used more in the context of preventive and scheduled maintenance as the criteria to define when the maintenance should be executed.

The another important concept is the flight phase. As the name suggests, a flight phase corresponds to a specific stage of a flight. Each phase division comprises specific values regarding the systems parameters, as ground speed, pressure altitude, temperature, etc [23]. Thus, specific data analysis regarding these metrics can be performed in order to monitor the aircraft performance in each phase, and also increase the security by identifying possible data outliers.

The phases division can be performed differently, though, the more common division, according to SKYbrary, is the following [24]:

- | | |
|------------------|----------------------|
| 1. Standing | 5. En route (Cruise) |
| 2. Taxi | 6. Maneuvering |
| 3. Takeoff | 7. Approach |
| 4. Initial Climb | 8. Landing |

Depending on the entities, different divisions and taxonomies regarding the flight phases can be performed.

2.6 Flight Deck Events (FDE)

The Flight Deck Events (FDE) are alerts that are automatically triggered by the system in order to warn the maintenance team about a specific occurrence.

These concepts are important and relevant in the aircraft context, as it allows the system, placed in the aircraft, to alert the maintenance team about some problem found, like a sensor value lower or higher than the standard values, the presence or absence of a particular substance or the failure in some required actions, like the opening/closing of a valve.

An FDE may have a priority flag associated in order to indicate its importance for the system's proper operation.

2.7 Removals

The removals correspond to the substitution of certain parts or components of the aircraft. This can be performed because of several reasons like scheduled maintenance routines, bad functioning of the aircraft parts or removal of parts for other aircraft.

Each removal doesn't mean that there was a problem associated, as it can result in a No Fault Found (NFF) situation.

This page is intentionally left blank.

Chapter 3

State of the art

The development of PHM methodologies integrated with the Condition-Based Maintenance (CBM) principles, encompass several steps, from the extraction and analysis of the sensors data, to the test and validation of the obtained methods. Henceforth, an important step in the idealization of the methodologies to develop is the study of the State of the Art regarding the different steps in the methodology pipeline.

First, in Section 3.1 the most common aircraft maintenance practices are presented, as well as, the discussion and comparison of the most used proceedings in the aircraft context.

The exploration and research of the State of the Art of the main development stages are performed from Section 3.2 onwards.

In Section 3.2 the importance of the feature selection in the algorithm's performance is discussed and presented. According to the State of the Art, there are 3 types of feature selection: Filter methods, Wrapper methods and Embedded methods.

In Section 3.3 the feature reduction relevance is explained and 3 different types of feature reductions are described.

Section 3.4, 3.5 and 3.6 present the techniques State of the Art of three core steps in the algorithms development: the HI extraction, the RUL computation and the PHM methodologies development. In these sections, multiple approaches, for each purpose, are described and discussed with regards to their relevance in this work, particularly with the work applied to the Air Bleed system, which corresponds to the main contribution.

Finally in Section 3.7, accuracy based metrics for measuring the algorithms performance are presented.

3.1 Current Aircraft Maintenance practices

Currently, aircraft maintenance has significant impact on the management and scheduling of airplanes and their respective flights, as they are dependable on the correct execution and planning of the maintenance routines to be performed on the aircraft.

There are multiple approaches to executing aircraft maintenance routines, the following are highlighted [25]:

- **Failure Based Maintenance (FBM):** This type of maintenance is performed when a failure occurs, there is no action in order to prevent or detect future failures. This method is risky and has high maintenance costs associated.
- **Preventive Maintenance (PM):** In this category, the maintenance interventions are performed in pre established points in time, which can be based on the usage of the aircraft component or in time intervals. Generally, the replacement schedule is performed in a conservative way in order to prevent common failures to happen [26]. Using this strategy, the aim is to reduce the number of failures without adding major economical costs to the company.
- **Condition-Based Maintenance (CBM):** This category is focused on the prediction of the failure. Historical data containing the components' failure behavior is obtained from components condition monitoring and is used to create a decision-based model, that can help diagnose and predict future failures. This way, unnecessary replacements are avoided and the aircraft components can run during their full lifetime [27].
- **Reliability Centered Maintenance (RCM):** This technique focuses on the most important functions of the system, that is, the most cost-effective functions. It cannot prevent all the failures, just the critical ones, according to the defined criteria. This criteria can be based on potential human injuries, environmental damage, production loss, etc [28].
- **Total Productive Maintenance (TPM):** This company-wide approach aims to improve the overall effectiveness of the equipment, and to achieve that, it defends that all company's departments should be involved, from the maintenance to the project engineering [29].

The maintenance routines executed by the airline companies are focused on optimizing three key points: profit, safety and planning [30]. In order to achieve the right balance between these three values, different actions and rules can be set by the companies.

Every airline needs to run an inspection program in order to guarantee the proper functioning of the aircraft and that program needs to be accepted by the certified authorities like FAA or EASA.

In literature, the most studied and used types of maintenance are the PM and the CBM. While PM proceeds to inspections and equipment replacement at regular intervals, CBM is focused on the monitoring of the aircraft condition degradation and predicting the remaining useful life of the aircraft components.

Respecting the efficiency, CBM might be a better option as it can diagnose and predict failures to happen, increasing the reliability of the aircraft. In terms of planning management, CBM adds a significant amount of uncertainty as it is condition-based, opposingly, the PM methodology makes the planning and scheduling more simple to execute. Regarding the financial aspect, the CBM approach “enforces” the retaining of the spare parts inventory, increasing the inventory value which is not a positive aspect, this also increase the lead time for the aircraft components. On the contrary, PM requires regular and predetermined planning which increases the inventory flexibility [6], although it can incur into high extra costs if a last minute part is needed for replacement of another.

Following the PM methodology, some companies have adopted the maintenance policies defined by the Federal Aviation Administration (FAA). There are 4 main types of checks to perform that can vary in scope, frequency and duration, namely [31]:

- **Type A check:** This is the most frequent check, and it states that inspections should be performed every 65 flight-hours. The inspection should include all major systems such as landing gear and engines.
- **Type B check:** The second check, the type B check, should be executed every 300 to 600 hours and should include more extensive visual inspection and also lubrication of all moving parts.
- **Type C check:** The type C check should take place once every year and it should last one or two weeks.
- **Type D check:** The last check has the longest interval, it varies from a three weeks to one month inspection and it should be done once every four years. These inspections should include stripping, painting and cabin refurbishment.

Overall, CBM approach has the most potential for improving aircraft maintenance, by reducing its costs and optimizing the components' useful life. Furthermore, the CBM concept is within the ReMAP fundamentals, thus, is followed and explored in this work.

With the objective of developing an accurate and trustworthy algorithm, capable of diagnosing a system condition (represented by the HI) and, consequently, predicting the future health condition (determined by the RUL), a set of important tasks should be taken into consideration, in order to assess their relevance in the specific context.

3.2 Feature Selection

One important step is the feature selection. Not all features have the same relevance and importance for the computation of the HI and, subsequently, of the RUL. Thus, the feature selection process aims to obtain a compact set of features that are more useful and relevant for the predictions, where the redundancy and correlation between features is the lowest [32]. This is achieved by eliminating noisy features that don't add any helpful information for the predictions and have not any correlation to the target classes. The feature selection also helps to reduce the complexity in the prediction, as the dimensionality (number of features) is lower.

According to the literature, the feature selection models can be divided in Filter methods, Wrapper methods or Embedded methods [33] [34].

3.2.1 Filter methods

The filter methods focus on selecting the most relevant and discriminative features. This is performed using a ranking approach.

Generally, the filter methods encompass two steps. The first step consists of ranking the features according to a specific criteria. The second step consists of selecting the features with the highest ranking. These methods are highly dependent in the choice of the ranking criteria. The chosen criteria should represent the feature relevance, which describes the features importance and value in discriminating the different classes [35]. Some of the most used criteria to rank the features are the correlation criteria, mutual information criteria, the reliefF criteria and hypothesis testing.

Correlation criteria: The correlation between features or between the features and the expected response can provide useful information regarding the features relevance. One widely used method for calculating the correlation between variables is the Pearson correlation coefficient [36], that calculates the linear correlation between two variables. It is calculated using the following formula:

$$r(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (3.1)$$

where x and y are the random variables, \bar{x} is the mean of vector x and \bar{y} is the mean of vector y .

Mutual information criteria: The mutual information is another useful criteria for determining the feature relevance. This criteria estimates the shared information between two variables [37]. The features with the most shared information with the target values are useful for discriminating the different classes, and thus, are the ones selected. The mutual information can be calculated using the probability distributions of the variables x and y , in the following form [37]:

$$\sum_x \sum_y P(x, y) \log \frac{P(x, y)}{p(x)p(y)} \quad (3.2)$$

where $P(\cdot)$ represent the probability distributions.

RelieF criteria: The reliefF criteria aims to determine the features that are more statistically relevant, considering the target values [38]. The reliefF technique computes the features capability of correctly distinguish random instances in the dataset. Assuming a binary classification scenario, the algorithm starts by, randomly, selecting an instance $x = x_1, x_2, \dots, x_n$ and then identifies the nearest same class instance (H) and the nearest opposite class instance (M). Based on the distance between the chosen instance and H and M , the features weights are updated. This process is repeated over a defined number of instances [39].

Hypothesis Testing: Statistical tests are another approach for selecting the most important features. By defining certain condition (hypothesis), a statistical method can be applied to a data sample, in order to conclude its veracity for an entire population. The result of the statistical method, generally represented by the *p-value*, should be interpreted in order to accept or reject the hypothesis conceived. There are many types of statistical methods with specific criteria for its usage. Some common methods are ANOVA, when the data samples follow a normal distribution [40], and Kruskal Wallis, when the data sample are not normally distributed [41].

3.2.2 Wrapper methods

The wrapper methods use learning models to evaluate subset of features. The feature selection algorithm wraps the induction algorithm, that is used as a black box, in order to find the optimal set of relevant features [42]. Figure 3.1 illustrates a general representation of a wrapper method.

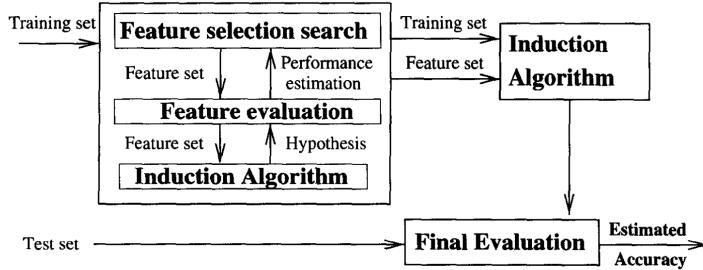


Figure 3.1: Wrapper method approach (from *Wrappers for feature subset selection* [42])

As it is shown, initially, the best set of features is found by evaluating different sets of features applied in the induction algorithm. Then, the set of features with the highest evaluation is returned and used as the final set, that will be run in the induction algorithm. The result is then evaluated (final evaluation) and the accuracy estimated. Here the goal is not to maximize the accuracy, but instead try to find the optimal set of relevant features.

Wrapper methods can be broadly divided in two categories the heuristics-based methods and sequential-based methods [43] [35].

Heuristics-based methods: The heuristics-based methods, focus on the application of certain heuristics in order to reach the optimal set of features in a quicker way.

One widely used method is the Genetic Algorithm (GA) [44]. In GA an initial population is created. This population contains multiple individuals called chromosomes, that represent the different candidates for the optimal solution and each individual is composed by a set of genes. In this particular case, each individual will correspond to a different set of features, each feature will be a gene. Then in each iteration of the algorithm, each individual is modified by specific operators in order to introduce variability in the population, those operators are mutations, that change (flips) the genes' state and crossover, that cross and fuse parts of the chromosome of different individuals. Each individual is evaluated according to a fitness function. In each iteration only part of the population passes to the following iteration, this selection of individuals is performed based on the fitness values [43].

Another used heuristic method is the Particle Swarm Optimization (PSO) [45]. The PSO is also an optimization problem that is inspired in social movement of flock of birds or school of fish. PSO representation, is similar to the GA, initially there are multiple particles that represent the different solution candidates and in each iteration these particles move together in order to improve and come closer to the optimal phase space.

Each particle has two properties, position and velocity. These are both updated in each iteration. In each iteration, the new velocity value of each particle is calculated based on the previous velocity and the position of the best particle of the previous iteration (the one with the highest fitness value). The new position of each particle is calculated based on current velocity of the particle and in its previous position [43].

The idea is that, hopefully, the particles move, as a group, towards an optimal phase space. In the particular case of the features selection, the goal is to improve the candidates (set of features) in order to reach the optimal phase space, that corresponds to the optimal features set.

Sequential-based methods: This category of methods work iteratively, initially a set of empty features is created, and then features are added incrementally according to a pre-defined objective function.

The Sequential Feature Selection (SFS) [46] is an example of this type of methods. The SFS method starts with an empty set of features and individual features are added in iterations. In each iteration, the feature that produce the maximum classification accuracy is permanently added to the set of features. The process stops when the defined number of features is reached. The Sequential Backward Selection (SBS) [46] idea is similar to the SFS, but in this method the initial set of features is composed of all existing features, and these are then removed, one by one, according to the maximum classification accuracy obtained in each iteration.

These two approaches, although simple to use, have some limitations, such as the fact that they don't take into consideration the relations and dependency between features.

Another sequential-based method for feature selection is the Sequential Floating Forward Selection (SFFS) [46]. This method is similar to the SFS method but it contains an additional step. In each iteration, after adding a new feature, one feature is removed. If the maximum classification accuracy drops, then the feature returns to the set, if the accuracy increases, then the feature remains removed and that specific iteration starts over again. Figure 3.2 shows the SFFS operation chart.

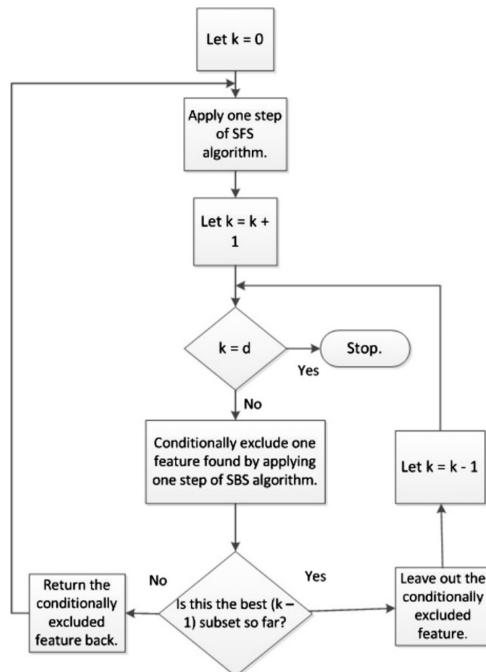


Figure 3.2: SFFS operation chart (from *A survey on feature selection methods* [35])

3.2.3 Embedded methods

The last category of feature selection techniques are the embedded methods [47].

These methods encompass the feature selection process in the training stage, this way, contrary to the other methods (filters and wrappers), there is an incorporation and interaction between the selection of the features and the learning process of the classifier, which may introduce valuable information not achieved by the other types of methods [48].

Some common embedded methods are decision trees methods and LASSO methods.

Decision tree methods: The decision tree based methods will be discussed in Section 3.6.2. Decision trees are generated based on chosen features, these are selected based on the feature discrimination capability.

Other more complex methods were created based on decision trees, as, for example, the Random Forest methods [49]. The Random Forest methods [50] correspond, essentially, to a multiple set of decision trees, that are then combined and fused in order to generate a more accurate and valuable prediction model. These methods are easy to use and can be applied both to regression and classification models, which make them useful in many different contexts.

LASSO methods: LASSO stands for Least Absolute Shrinkage and Selection Operator [51] and consists in a linear regression method that seeks to improve the model prediction accuracy by performing two different tasks: the shrinking (regularization) process and the feature selection.

First, the shrinking process is used in order to penalize the regression coefficients by shrinking some to the value of zero, then in the feature selection process, the variables with non-zero coefficient values are chosen and used in the prediction model [52].

This method is significantly dependent on the penalty score. The penalization impact in the regression coefficients is defined by the λ parameter. The λ should be chosen wisely, if it is too low (closer to zero) the penalization won't take effect and the regression turns into a Ordinary Least Regression (OLS), this will decrease the bias but increase the variability in the regression. On the other side, if the λ value is too high the regression coefficients will quickly shrink to zero, which will reduce the problem dimensionality. The higher λ will also reduce the variance, but increase the bias in the prediction model. Thus, a good λ value should be chosen, taking in consideration the compromise between bias and variance.

Using this process, it is expected that the prediction model accuracy increases, as the prediction error decreases.

The feature selection is an important step in the development of machine learning algorithms, specially, if the dataset contains multiple features. As explored in this Section, there are multiple approaches for the correct choice of features to use and they can depend on the type and characteristics of the available data.

In this work, specifically in the Air Bleed dataset, due to the short number of features present in the data and the characteristics of the dataset, a simple filter method is applied. In particular, a correlation criteria may be suitable for the available dataset. As the dataset contains data from different systems, extracted from similar sensors, a correlation criteria should be valuable in order to identify which features represent the same sensor in the different systems. Besides that, some possible redundancy is expected to be eliminated.

3.3 Feature Reduction

Even after applying features selection techniques, the dimensionality, which is given by the number of features to use, might still be high. This high dimensionality is generally avoided, as it increases complexity in the prediction models, making them less accurate and, from the user point of view, it is more challenging to understand and conceptualize the features values in a higher dimensional space [53].

Thus, in order to reduce the dimensionality, feature reduction techniques can be applied to the data. There are many different feature reduction methods [54] [55], but the most used techniques are the Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Self Organizing Map (SOM).

3.3.1 PCA technique

PCA is one of the most known techniques for feature reduction. It performs a multivariate statistic analysis in order to extract useful information from the data that is expressed and represented by orthogonal variables called principal components [56].

These are found by analyzing the data covariance matrix and identifying the directions where the data variance is higher. The directions are computed using the eigenvectors of the covariance matrix and the ones with a higher eigenvalue indicates that the data variance considering that direction is higher. Thus the directions (eigenvectors) with bigger eigenvalues should be selected. There are different criterias for choosing the correct number of eigenvectors to select. The Kaiser criteria and Scree criteria are often used criteria.

Figure 3.3 illustrates a two dimensional scenario, where x_1 and x_2 represent two different features. Furthermore, the red and blue lines represent two eigenvectors, which represent the directions where the data variance is bigger.

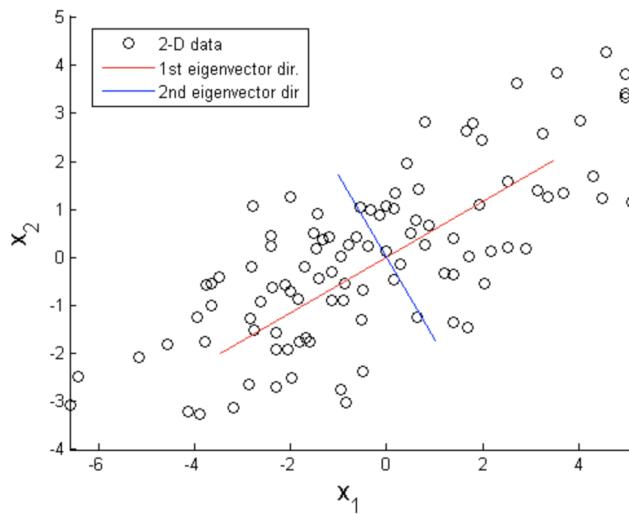


Figure 3.3: Principal Components (from *Basics and Examples of Principal Component Analysis (PCA)* [57]

After identifying the best principal components (directions that lead to bigger data variance), the data is projected according to these components and thus the number of features is reduced, which will lead to the dimensionality reduction.

3.3.2 LDA technique

LDA is another well known feature reduction technique that aims to project high dimensional data into low dimensional data [58]. This technique is similar to a regression model as it computes the new features from linear combinations of the original features.

The overall idea is to project the data in directions that minimize the distance between same class instances and maximize the distance between the means of different classes. This way, the class discrimination will be maximized.

As can be depicted, this technique is supervised, which means that the data must be labeled.

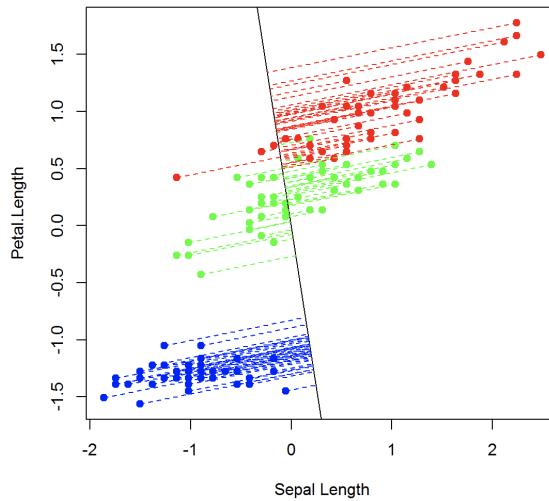


Figure 3.4: Linear Discriminant (from *Discriminant Analysis* [59])

Figure 3.4 represents an example of the application of the LDA. Here, an optimal direction for class discrimination is illustrated. Using the indicated direction, the separability between classes is maximized and the distance between same class instances is minimized.

3.3.3 SOM technique

The last technique for feature reduction discussed is the Self Organizing Map (SOM). This technique reduces the dimensionality to one or two dimensions by training and applying an Artificial Neural Network (ANN). The training method will not be based on the accuracy or prediction error, as this is an unsupervised method. Instead competitive learning will be used, this means, that the neurons will be competing against themselves [60].

The idea is to, using the Artificial Neural Network (ANN), establish a mapping between the dataset space X and the feature space W . This process is established, applying the following steps [61] [62]:

1. Initialize each node's weights w_j .
2. Choice of a random training input vector x from the input space.

3. Identification of which node's weight is more similar to the input vector. This identified node is considered the Best Matching Unit (BMU).
4. Update of the BMU weight, as well as of his neighbours (other nodes). The BMU node will be displaced in order to be more similar to the considered input vector, the neighbors will also be more similar to the chosen input vector depending on their distance to the BMU, bigger distance produces smaller displacement.
5. This process is repeated for a fixed number of iterations.

Figure 3.5 corresponds to an illustration of the mapping of the input space into the feature space. A BMU node and its respective neighbors are also represented.

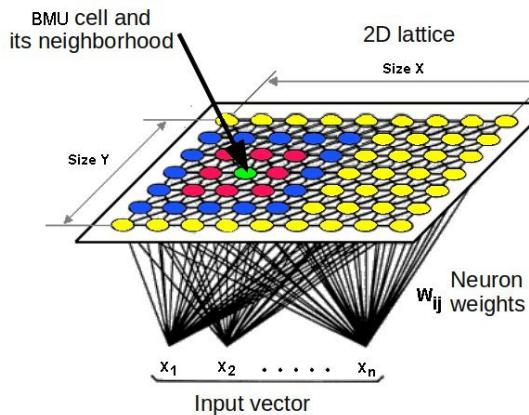


Figure 3.5: Self-Organizing Mapping (from *Using Complex Network Topologies and Self-Organizing Maps for Time Series Prediction* [63])

For this work, the use of feature reduction shouldn't add significant value to the algorithms performance, due to the fact that the dimensionality of the datasets is small, thus it does not add significant complexity to the prediction.

Nevertheless, for other aircraft systems comprised in the ReMAP proposal (for example the Cabin Air Conditioning and Temperature Control System), where the dimensionality is significantly higher, the application of feature reduction techniques should be valuable for reducing the models complexity and thus obtaining better results.

3.4 Health Indicator (HI) extraction

The computation of the Health Indicator (HI), in this case, of a specific aircraft component or subsystem, plays an important role in the algorithms that aim at accurately computing the RUL. The HI value represents the health condition of a specific aircraft component [64] [65] and is calculated based on the different data or signals retrieved from the aircraft sensors related to that component, like vibration signals, Acoustic Emission (AE) signals, ultrasound signals and temperature measurements, amongst others [66] [67]. The HI is used to determine the component condition, in order to identify the possible failures or degradation inherent to that component. There are two different categories of HI: Physical Health Indicator (PHI) and Virtual Health Indicator (VHI) [16] [68].

3.4.1 Physical Health Indicator (PHI)

The PHI uses physical signals, like vibration signals and acoustic emission signals, retrieved from the sensors, as direct metric for the construction of the HI value. In order for the PHI to correctly reflect the condition of a certain component, a broad understanding of the impact and influence of the physics in the component failure or degradation state is required. Furthermore, it is essential that the signals being retrieved from sensors contain relevant information regarding the component condition. This information is normally extracted using traditional statistical methods or signal processing methods [16]. These approaches analyse and study the different features from the signals like, Root Mean Square (RMS) [18], peak [69], kurtosis [70], amongst others [71].

3.4.2 Virtual Health Indicator (VHI)

The other type of HI are the VHI. These are generally used in situations where the physical signals don't effectively represent the components' condition. The VHI is computed by fusing and combining multiple PHI or different signals retrieved from sensors [16]. The result of this fusion is a one dimensional VHI value which represents the global state or condition of the component and doesn't have a physical meaning associated to it. The VHI can be extracted by applying different preprocessing methods on the available dataset, normally composed of different type of signals that represent all the system evolutions, i.e., from the healthy stage to the faulty stage. Common VHI extraction techniques include linear data regression [17], Mahalanobis distance [72], SOM [73] and PCA [74] [75].

Comparing the suitability of the PHI and VHI approaches for the work applied over the Bleed dataset, the VHI is considered to be more appropriate. The PHI approach is highly dependent of the type of the acquired signals and the physical knowledge about the system's normal operation. Hence, as no expert knowledge regarding aircraft physics is present, this is not the most advisable approach to be followed in this work. Opposingly, the VHI approach is more convenient for this work. As the aircraft systems are characterized by different types of signals and sensors, the combination of all this information will improve the algorithms accuracy, as it is expected to represent more accurately the system condition. Thus a VHI approach should be more useful. Moreover, in terms of explainability, the usage of VHI is better, as the understanding of the meaning of the HI values is clearer.

3.4.3 HI Computation

As mentioned before, the computation of HI is an essential step in the RUL computation, as it represents the component health condition.

In particular cases, the HI can correspond to relevant features extracted from the data that represent the system/component degradation, like the RMS or kurtosis. In other cases, the HI can be computed by combining different features or characteristics of the data in order to calculate a new system feature or value representing the HI.

Furthermore, in particular cases, it can be more challenging to extract the components condition from the data due to noise, poor generalization capability, lack of relevance or absence of sensors data [64]. Therefore, the methods used for HI extraction should be adequate for each type of component and dependent on the available sensor data retrieved from that component. Taking this into consideration, a study for common methods in the literature regarding the HI computation was performed.

The research was performed considering two different case scenarios. In the first scenario, it is assumed that the data is known, that is, all the variables and features present in the dataset are clearly identified. In the second case scenario, it is considered that the dataset available contains anonymized data, thus there is no information regarding the characteristics of the dataset features.

First Case Scenario

In the first scenario, three different domains were considered [76] [77]: Time domain, Frequency domain and Time-Frequency domain, where methods for HI computation were investigated for each domain. Also a brief analysis of the signals that suit the identified methods was performed.

Time domain: Time domain based methods are the simplest approaches for detecting the degradation of a specific system or component. From a general perspective, these methods focus on the analysis of the signal morphology in the time domain or in the extraction and study of some statistical parameters of the signals. The most common studied parameters are RMS, kurtosis, crest factor and peak [71] [78].

The parameter *peak* corresponds to the maximum value of the signal interval. It can be represented in the following formula [71]:

$$x_{peak} = \max(x_1, x_2, x_3, \dots, x_n) \quad (3.3)$$

where n is the length of the time series and x is the time series of the considered signal.

The parameter *RMS* corresponds to the square root of the mean square, and is represented by the following formula [71]:

$$x_{rms} = \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \quad (3.4)$$

where n is the length of the time series and x is the time series of the considered signal.

The parameter *kurtosis* indicates the flatness of the waveform under analysis. If the kurtosis values is lower than 3, it suggests that the signal has a flat peak, if the kurtosis values is higher than 3, it means that the peak will have higher amplitude. This parameter is calculated using the following formula [71]:

$$\text{kurtosis} = n \cdot \frac{\sum_{i=1}^n (x_i - \bar{x})^4}{(\sum_{i=1}^n (x_i - \bar{x})^2)^2} \quad (3.5)$$

where, n is the length of the time series, x is the time series of the considered signal.

The parameter *crest factor* is the ratio between the peak and the RMS of a signal waveform. It is calculated using the following formula [71]:

$$cf = \frac{x_{peak}}{x_{rms}} \quad (3.6)$$

where x_{peak} corresponds to the peak value and x_{rms} corresponds to the RMS value.

The vibration signals, are the most used signals in the determination of the degradation in systems like bearings, batteries, gears, etc.

Z. Zang et al. [70] applied a band-pass filter in the kurtosis parameter, in order to estimate the degradation path of a dataset containing bearing data. A. Malhi et al. [69] extracted and used the peak and RMS values from the obtained wavelet coefficients. The peak value was used to predict long term effects of the defects propagation and both peak and RMS parameters were used for competitive learning. N. Li et al. [18] also used a time domain approach, where the RMS and kurtosis parameters were extracted from the available vibration signals and used for the degradation assessment. In particular, the kurtosis was used for health monitoring and the RMS for the RUL prediction, as its values correspond to the degradation index.

J. Sun et al. [79] followed a different approach to predict the HI of an Air Conditioner system. Instead of extracting and analysing statistical parameters from the signals, they decide to use a signal processing method. In this approach they used a Multivariate State Estimation Technique (MSET) for computing the HI. The MSET, first, uses the historical data to describe relationships between multiple variables (features), based on the sensors data, and assign weights to each feature. In the end, the HI is calculated through the weighted average of historical training sample values.

Although vibration signals are the most used in degradation assessment of components or subsystems, these statistical techniques can also be applied in other types of signals like AE [80] and ultrasound [81].

Frequency domain: Methods applied in the frequency domain of the signals are also considered appropriate for degradation or fault detection. These methods focus on the isolation and study of the frequency spectrum of the signals, which is assumed to be regular over time.

The most common methods applied in the fault monitoring are **Fast Fourier Transform (FFT)** and **Envelope analysis**.

The **FFT** algorithm computes the Discrete Fourier Transform (DFT) using an optimized and efficient approach. In general terms, the FFT can be calculated using the following formula [82]:

$$FFT(k) = \sum_{c=1}^n x(j)\omega_n^{(c-1)(k-1)}, \quad (3.7)$$

$$\omega_n = e^{(-2\pi i)/n} \quad (3.8)$$

where, n is the number of samples for a certain signal, $x(j)$ is the input signal and ω_n is the n th root.

Z. Zhang et al. [82] proposed a technique for predicting the degradation of components in machines, where a Neural Network (NN) was used for computing the RUL of specific components. The NN received as input the peak values selected from the FFT series obtained from the decomposed signals and calculated the RUL values. L. Liau [83] used FFT for decomposing the original signal in frequency bands, the energy of each frequency band, which is the sum of squares of the amplitudes, was used as features for the calculation of HI. The HI was calculated through the application of a genetic programming method that identified the combination of features that better described the fault progression. B. Wu et al. [84] proposed a novel degradation indicator, the Sensitive Frequency Band (SFB) power value. The SFB corresponds to the frequency bands where the FFT spectrum is considerably different. These were identified by calculating and analysing the accumulated amplitude difference between the different frequency bands.

The SFB power value was then used as input for a Radial Basis Function Network (RBFN) neural network for computing the respective RUL.

Another widely used technique is the **Envelope analysis** [85]. This technique is used for detecting faults on certain systems by analysing the vibration signal frequency components. The Envelope analysis includes two steps [77]: The first corresponds to the application of a band-pass filter in order to isolate and extract the frequencies of the signal that corresponds to the vibration phases of the system being analysed. The second step consists of the cleaning of the signal by removing the resonance in order to detect the defect impact frequency.

P. Boskoski et al [86] proposed the combination of the envelope statistical complexity of vibration signals and the Gaussian process models in order to compute the RUL of bearings. M. Gasperin et al. [87] used this technique to estimate the RUL of operating gears. The envelope analysis was performed in order to extract the power density of the gear-mesh frequencies identified in the envelope spectrum. These were then used in a stochastic model for predicting the gear wear.

These frequency domain techniques may also be suitable for AE signals [88], however they are not well fitted to ultrasound signals as these normally contain reflections, which require the use of techniques that consider the variable time in the spectral analysis of the signal [89].

A drawback of the frequency domain techniques is that it can only be applied to stationary signals, which means that the frequency spectrum of the signal is constant. However, this assumption is not true in most of the real case scenarios. For the scenarios where the available signals are non stationary the time-frequency techniques are a good solution.

Time-Frequency domain: Time-frequency techniques consist on the extraction of both time and frequency information from the available signals. Furthermore, they are a powerful tool for the analysis of non stationary signals as variations in the time domain can be identified in the frequency spectrum [77].

The two most common methods in the time-frequency domain are the **Short Time Fourier Transform (STFT)** and the **Wavelet Transform (WT)**. These types of methods are commonly applied to non stationary signals where the frequency spectrum is not constant [90].

The **STFT** techniques consist on the application of a traditional FFT using a fixed sliding window approach. The results (squared magnitude of the STFT coefficients calculated using equation 3.10) of the STFT computation are then presented and analysed in a spectrogram. The accuracy of this technique is dependent in choosing the appropriate window size, in order to correctly analyse the frequency components in each interval [77]. The STFT can be expressed by the following formula [91]:

$$STFT(f, t) = \int_{t-T/2}^{t+T/2} \omega(t-\tau)x(\tau)e^{j2\pi f\tau} d\tau \quad (3.9)$$

where $\omega(t-\tau)$ is the sliding window, $x(\tau)$ is the signal, T is the window size, t is the time, f is the frequency and $e^{j2\pi f\tau}$ is the complex exponential [91].

The energy density spectrum of the signal is calculated using the square magnitude of the STFT:

$$S(t, f) = |STFT(t, f)|^2 \quad (3.10)$$

Y. Zhang et al. [92] used a hybrid feature extraction technique which consisted of the combination of STFT and Non-Negative Matrix Factorization (NMF) in order to extract features from the vibration signals under analysis. Next, the health indicator was calculated by quantifying the similarities between features using a SOM network. P. Harihara et al. [93] proposed a method to calculate the fault indicator in pump bearings which combined the computation of STFT and the RMS. The fault indicator (Fi) was calculated using the flowing expression:

$$Fi = \frac{1}{3} \sum_{a,b,c} \frac{\sum_k l_k^2}{l_f^2} \quad (3.11)$$

where a, b, c correspond the the three phases of the motor, l_k is the value of the RMS of the k-th harmonic component and l_f is the fundamental frequency component [93].

The **Wavelet Transform (WT)** is another well known method for detecting failures or degradation through the analysis of non stationary signals. It mathematically represents the signal as a set of functions with different frequency domains, that are called wavelets [94]. Similarly with STFT, the WT also uses the sliding window approach, but instead of having a fixed window size, the window size of the WT is dynamic, as it adapts to the frequency resolution being analysed.

A significant work in the health monitoring context has been developed using WT. X. Jin et al. [95] proposed a new technique to calculate the HI that represents the degradation in bearings. The indicator is calculated using a Autoregressive (AR) for extracting noise from the vibration signal, then WT is applied to extract the features that better describe the bearing degradation and finally a Mahalanobis distance is calculated to compare the correlation between features, taking into consideration the healthy bearings [95]. The HI calculation is illustrated in the Figure 3.6

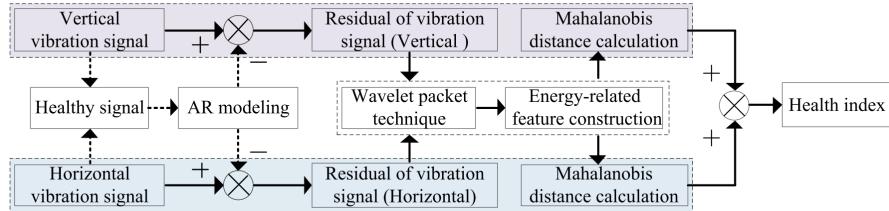


Figure 3.6: HI calculation (from *Anomaly Detection and Fault Prognosis for Bearings*

W. He et al. [96] also presented a technique for estimating the degradation index in fan bearings using wavelets. In this approach, the wavelet is used for extracting the bearings characteristics from the vibration signals and then the index is calculated by summing up the amplitudes of the identified characteristics to evaluate the degradation of the bearings.

These type of methods were already applied in different contexts and using different types of signals like AE [97] and ultrasounds [89]. The STFT and WT techniques are the two most common techniques in this domain (time-frequency), nevertheless there are other possible approaches like Wigner-Ville Distribution [98] , Hilbert–Huang Transform [99] and Empirical Mode Decomposition (EMD) [100] that have not been analysed in detail.

As can be concluded, different methods for computing the HI value can be used, depending on the data domain and the type of data. The type of signals most used for research in the literature for health diagnostic and prediction is the vibration signals due to the capability of monitoring several types of systems like gears, bearings, engines and shafts.

In particular, bearings are widely used due to their comprehensiveness of application and importance in the operation of rotating systems [77].

Overall, in order to correctly apply this type of methods, a good understanding of the system and its variables is important. This way, the most important and relevant signals/sensors regarding the system health condition can be identified and analyzed using approaches from different domains.

Second case scenario

For the second scenario, because the available data is anonymized, it is more challenging to extract relevant information in order to compute the HI of the system being analysed. Therefore, feature extraction and the methodology for computing the HI has a significant relevance for the accuracy of the developed methods. As the source of the obtained data is unknown, the VHI is more suitable to this scenario than PHI.

According to the literature there are possible sets of approaches that can be adapted for this specific scenario.

Y. Wang et al. [101] used the Mahalanobis distance of statistic features as health indicator for representing the degradation of the system. The considered features include the max, min, mean, RMS and kurtosis of the given signal. The HI was calculated using the following Mahalanobis distance formula [101]:

$$MD_i = \frac{1}{n} z_i C^{-1} z_i^T \quad (3.12)$$

where the vector z was composed by the considered features and C was the correlation matrix.

When the HI value exceed a certain threshold ($\mu + 3\sigma$) it was assumed that the degradation phase of the system was starting, then the RUL was calculated using a Winer Process based degradation model. This techniques was tested on bearings.

Another way to compute the HI of anonymized data was presented by T. Wang et al. [17]. This approach requires that sensors data regarding the entire life cycle of the component or subsystem under analysis is provided, this means, from the moment where the component is healthy and without any type of failure or degradation to the moment where the component or subsystem is malfunctioning. In this approach the authors use a multivariate linear regression model for representing the HI based on a set of selected features. The regression model is presented in equation 3.13.

$$y = a + \sum_{i=1}^N \beta_i x_i + \varepsilon \quad (3.13)$$

where y represents the HI, $x = (x_1, x_2, \dots, x_N)$ is the feature vector with dimension N , $(a, \beta_1, \beta_2, \dots, \beta_N)$ are the model parameters that will be estimated and ε refers to the noise term.

For the parameters estimation a sample of the dataset is used. This sample contains a percentage (ex: 5%) of the initial sensors data of the available life cycles where it is assumed that the HI is 1 and a percentage of the final sensors data of the same life cycles where it is assumed that the HI is 0. Having this sample of dataset with information regarding the y (HIs) and the x vector (sensors data) it is possible to estimate the α and β s parameters.

Having the estimated parameters it is possible to compute the HI for any set of the considered sensors. This particular approach was the winner of the PHM08 Challenge Data Set [102].

Also, L. Gou et al. [103] proposed a novel technique for calculating the HI using a Convolutional Neural Networks (CNN). A deep learning approach was used due to its capacity to capture and extract useful data from the given input. The CNN architecture is illustrated in Figure 3.7. As can be observed, the CNN receives as input the raw signals available and the HI is computed by using two convolutional layers, 2 pooling layers and 2 fully connected layers. The system will be considered faulty when the HI value exceeds a defined threshold, in this case the defined failures threshold was $\mu + 3\sigma$.

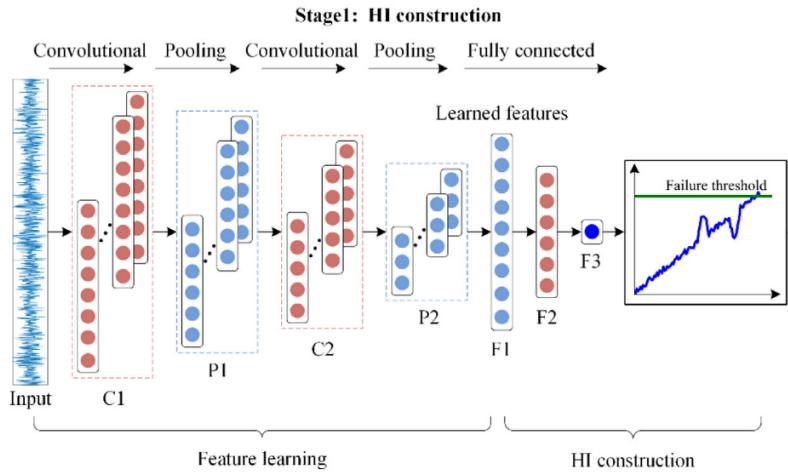


Figure 3.7: HI calculation (from *Machinery health indicator construction based on convolutional neural networks considering trend burr*

Comparing the two scenarios' appropriateness to this work, particularly in the Air Bleed system, scenario two is considered more well suited. The fact that the aircraft data, provided by ReMAP for this work, is anonymized enhances the parallelism between this work and the second case scenario.

Nevertheless, despite the similarity of this work context with the context of the presented approaches applied to the second scenario, none of these were considered valuable to be applied to this work. The main reason that supports this conclusion is the fact that the data used in the described approaches is not related with the type or characteristics of data retrieved from aircraft sensors, that is used for this work.

Depending on the type of signals and data acquired, the expression and manifestation of possible signs of degradation is different and thus the method for identifying the systems health condition should be different.

Furthermore, as seen in this Chapter, little work was found that used aircraft sensors data for developing algorithms capable of predicting the system health condition and consequently estimate the RUL. Moreover, significant part of the work found related to the aeronautics field, was using a public dataset, the Turbofan dataset available in the NASA Prognostics Data Repository [104], which was used in the first part of this work, as an experimental case study. Nevertheless, this dataset was generated synthetically and oversimplifies a real case scenario of aircraft degradation.

3.5 RUL computation

In order to predict the system future degradation behavior, that is, the forecasting of the HI values, the RUL value is helpful. The RUL value, expresses the number of cycles remaining of useful life regarding a particular component or subsystem.

Over the years, research in the prognostics field has increased considerably. A significant percentage of these publications concern the development of technical and theoretical models for the prediction of the evolution of the system condition, as well as, failure behavior, with the objective of predicting the RUL of specific components of the aircraft.

Due to the complexity embedded in aircraft systems, the research regarding the RUL estimation has been focused on specific aircraft components or subsystems [105], such as engines, power system and structural components.

3.5.1 Engines

The engines play an important role in the operation of the aircraft therefore the maintenance of the engines, like the gas turbofan engine, is important in preventing possible anomalies or failures to occur. The prognostic of failures can be performed using different techniques [106]. Regarding to data driven methods, Feng Xue et al. [107] developed a fuzzy instance-based approach for predicting the RUL of aircraft engines. Using local fuzzy models, a set of similar instances with identical operational characteristics to a given engine is defined. The RUL for the given engine is calculated through a fuzzy aggregation of all the instances RUL. Y.G. Li and P. Nilkitsaranont [108] used a combination of linear and quadratic regression models in order to better predict the degradation behaviour and, consequently, predict the RUL of turbofan gas engines. The developed methods were applied in a model industrial gas turbine similar to a Rolls-Royce industrial AVON. A. Vatani et al. [109] implemented two different approaches regarding the prediction of the RUL of gas turbine engines, based on Neural Networks. In the first approach a Recurrent Neural Network (RNN) architecture was used in order to learn the degradation trends and dynamics presented in the data and then predict the degradation growth for a point in the future. In the second approach, which obtained better results, a nonlinear auto regressive neural network (NARNN) was used for the same purpose. The ultimate objective was, using the predictions regarding the degradation growth over time, to calculate the RUL of the mentioned engines.

Physical models are also used for RUL prediction methods, specifically, the bearing spalls are a wide subject for prognosis [110]. S. Marble and B. Morton [111] conducted an experimental study of bearing spall progression by implementing a physics-based model for predicting the spall progression over time. The spall propagation modelling includes the determination of the dynamic loads and stresses occurring during the spall and the development of algorithms for the calculation of the spall propagation rate based on the diagnosed stress, in order to predict the spall damage in the future. N. Bolander et al. [112] implemented a physics-based model for predicting the RUL of engine bearing. The RUL prediction method was based on a particle filter approach with Bayesian updating, where first the model was initialized with expert knowledge, and then improved by optimizing the particle trajectories weights using incoming and past diagnostic data from specific sensors.

3.5.2 Power systems

Power systems are other type of aircraft components prone to failures, on which research has been made [113]. T. Batzel and D. Swanson [114] developed a physics-based method for estimating the RUL of aircraft power generators using a Kalman Filter. By applying a kinematic state model which tracked the position, velocity and acceleration of the observable features, valuable information was generated and used for the RUL prediction.

An interesting power system component that is being widely used in aircraft due to its good performance and long life is the lithium-ion batteries, which have also been subject to research. J. Liu et al. [115] implemented an adaptive recurrent neural network (ARNN) in order to predict the RUL of lithium-ion batteries. The neural network was trained using a recursive Levenberg-Marquardt (RLM) in order to optimize the network weights. D. Liu et al. [116] used, instead, a Nonlinear Degradation AutoRegression (ND-AR) for predicting the RUL of the lithium-ion batteries. Initially a simple Autoregressive (AR) model was created for the prediction of the degradation of the battery capacity and then, in order to make the degradation trend more precise, a nonlinear degradation process, which consisted on accelerating the degradation factor while the cycle was increasing, was applied to the model.

3.5.3 Structural Components

As for the prognostics of failures in structural components of aircraft, different sub components have been studied. One example is the wearing associated with the brakes , these can be detected and predicted using different methodologies. The most simple technique includes the definition of a standard landing wear rate, based on previous experience, and using that given rate, calculate the time or cycle where the brakes wearing overpasses a specified threshold [117]. Another simple method, can be the use of a simple extrapolation of the brake wear percentage, based on real data from historical events. With the objective of improving the accuracy of the brake wear prediction, S. Ferreiro et al. elaborated two approaches based on Bayesian network models for the prediction of the brake wearing in the plane over time [118]. The first approach, *PhysicalBN*, focused on the analysis of the parameters with major influence on the degradation of the equipment, these are: aircraft weight, landing velocity and brake operation during landing. The parameters probabilities were calculated through the statistical analysis of the data. The second approach, *OpBN*, complements the first approach as it takes into account extra operational parameters, like flight distance and the weather, that allows the configuration of causal relationships with the *PhysicalBN* model input and thus a better estimation of the model input parameters, with the objective of better predicting the final RUL.

Other studied component, prone to failures and cracks, are the the fuselage panels. A. Coppe et al. [119] developed an approach for estimating the RUL of panels with cracks, this approach consisted in the combination of a least-square method for filtering data, with a Bayesian updating method for improving the RUL calculation. The crack growth was estimated using the a simple damage growth model, the Paris law, with two parameters. Y. Wang et al. [120] considered another approach for the crack propagation in fuselage panels. A model-based prognostics was implemented, in which, a Extended Kalman Filter (EKF) was used in combination with a linearization method. This method was divided into two steps. In the first step, the EKF was used for the estimation of the unknown model parameters and the current damage state. In the second step, a linearization method was applied for computing the distribution of the damage evolution in the future.

Besides the part or component of the aircraft that is being analysed, the type of data is also important in the choice of the best approach for the RUL computation.

Depending on the data available, the degradation or failure presented in the aircraft components should be different, and thus its behavior is different. This fact should be taken in consideration when choosing the approach to follow for the prediction of the component's future behavior.

3.6 Prognostics and Health Management

As mentioned before, a PHM system can have a great impact in aircraft maintenance. Using a PHM system it is possible to monitor and accurately estimate the components' RUL, this would bring numerous advantages [19]: 1) Decrease damage caused by the failure in that particular subsystem as well as in the surrounding healthy subsystems; 2) Help create a more efficient maintenance plan thereby reducing the logistics costs; 3) Performing aircraft service (equipment replacement) only when needed; 4) Increase reliability and confidence in the system, helping increase airline reputation.

PHM methods are broadly divided into three categories [121], [122] : Model-based methods, Data-driven methods and a Hybrid approach of the two methods. All methods goal is to accurately predict the system RUL, using different approaches, for a given time frame in the future, by estimating the system's health state.

3.6.1 Model-based methods

The Model-based methods use physics and mathematical models to represent the system behaviour and consequently, make the predictions regarding the system future state. Through these models there is an incorporation of physical understanding, like physical laws, requiring human knowledge in order to estimate the system health status and ultimately predict the RUL.

As can be seen in Figure 3.8, the constructed mathematical models are used to assess similarities and correlations with data retrieved from sensors of real systems, and generate residuals that will be evaluated and used as features in the diagnostics and prognostics step. The accuracy of these models is significantly dependent on the quality of the developed mathematical models.

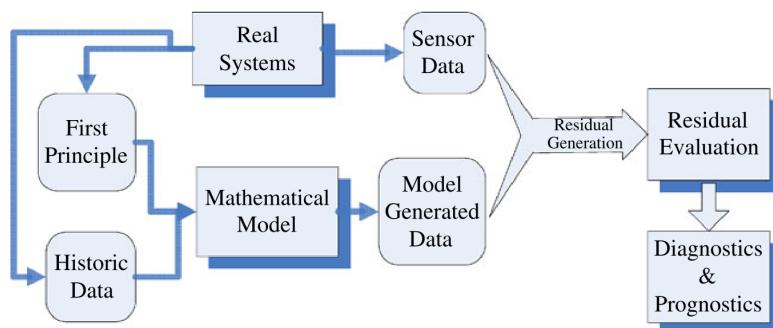


Figure 3.8: Model-based flowchart applied in CBM context
(from *Current status of machine prognostics in condition-based maintenance: a review* [123])

The advantages of these type of methods are the fact that estimations are less dependent on historical data and once the model is defined, it can be used in different cases by making small adaptations. A relevant disadvantage is the difficulty of implementing the models due to the need of expert knowledge [122].

Furthermore, this type of methods has been widely used in the health monitoring context. Y. Li et al, presented a model-based approach for predicting the remaining life in bearings. In this approach, fine parameter tuning was performed in the propagation model by comparing the measured and estimated defect sizes in the bearings [124]. Y. Wang et al. [120] used a model-based approach for estimating and predicting the crack propagation in fuselage panels. In this approach Extend Kalman Filter was applied to the model parameters estimation and current damage state. Then a linearization technique was used to predict the evolution of future damage. Furthermore, fuselage cracks propagation are a common research field in the model-based context, as the crack evolution can be accurately defined and simulated by mathematical models, as the Paris law [125] [126].

3.6.2 Data-driven methods

The Data-driven methods create the predictions of RUL using models learned from the data, normally represented in time series. These models tend to represent the degradation behavior of the system based on historical data. The advantages of this approach are simplicity and the speed in the implementation. However, the need of large amounts of data and the generalization with respect to the components' degradation behavior are some of the drawbacks. The Data-driven methods can be mainly grouped on the following categories:

Regression-Based

Regression-based models are considered one of the simplest predictive methods. Historical data, in the form of a time series, is used to fit a regression model that can accurately explain the system behavior.

The most used method for fitting the regression models is the Least Square Method. This method chooses the parameters that minimize the sum of the squares of the differences, between the data points and the predicted line.

The model can then be extrapolated to determine some points in the future in order to make predictions. The general formulation of a regression model is the following [127]:

$$x = \beta_0 + \sum_{p=1}^P \beta_p \phi_p(t) + \varepsilon \quad (3.14)$$

where β represents the model parameters, $\phi_p(t)$ is a representative function of t (it can be either linear or non linear) and ε is the noise term.

The regression-based methods can follow different modeling techniques, some common approaches are linear regression [17], logistic regression [128] and regression with nonlinear behaviour [127]. The main difference between these methods is the range and behaviour of the dependent variable. In a linear regression the dependent variable is continuous and does not have a fixed interval of values. In the logistic regression there is a fixed interval for the values of the dependent variable, these are between 0 and 1, which may make this method suitable when working with probabilities.

Using these types of regression models, it is possible to simulate the degradation behavior of a component or a system, in order to predict future failures or anomalies by identifying similar pattern behaviours between individuals in the same environment. This results in a poor capability of generalization of degradation models [129].

Nevertheless in particular cases, the system behaviour might be complex to represent and predict due to external factors (like fault conditions or combination of different degradation progressions), and so a linear or logistic approach might not be enough. In these cases, a nonlinear approach might be best to predict the system evolution over time. These nonlinearities might result from a combination of multiple linear or logistic regressions or the application of more complex techniques like NN [127].

Filter-Based

Filter-based approaches assume that a desired state of a system can't be measured directly but only through other measurable variables. One typical example is the Kalman Filter.

The Kalman Filter predicts the hidden states for the next time step given the history of estimated states and observing noisy outputs [130]. The way to make and improve the future state predictions is by minimizing the error between the state estimation and the actual measurement [131].

As explained by Welch G. and Bishop G. [132], a discrete Kalman Filter can be represented by a cycle (Figure 3.9) between two different processes. One is the *Time Update* where the unknown state is estimated in a future time, based on a model defined by mathematical equations. The other process, *Measurement Update* will then provide feedback to the estimation made, adjusting the estimation based on an actual measurement.

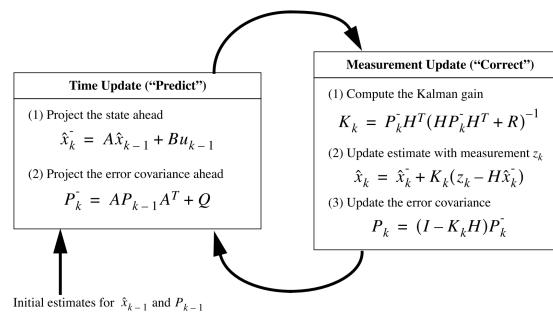


Figure 3.9: Discrete Kalman Filter cycle (from *An Introduction to the Kalman Filter* [132])

As illustrated in Figure 3.9, in the *Time Update* step, the update is performed using the following equations:

1. Projection of the state ahead, \hat{x}_k^- :

$$\hat{x}_k^- = A\hat{x}_{k-1}^- + B\mu_{k-1} \quad (3.15)$$

In this equation, the *a priori* state \hat{x}_k^- is calculated based on the previous state (\hat{x}_{k-1}^-) using a linear difference equation, where A and B correspond to matrices that relate the states in different time steps and relate the state with the variable μ which represent an optional control input [132], respectively. These matrices might be updated in each iteration.

2. Projection of the error co-variance ahead, P_k^- :

$$P_k^- = AP_{k-1} + A^T + Q \quad (3.16)$$

In this second equation, the *a priori* error co-variance P_k^- is computed based on the previous value P_{k-1} , in the matrix A and in the variable Q which represent the process noise co-variance.

The second step, *Measurement Update*, aims to “correct” the models of estimation based on measurements, this is achieved by the following equations:

1. Computation of the kalman gain, K_k :

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (3.17)$$

The kalman gain, K_k , corresponds to a matrix that is computed and used to minimize the P_k value. In this equation the P_k corresponds to the *posteriori* error co-variance, the variable R to the measurement noise co-variance and the matrix H relates the state x with the actual measurement z_k . The kalman gain reflects the change required to be performed in the state estimator. The actual measurement z_k is computed using the following formula:

$$z_k = Hx_k + v_k \quad (3.18)$$

where x_k represents the state and the v_k the measurement of noise associated.

2. Update of the *a posteriori* state estimator, \hat{x}_k :

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (3.19)$$

As illustrated in this equation, the *a posteriori* state estimator \hat{x}_k is calculated based on the sum of the *a priori* state estimator \hat{x}_k^- with the kalman gain, multiplied by the difference between z_k and $H\hat{x}_k^-$. This difference $z_k - H\hat{x}_k^-$ reflects the distinctness between the actual measurement and the estimation and should be minimized along the iterations.

3. Update of the error co-variance, P_k :

$$P_k = (I - K_k H)P_k^- \quad (3.20)$$

This equations aims to calculate the new P_k based on the *a priori* error co-variance, the kalman gain, the matrix H and the I representing the identity matrix.

Artificial Intelligence (AI)-Based

This category includes different type of methods, all of them, related to Artificial Intelligence. The AI field has grown significantly in recent years due to the interest of companies in developing machines able to simulate human behaviour and thinking. AI has several sub-fields like machine learning, natural language processing, pattern recognition, image processing, etc. In these sub-fields, different approaches can be applied depending on the goal and objective of usage. Particularly, in the prognostic field, commonly used approaches include Neural Network (NN), Decision Trees, Support Vector Machine (SVM) and Fuzzy Logic.

Neural Network (NN)-based methods are widely used in prognostics methods due to fact that they can receive any type of input. In addition, the user doesn't need expert

knowledge about the system behaviour in order to model it [133]. The NN establish relationships between the input and the desired output, adjusting the parameters and weights for optimal performance. In prognostic fields, the input may correspond to raw data extracted from sensors or measurements and by simulating the NN, the obtained output by the user may correspond to the desired prediction or future values of the measured data. Another advantage of the NN is the capability of representing and modeling non linear relationships between input and output, this might be particularly important for the prediction of future states in systems that have a non linear behaviour, like, for example, the degradation of certain components in an aircraft [131].

For each Neural Network there are several parameters that should be changed according to the expected output, namely: number of inputs, number of neurons, number of layers and the activation function of each layer.

The most known Neural Network architectures [134] includes Feedforward (FF) [135], RBFN [134] and Recurrent Neural Network (RNN) [69].

Figure 3.10 is an example of a Feedforward Network. These type of architectures are the simplest in the NN field. As illustrated by Figure 3.10 the data is transmitted in one direction, from the input nodes to the output nodes, passing through possible hidden layers. The FF are applied to fields like computer vision and image processing [136].

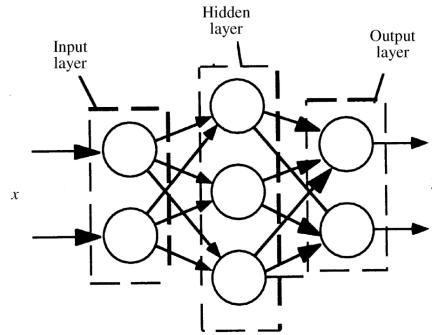


Figure 3.10: example of a Feedforward Neural Network (from *Diagnostic rule extraction from trained feedforward neural networks* [135])

The Radial Basis Function Network (RBFN), illustrated in Figure 3.11, is also well known and used in different research areas. This architecture, contrary to the common NN, like the FF, has a specific and unique activation function, the radial basis function. The radial basis functions take in consideration the distance of points to a specific center [137], that is updated over the iterations using the *k-means* clustering technique. The NN output is computed by performing the weighed sum of the output from the RBFs, applied to each of the neurons in the hidden layers.

The RNN is another widely used NN architecture, represented in Figure 3.12. The main distinction of this type of NN architecture is the feedback connections. These connections bridge the hidden/output layers to the previous layers, which allows the storage and usage of information between consecutive time steps. The training is performed by propagating the error regarding the neurons response and desired response, which results in the updating of the respective weights [69]. These types of NN are commonly used in the speech recognition field [138].

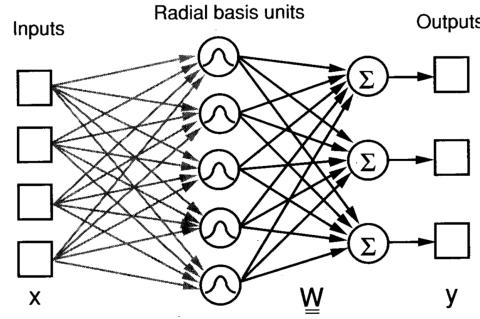


Figure 3.11: example of a Radial Basis Function Network (from *Radial basis function networks for classifying process faults* [137])

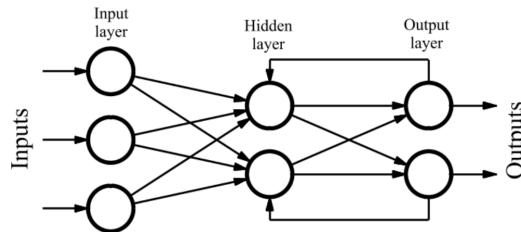


Figure 3.12: example of a Recurrent Layer Network (from *Computational modeling of machining systems* [139])

Decision Trees are another known sub-field of AI. The decision trees are used as classifiers to make decisions, based on knowledge extracted from the data and represented in a tree structure. These trees present all the possible decisions and their consequences [140]. At the end of the process, a decision tree will be generated, where the final decisions to be taken are presented in the leaves (end nodes).

In Figure 3.13 a simple decision tree is illustrated. The continuous rectangles correspond to the *decision blocks*, the dashed rectangles correspond to the *terminating blocks* or *leaves*, the lines are denominated as *branches* and they connect different blocks, the words associated with the lines are the different *classes* of the chosen feature.

The trees are constructed based on the data. Using the data, certain rules are created, which define the way that decisions are made. These rules simulate the human expert reasoning, which is why decision trees are considered machine learning algorithms.

There are different methods for constructing the decision trees, like the ID3 and C.4.5 [142]. Figure 3.14 represents a pseudo code for the function *createBranch()* that generalizes the tree construction.

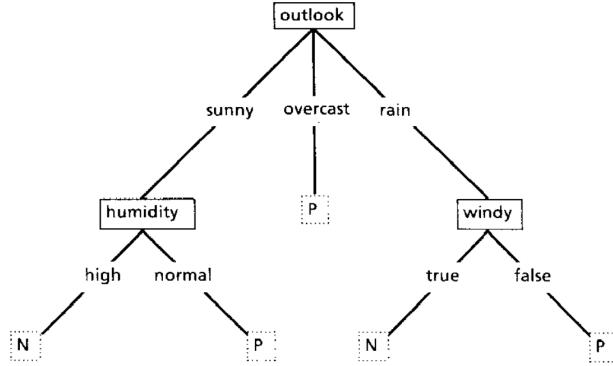


Figure 3.13: example of a Decision Tree (from *Induction of decision trees* [141])

```

Check if every item in the dataset is in the same class:
If so return the class label
Else
  find the best feature to split the data
  split the dataset
  create a branch node
  for each split
    call createBranch and add the result to the branch node
  return branch node
  
```

Figure 3.14: Pseudo code of *createBranch()* function (from *Machine Learning in Action* [143])

In general terms, a feature is initially selected for splitting the data, and the data is split in subsets according to the different feature values. No further development is required if all the subset classes are the same, if not, the subset needs to be split again according to a new selected feature and the process starts over again. The iteration ends when all the subsets are classified [143].

An important step in the construction of the decision tree, is the splitting criteria, different algorithms may have different techniques for choosing which feature to use in the splitting phase. Common splitting criteria include the impurity based criteria, and the information gain [142].

The main advantages of the decision trees are the ease of understanding and interpreting of the resulting decision trees, the low computational cost and the facility of dealing with irrelevant features by not selecting them. The main disadvantage is the possibility of overfitting, specially if there is a low quantity of data available and a high number of nodes created.

Another used technique is **Support Vector Machine (SVM)**. This technique is a supervised learning method for classification, that aims to find the Optimally Separating Hyperplane (OSH) that successfully divides the different data classes [144].

The optimal plane is defined in a way that the separation between classes (margin) is maximized, as such, the probability of the plane leading to wrong classification of the data, is lower. The points closest to the separation plane are called support vectors and

they have influence in the position of the OSH.

In Figure 3.15 there is an example of a 2 dimensional plane with a line that optimally separates the two considered classes.

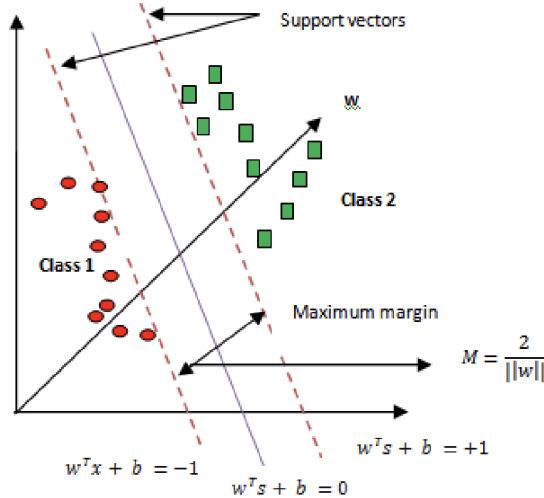


Figure 3.15: example of a SVM hyperplane (from *Recognition of Emotions from Human Activity Using STIP Feature* [145])

As can be seen the optimal line is defined by the formula:

$$w^T x_i + b = 0 \quad (3.21)$$

The line generated by the support vectors labeled as red is defined by:

$$w^T x_i + b = -1 \quad (3.22)$$

And the line representing the support vectors for the green data corresponds to:

$$w^T x_i + b = 1 \quad (3.23)$$

The margin corresponds to the distance between these two support vector lines and is represented by $M = \frac{2}{\|w\|}$.

As previously stated, the goal of Support Vector Machine (SVM) is:

1. Create a decision boundary (OSH) that correctly classifies the data.
2. Maximize the margin value.

As such, the values of w and b need to be calculated taking these two goals into consideration.

Mathematically, this problem is represented in the following form:

$$\begin{cases} \text{minimize} & \frac{1}{2} w^T w, \\ \text{subj.to} & t_i(w^T x_i + b) > 1, \forall i = 1, \dots, n \end{cases} \quad (3.24)$$

Using this formulation, the margin is maximized (as $M = \frac{2}{\|w\|}$) and the correct classification is assured by the only constraint $t_i(w^t x_i + b) > 1$. In this constraint t_i corresponds to the data target which has the values of -1 or +1, instead of 0 and 1. This way, when multiplying t_i by the output $(w^t x_i + b)$, the result will be positive if the two are the same (which means correct classification) and negative otherwise [146].

When the data contains more than two dimensions, a linear separation of the data is not enough, and so kernel functions are used to work with the extra dimensions. These kernel functions map the feature space in a high dimensional space, using the inner product of the new vectors and then finding the surface hyperplane that maximizes the margin in that high dimensional space [147].

Besides the linear kernel (illustrated in Figure 3.15), other known kernels are Polynomial kernels, Sigmoid kernels and Gaussian Radial Basis Function (RBF) kernels [148]. Each of these kernel functions follows a different strategy to perform the high dimensional mapping. The kernels formulas are represented in Table 3.1.

Table 3.1: Kernel Formulas

| Kernel Function | Formula |
|-----------------|--|
| Linear | $K(x, x') = (x, x')$ |
| Polynomial | $K(x, x') = (\gamma(x, x') + r)^d$ |
| Sigmoid | $K(x, x') = \tanh(\gamma(x, x') + r)$ |
| RBF | $K(x, x') = \exp(-\gamma \ x, x'\ ^2 + C)$ |

With the objective of improving the classification accuracy, in each kernel function, certain parameters need to be tuned in order to achieve their optimal values. Particularly, C and γ parameters can have significant influence in the methods performance [149].

- C : The C parameter corresponds to the cost variable, as it reflects the penalty of misclassification in the data. If the C values are small, it admits some misclassifications in trade of a smoother decision boundary. If the C values are big, the misclassifications will have a bigger penalty cost and thus it can lead to overfitting.
- γ : The γ parameter, also named as kernel parameter, reflects the number of features (support vectors) in the creation of the decision boundary. If the γ values are small it means that points far from the decision boundary will be considered to calculate the best decision boundary. On the opposite, if the γ values are high it gives more importance to the points near the boundary, which can result in data overfitting.

Thus, these hyperparameters should be chosen in accordance to each specific scenario in order to achieve their optimal values, and consequently, better accuracy results.

SVM are used in different fields as they can be useful in different contexts, like handwriting recognition and face detection. The main advantages of SVM are the good generalization capacity and the ease in interpreting the results. The biggest drawbacks is the required expertise to tune the parameters in order to obtain good results.

One last group of AI-based technique is **Fuzzy Logic**. It is a decision-making method similar to human reasoning, as a decision can be made between 0 (No) and 1 (Yes), i.e., there can be different states of truth (partial truth) [25]. Through the definition of a fuzzy set and rules, an input space can be computed to an output space.

Figure 3.16 illustrates a structure of a Fuzzy Logic System. As can be observed, the crisp inputs, normally retrieved by sensors, are made available to the fuzzifier that will transform

the crisp inputs into fuzzy set inputs. These will be the input to the Fuzzy Inference module, that will simulate the human reasoning, based on the established rules, the output of this module will be the fuzzy set output. This output will then be transformed back to crisp values (crisp output) by the defuzzifier.

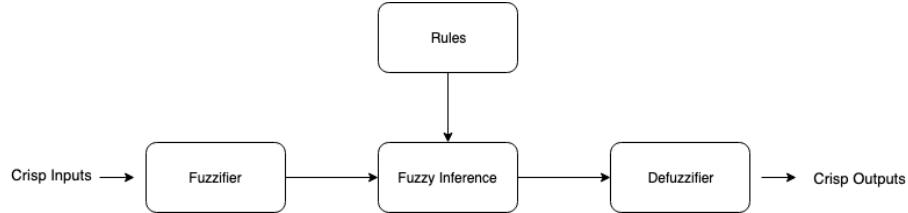


Figure 3.16: Fuzzy Logic System example (adapted from *Fuzzy Neural Networks for Real Time Control Applications* [150])

Fuzzy logic algorithms can be applied to different contexts. One real case scenario that is particularly known for the suitable application of fuzzy logic, is the control of the room temperature, using an air conditioner system. To elaborate a fuzzy logic algorithm, the following steps should be considered [151], [152]. For a better understanding, these steps will be applied to the air conditioner case scenario:

1. Definition of the terms and variables:

The variables correspond to the input and output of the system. In an air conditioner system, the input variable is the Temperature, that can have different terms like: Very Cold, Cold, Warm, Hot and Very Hot.

2. Construction of the membership function:

The membership function defines the behaviour of the terms of the considered variable. They can follow different forms like Gaussian, trapezoidal and triangular functions [150]. Figure 3.17 shows an example.

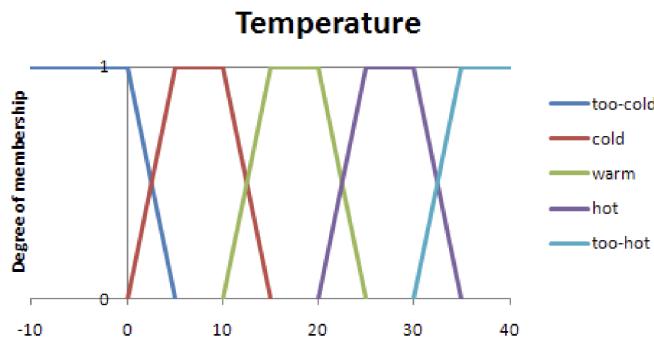


Figure 3.17: Temperature Memberships (from *Temperature Control using Fuzzy Logic* [151])

3. Construction of the fuzzy rules:

The fuzzy rules contain an IF-THEN structure, and they define the output variable. This is an important component in order to simulate the human reasoning. Table 3.2 shows examples of possible rules.

4. Fuzzification:

The fuzzification process corresponds to the conversion of crisp data to fuzzy sets.

Table 3.2: Fuzzy Rules

| Condition | Output |
|--|-----------|
| IF temperature = Cold AND target = Hot THEN | Heat |
| IF temperature = Very Hot AND target = Warm THEN | Cool |
| IF temperature = Warm AND target = Warm THEN | No change |

Each rule is then evaluated using the knowledge base, and the result of each rule is combined, using fuzzy set operators, to achieve a final value. The most used fuzzy set operators are the OR, which corresponds to a MAX operation, and an AND, which corresponds to a MIN operation.

5. Defuzzification:

The defuzzification is the last step and it corresponds to the conversion of the computed fuzzy value into a crisp value. This is performed using the membership function established for the output variable. Figure 3.18 illustrates an example of the defuzzification method.

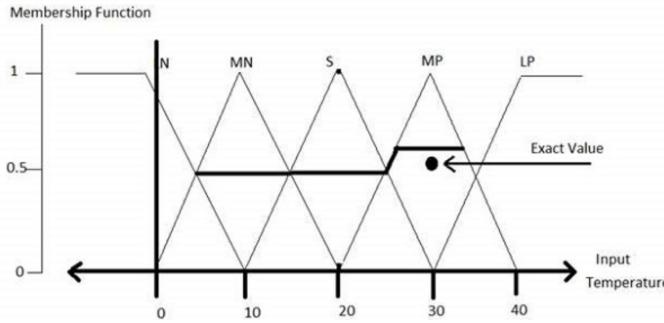


Figure 3.18: Defuzzification method (from *Artificial Intelligence - Fuzzy Logic Systems* [152])

There are different methods for defuzzification [153], [154]. These include methods like the Center of Gravity (COG), which computes the central point that divides the resulted output fuzzy set into two parts with the same area [155]; the Mean Of Maxima (MOM), that computes the average of the elements in the fuzzy set, where the height is the highest [154] and the Root Mean Square (RMS), which calculates the mean square value of the squared values of the output fuzzy set [156].

The fact the concepts behind fuzzy reasoning are theoretically simple (in cases with low complexity) is one of the reasons for using fuzzy logic. Also the limited need of expert knowledge and the similarity with human reasoning, are other reasons for the use of Fuzzy Logic [157]. On the other hand, fuzzy logic also has disadvantages in its usage, like the high number of parameters to deal with, the high running time of the system and the complexity in designing and understanding the fuzzy system, when the number of inputs is significantly higher [150].

Markov-Based

Markov-Based methodologies are based on memoryless Markov processes. There are some relevant variations like **Markov Chain Model** and **Hidden Markov Model (HMM)**.

Markov Chain Model is a stochastic model that describes the probability distribution of the state transitions of a system. It follows the Markov property, which states that the future system state only depends on the current state, and not on events that occurred in the past. It is mathematically expressed as the following:

$$P(X_{n+1} = x | X_0, X_1, X_2, \dots, X_n) = P(X_{n+1} = x | X_n) \quad (3.25)$$

where x is the system state, and $X_{0\dots n}$ is a set of random variables, that represent previous states. Furthermore, in order to compute the probability of a sequence of states s_1, s_2, \dots, s_n the following formula can be used:

$$P(s_1, s_2, \dots, s_n) = \prod_{i=1}^n P(s_i | s_{i-1}) \quad (3.26)$$

Another important concept of the application of Markov chain models is the transition Matrix. This corresponds to a matrix with dimension $i*y$ that contains the probability of consecutive actions to happen. One particular case study using Markov chain models is the weather prediction [158].

The following matrix shows an example of a transition matrix A for weather prediction [159].

$$A = \begin{matrix} & \text{Rainy} & \text{Windy} & \text{Snowy} \\ \text{Rainy} & \left(\begin{array}{ccc} 0.5 & 0.5 & 0.25 \end{array} \right) \\ \text{Windy} & \left(\begin{array}{ccc} 0.5 & 0 & 0.5 \end{array} \right) \\ \text{Snowy} & \left(\begin{array}{ccc} 0.25 & 0.25 & 0.5 \end{array} \right) \end{matrix}$$

Each row represents the current day weather and each line represent the weather in the next day. As can be concluded, for example, the probability of snow tomorrow, knowing that today is rainy is 0.25.

Using the equation 3.26 it is possible to predict the weather j days from now. For example, the probability of rain knowing that today is windy is:

$$P_{w_r} = P(w, r, r) + P(w, w, r) + P(w, s, r) \quad (3.27)$$

Which can be converted in the following formula:

$$P_{w_r} = P(r|r) * P(r|w) * P(w) + P(r|w) * P(w|w) * P(w) + P(r|s) * P(s|w) * P(w) \quad (3.28)$$

Furthermore, using the transition matrix this formulation can be simplified. Assuming that the weather today can be represented by a initial state distribution $\pi = [1 \ 0 \ 0]$, the P_{1_3} can be calculated in the following way:

$$P_{1_3} = P_{1_2} * P_{2_3} \quad (3.29)$$

Considering that $P_{1_2} = \pi * A$ and $P_{2_3} = P_{1_2} * A$, the previous equation can be simplified in the following formula:

$$P_{1_3} = \pi * A^2 \quad (3.30)$$

When calculating the matrix P_{1_3} it is possible to identify the probability of rain 3 days from now, which makes the computation for the weather prediction simpler.

The **HMM** is also a Markovian-based model which assumes [160]:

1. The Markov property (eq. 3.25).
2. The system state cannot be observed directly, i.e., it is hidden from the observer.

A correlated and observable state is used to infer something about the system's current state, the overall diagram can be seen in Figure 3.19, where $S_{0..T}$ are the hidden states from time 0 to T and $Y_{0..T}$ are the variables observed from time 0 to T [158].

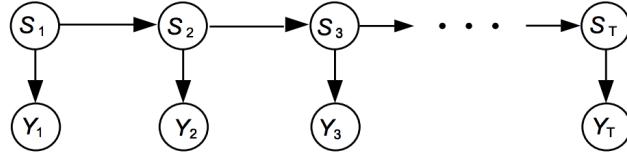


Figure 3.19: HMM diagram (from *An Introduction to Hidden Markov Models and Bayesian Networks* [158])

According to literature, a Hidden Markov Model (HMM) can be defined through five elements [161] [162]:

1. Number of states, K :

As mentioned before, these correspond to hidden states that cannot be directly observed. The hidden states S of dimension $K * 1$ follow the Markov property which states that the state S_t only depends on the previous state, i.e., the S_{t-1} .

2. Number of observations, Ω :

The observations Y of dimension $\Omega * 1$ corresponds directly to measurable states, that can help acquire insights over the probability of the occurrence of a sequence of states S .

3. State transition matrix, A :

Similarly to the previous section, this matrix gives the probability of transitioning from a specific state $S_{t-1,i}$ to a state $S_{t,j}$. This transition corresponds to the following formulation:

$$A_{i,j} = P(S_{t,j} = 1 | S_{t-1,i} = 1) \quad (3.31)$$

Besides the matrix representation, the state transition distribution can also be represented using a diagram, illustrated in Figure 3.20.

4. Observation model, B :

The B is a matrix with size $\Omega * K$ that describes the probability of making the observation $Y_{t,k}$ given the hidden state $S_{t,j}$. This can be formulated as:

$$B_{k,j} = P(Y_t = k | S_t = j) \quad (3.32)$$

5. Initial state transition, π :

The state transition π corresponds to a $K * 1$ vector that describes the initial probabilities. Mathematically:

$$\pi_i = P(Y_{1i=1}) \quad (3.33)$$

An example of initial state transition π can be $\pi = [0.75, 0.25, 0]$.

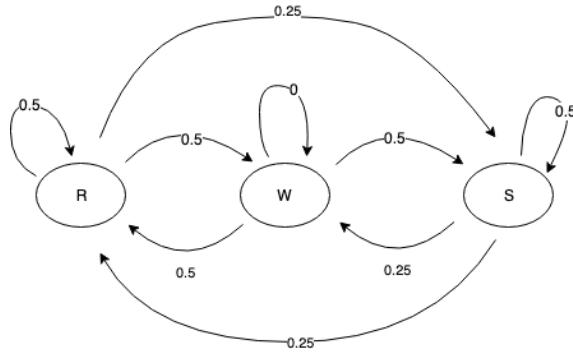


Figure 3.20: State transition diagram

These five elements totally describe the Hidden Markov Model (HMM). Nevertheless, it can be simplified into three elements [161]:

$$\lambda = (A, B, \pi) \quad (3.34)$$

Bayesian Networks

Several methods, already identified, use the probability theory for creating predictions. One particularly important method for predictions, which is heavily based on probabilities, is the Bayesian networks. These use the probability theory in order to create classifiers, able to classify data based on probabilities [143].

One relevant concept for the understanding of Bayesian ideologies is the perception of the Conditional Probability.

Conditional probability allows the calculation of the probability of a particular event to happen given that another dependent event has already occurred. Mathematically, this is expressed in the following form [146]:

$$P(c|x) = \frac{P(c,x)}{P(x)} \quad (3.35)$$

where c and x are two distinct events.

Furthermore, using this formula is possible to derive the Bayes Rule:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (3.36)$$

which is the basis of the application of Bayesian networks for prediction.

A Bayesian network is composed of a set of nodes that are correlated and dependent. Each transition between nodes has a certain probability associated [143].

Figure 3.21 illustrates an Bayesian Network representing variables that influence and help concluding if the student is ready for the exam or not [143]. In this network, the event 'A' indicates whether or not the student has participated in the course lectures, 'B' states whether course was boring or not, the event 'R' indicates whether the lectures were revised or not, and event 'S' reflects whether the student is scared or not of the exam.

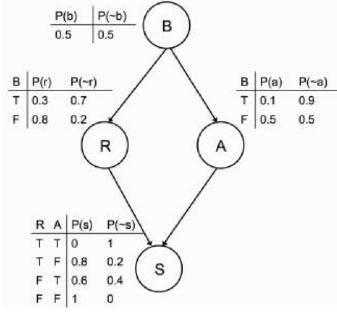


Figure 3.21: example of a Bayesian Network (from *Machine Learning - An Algorithmic Perspective* [146])

On the analysis of the network graph is possible to make observations regarding the dependence between the different events.

- The event 'B' is not dependent of any other event.
- The events 'R' and 'A' are both dependent of the event 'B'.
- The event 'S' is dependent of events 'A' and 'R'.

These observations allow the computation of the probability of the student being scared of the exam (event 'S') under specific conditions. Generally, the probability of the student being scared of the exam is formulated in the following way [143]:

$$P(s) = \sum_{b,r,a} P(b, r, a, s) \quad (3.37)$$

which taking into consideration the previous conclusions regarding dependencies, can then be formulated as:

$$P(s) = \sum_{b,r,a} P(b) * P(r|b) * P(a|b) * P(s|r, a) \quad (3.38)$$

Assuming some facts, for example, that the course was boring (which means $P(b = True) = 1$), it is possible to ignore some equation terms and calculate the probability of the student being scared for the exam:

$$P(s = T) = \sum_{r,a} P(r|b) * P(a|b) * P(s|r, a) \quad (3.39)$$

By decomposing the equation terms, the following equation is obtained:

$$\begin{aligned}
 P(s = T) = & P(r = T|b = T) * P(a = T|b = T) * P(s = T|r = T, a = T) + \\
 & P(r = T|b = T) * P(a = F|b = T) * P(s = T|r = T, a = F) + \\
 & P(r = F|b = T) * P(a = T|b = T) * P(s = T|r = F, a = T) + \\
 & P(r = F|b = T) * P(a = F|b = T) * P(s = T|r = F, a = F)
 \end{aligned} \quad (3.40)$$

Finally by replacing the equation with the correct probabilities from Figure 3.21, the following equation is formulated:

$$P(s = T) = 0.3 * 0.1 * 0 + 0.3 * 0.9 * 0.8 + 0.7 * 0.1 * 0.6 + 0.7 * 0.9 * 1 = 0.89 \quad (3.41)$$

The probability of the student being scared given that the course was boring is 0.89.

Overall, the advantages of the application of Bayesian networks are that they can work with several classes (although it can escalate very quickly) and work with a low quantity of data. The main disadvantage is that it can be computationally expensive, specially with high dimensional data [143].

3.6.3 Hybrid Prognostic approach

As the name suggests, this approach combines both Model-based and Data-Driven methodologies in order to predict the RULs values.

In real-world scenarios, sometimes the trends of the components behavior are complex and difficult to predict with a single RUL estimation method. A hybrid approach aims to solve this problem by using multiple techniques of RUL estimations altogether [122].

The aim of this technique is to overcome the limitations of a single prediction method, improving the accuracy of the RUL predicted. The fusion of techniques can be applied in the different stages of the PHM algorithm implementation, like in the data extraction, data analysis and model training.

Comparing the suitability of the application of the three types of PHM approaches within this work, the data driven methods are the most appropriate.

The Model-based methods are not appropriated for this work, as physical knowledge regarding the aircraft systems is required in order to develop the prediction models. In addition, the fact that the provided data is anonymized, increases the complexity and difficulty in applying a models-based methodology.

Data driven approaches are the best option to be follow in this work as no expert knowledge of aircraft systems is required and a big quantity of data, from aircraft systems, is provided for this work. Furthermore, it is expected that the provided sensors data reflects the system condition, and thus, allows the development of methods for the diagnostics and prognostics of the system condition.

Following the same idea regarding the models-based, the hybrid models are not considered fitted to this work.

3.6.4 Types of learning

In the Artificial Intelligence field, different methodologies can be implemented for particular purposes. These methods can follow distinct types of learning, depending normally of the existing data, the most accepted are [133]:

- **Supervised learning:** In this type of learning the data is labeled, which means that the true state of the data instances is known.

This way models that map the input into output can be learned from the data. Examples of techniques following this type of learning are *Regression* and *Classification*.

- **Unsupervised learning:** In the unsupervised learning the data is unlabeled as the classes are not known. This type of learning tries to find some pattern or distribution from the data in order to learn more about it. Typical examples are *Clustering* and *Association Rules*.
- **Semi-Supervised learning:** This is a combination of Supervised Learning and Unsupervised Learning due to the fact that some data is labeled and other is not.
- **Reinforcement Learning:** The *Reinforced Learning* is based on actions. The agent is continuously learning from the interactions with the environment and the goal is to take actions that maximize a reward function. One well-known algorithm that uses Reinforcement Learning is *Q-Learning*.

3.7 Performance Metrics

In order to assess the accuracy in the predicted RULs, some performance metrics were established. Due to the dataset characteristics, these were only applied on the NASA's Turbofan dataset.

The presented metrics are traditional performance metrics, that evaluate independently the final estimated RULs. The following three are described:

- **Score:** The PHM Conference 2008 organization created a formula that calculate the prediction score comparing the estimated *RULs* and the true *RULs*. This metric values have not a specific meaning, they are only used to compare results with other approaches developed using the same dataset, considering it as a benchmark.

The scores of the approaches developed to the PHM 2008 Data Challenge were already published [163].

The Score formula is the following:

$$\begin{cases} \sum_{i=1}^n e^{-(\frac{d(i)}{13})} - 1, & d < 0 \\ \sum_{i=1}^n e^{(\frac{d(i)}{10})} - 1, & d \geq 0 \end{cases} \quad (3.42)$$

where $d(i) = \text{Estimated RUL} - \text{True RUL}$, of unit i .

Unfortunately, this metric can be applied only to the Turbofan engines datasets.

- **Root Mean Square Error (RMSE):** This metric calculates the root of the average squared of the difference between the estimated values and true ones. It measures the average amplitude of the error and can be applied in the PHM context. The formula is the following:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3.43)$$

where n is the number of samples, y_i the estimation and \hat{y}_i the ground truth.

Chapter 4

Proposed PHM Approach

As mentioned in the previous Chapters, the goal of this work is to develop a PHM methodology, encompassing the CBM principles. This PHM system is expected to be able of accurately diagnosing and predicting the health condition of specific aircraft systems based on the sensors data.

Thus, this Chapter presents and details the approach followed in this work for the PHM development, in particular, the different steps considered useful to integrate the PHM models.

The correct analysis, estimation and prediction of the health condition of the considered aircraft subsystems requires the execution of a sequence of tasks on the original dataset. These actions process and transform the data in order to extract the relevant information regarding the system condition, that is used for the computation of the RUL associated with the respective aircraft component.

As illustrated in Figure 4.1, the pipeline initially requires, as input, the dataset containing the sensors data from the component in analysis, followed by the sequence of tasks: Data Preprocessing, Feature/Sensor Selection, HI computation and finally the RUL computation. In the end the output should be the estimated RUL.



Figure 4.1: Generic RUL pipeline

This pipeline is used as a guideline in the different case studies for the implementation of the PHM methodologies, capable of diagnosing and predicting the health condition of the aircraft systems considered in each of the case studies.

Each step in the pipeline is going to be described in the following Sections.

4.1 Datasets

The dataset is an important aspect of this work as it is the basis of the PHM system to be implemented for the diagnosis and prognosis of the condition of particular aircraft subsystems, following a CBM approach.

In order to perform an intelligent analysis of the data retrieved from the aircraft sensors, with the objective of diagnosing the different aircraft components' state and predicting possible failures, the data presented in the dataset should correspond to raw sensors data and must reflect a specific component/subsystem condition.

Furthermore, the dataset must contain data specific to a discrete and sequential time frame and should be collected from different airplanes of the same model aircraft in order to guarantee the same structural organization and identical mode of operation. Ideally, the respective aircraft parts shouldn't be replaced during that specific time frame, so that the degradation behavior of certain components can be analyzed and modeled over time.

The dataset should encompass sensors data extracted from specific subsystems of the aircraft in order to analyze, interpret and diagnose the condition of that particular subsystem, and predict possible failures. The techniques proposed in this work are focused on specific subsystems and not the whole aircraft system due to its complexity and comprehensiveness.

4.2 Data preprocessing

The preprocessing step is proposed in this approach, as a way of filtering and preparing the data for the application of the machine learning methods for the diagnosis and prognosis of the systems condition.

The data preprocessing step is performed by adjusting and transforming the raw data into a clearer format, eliminating possible noise or outliers present in the original data.

Due to the different data ranges and the fact that for each phase or aggregated phases of the flight different data patterns are identified, the analysis with regards to the preprocessing should be performed individually for each phase or set of phases.

4.2.1 Outliers detection and accommodation

As the dataset is expected to correspond to sensor data retrieved directly from the aircraft sensors at a specific sample rate, it is likely that the raw data contains embedded noise due to the logging and extraction flaws. Thus, the application of techniques for detecting and accommodating these outliers should be valuable and useful in this approach.

As mentioned before, the analysis should be performed individually for each flight phase or a set of flight phases, with the goal of isolating and analysing particular data patterns.

As such, a sliding window approach is used for detecting and adjusting the identified outliers. Figure 4.2 illustrates the overall flowchart of the detection and accommodation algorithm applied to the sensors data.

For each aggregated phase and a specific sensor regarding a specific flight, a sliding window approach was followed. This means that in each iteration, the mean and standard deviation of the data framed in the sliding window are calculated, then the following value is compared to the calculated window metrics. This value is then adjusted using the criteria:

- If $next_value > \mu + 3\sigma$: The next value is considered an outlier and so is adjusted, in which, $next_value$ becomes equals to $\mu + 2\sigma$.
- Else if $next_value < \mu - 3\sigma$: The next value is also considered an outlier and so is adjusted, in which, $next_value$ becomes equals to $\mu - 2\sigma$.

- Else: The next value is between the considered thresholds, hence is not considered an outlier

After analyzing the next value, the sliding window is shifted one unit forward and thus includes the value analyzed before. The sliding window starts at the beginning of the time range being analyzed and shifts forward until it reaches the end of the considered interval.

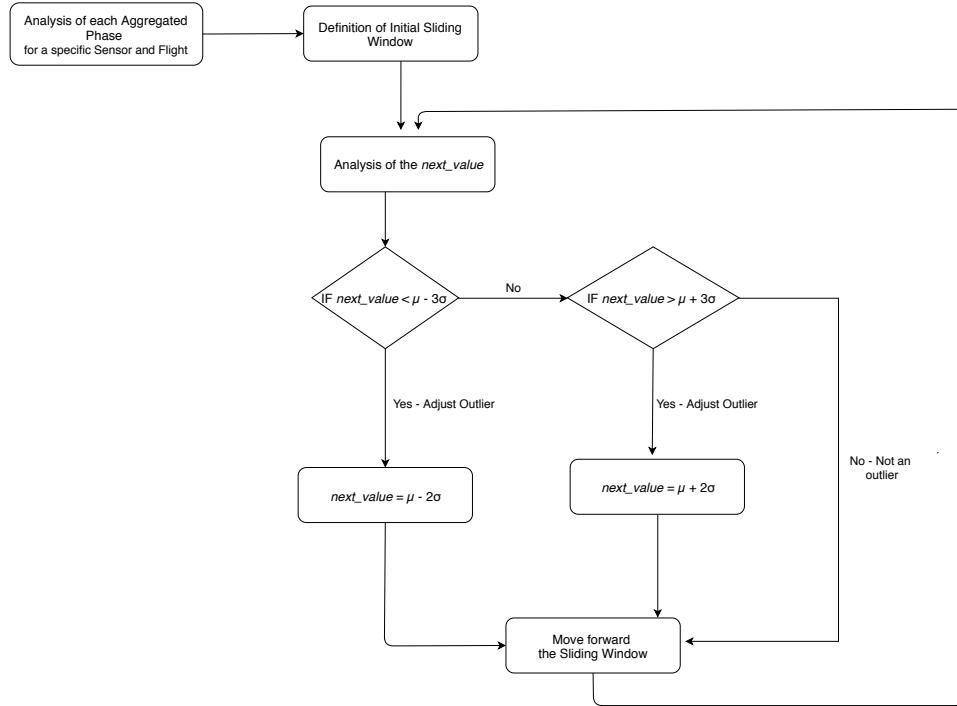


Figure 4.2: General flowchart of the outliers detection and accommodation

4.2.2 Flight Categorization

Another important preprocessing step corresponds to the correct identification and labelling of the data relative to each flight. This flight categorization is important in order to correctly identify the different phases in each flight, and thus perform a precise analysis regarding the degradation and also the presence of outliers, in each of the considered phases.

As each dataset has a different structure and possibly a different terminology and criteria for the number of phases considered in a regular flight, in each case study, an initial analysis of the information regarding the flights labelling needs to be performed. If it is concluded that the flights labelling is not accurate, an algorithm for a new and correct flight labelling needs to be performed, due to its impact in the following pipeline, specifically, in the computation of the HI, and consequently in the obtained results.

4.3 Feature selection

The next step in the pipeline is the feature selection. This step is important as it selects the most relevant features and removes possible redundancy and correlation in the data.

Similarly to the previous step, the choice of the feature selection algorithms to use depends on the characteristics of the data, specifically, the number and meaning of the features in the dataset.

In the case where the dataset contains a significant number of features, features selection should be performed in order to, besides removing redundancy, decrease the problem complexity. From all the different methods for feature selection, the Kruskal Wallis is evidenced, as it is a simple, and computationally less expensive than the common methods [164].

In the case the dataset containing multiples features referring to the same sensor but with respect to different systems, a correlation criteria should be convenient to use in order to detect which features refer to the same sensor. This way, the identification of the features specific to each subsystem may be performed.

Nevertheless, certain datasets may exist where the features are already compact and concise, maintaining their relevancy for the problem in question. In these specific cases, feature selection is not necessary to be implemented.

4.4 Health Indicator (HI) computation

In the aircraft context, the aim of computing the HI is to identify the health condition of certain aircraft subsystems, based on the sensors data.

Following the same idea as in the previous sections, the choice of the best approach for the HI computation depends on the data available, its format and the system physical structure.

If, from the considered system and respective data, there is a clear and evident feature that is highly correlated with the degradation occurred over time, the values regarding this variable can be used directly as the HI value. Generally, these conclusions are based on physical knowledge of the considered system. Thus, in this particular case scenario, a PHI approach is appropriate.

Nevertheless, when there is no expert knowledge regarding the system operation, information of different variables should be combined in order to compute the system health condition, resulting in the calculation of a VHI.

If from the dataset, the different degradation trajectories are well identified, that is from the moment when the component is new until the moment when it becomes inoperable due to degradation or failure, and the degradation pattern is known, a well used approach is to model its behaviour (degradation) using linear or exponential regression, based on the sensors data. This way, a standard degradation pattern is used as a baseline to model the degradation evolution over time.

In the case where, from the dataset, the different degradation trajectories are not well determined, a possible option is to focus the analysis in time domain features, as mean, standard deviation, and minimum. This way it is possible to detect, from the sensors data patterns, certain anomalies in the data that can indicate situations of extra degradation.

As can be concluded, the approach to follow in order to compute the HI of a specific component depends on multiple variables, such as the characteristics of the dataset, the knowledge regarding the system, and the relevancy of the type of information provided in the dataset.

4.5 RUL computation

The last approach of the PHM pipeline is the computation of the RUL. Following the pipeline organization, it receives as input the HI value and computes the number of flight hours remaining of useful life of the component being analyzed.

The information regarding the RUL is very useful for the maintenance team as it provides more information regarding the component condition, which, ideally, helps to perform more effective and optimized interventions.

Between the multiple approaches used for the RUL computation, the choice of the most suitable approach to use depends on the data characteristics, its structure and also the on HI meaning and its formulation. In particular, the degradation patterns, represented in the HI values, vary between the aircraft systems, as its degradation behavior may have a more linear trend or a more exponential trend. Thus the RUL computation method used should be based on the nature of the degradation, which is reflected in the HI evolution over time.

Hence, if the HI evolution follows a linear trend, then an linear extrapolation should be used, for predicting the HI future behavior.

Differently, if the HI evolution follows an exponential trend, that can be accurately represented and expressed by a polynomial of order two or three, then an extrapolation of order two, or three (depending on the error associated), should be applied for computing the RUL.

Nonetheless, in a real and structural complex system, the degradation may be characterized by irregular trends due to influences of external variables on the system. In these cases, the extrapolation method is not appropriate, as it is simplistic and does not take into consideration the impact of the external variables. Thus, the best method for computing the RUL may be a distance-based approach, also called a similarity-based approach. Using a similarity criterion, the goal is to assess and identify which degradation behavior presented in the training trajectories, better fits and describes the degradation behavior of the test trajectory, whose future degradation behavior needs to be predicted. As the training trajectories correspond to the entire degradation cycle, a set of possible degradation trends is obtained and can be used to compute the RUL of the test trajectory.

This page is intentionally left blank.

Chapter 5

Exploratory Case Studies

Chapter 5 presents the preliminary work performed with the Turbofan Dataset, available in NASA's Prognostics Data Repository [104] and the Brakes dataset, provided within the ReMAP context.

This exploratory work was performed with the intention of gaining some knowledge, experience and insights regarding different aircraft data structure, so that when the main dataset became available, some introductory work had already been carried out.

In fact, the main contribution of this work is the development of machine learning methodologies for the diagnosis and prognosis of aircraft components condition applied and tested in the Air Bleed Dataset, which is one of the 13 systems considered in the ReMAP proposal. Nevertheless, while the dataset was not available, the Turbofan dataset and the Brakes dataset were used for experimenting and testing different approaches regarding the diagnosis of the systems' health condition, and consequently, the RUL computation.

Henceforth, Section 5.1 presents the Turbofan Dataset and describes the methodologies implemented for the RUL computation.

In Section 5.2, the Brakes dataset is characterized and a simple methodology for the RUL computation is detailed.

5.1 Turbofan Engine System

In this dataset, the data was generated by an engine degradation simulation [165], carried out using Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) [166], with the objective of simulating a turbofan engine. This dataset was provided by the Prognostics CoE at NASA Ames.

In Figure 5.1 it is possible to visualize a simplified diagram of C-MAPSS with all the components considered.

Although the data presented in the datasets is modeled and not extracted from real aircraft systems, there is an approximation to real sensors data. Some important aspects of real aircraft flights, like initial components' wearing and the presence of noise in the system were taken into consideration when creating the model.

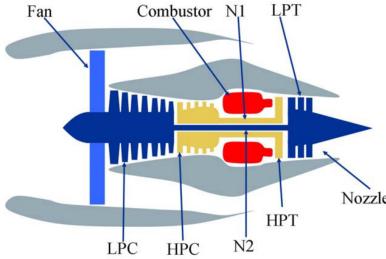


Figure 5.1: Simplified model of C-MAPSS (from *Damage Propagation Modeling for Aircraft Engine Prognostics* [165])

With the objective to explore State of the Art techniques for RUL computation and to gain some sensibility with the data, three data driven prognosis methods implemented for the PHM08 Challenge Competition were reproduced, namely:

1. The winner method (*A Similarity-Based Prognostics Approach for Engineered Systems*) [17].
2. An adaptation of the winner method using a Neural Network-based RUL prediction.
3. An adaptation of the winner method using a Extrapolation-based RUL prediction.

Moreover the technical report *Benchmark of methods applied to PHM08 Challenge Dataset* [15] was written with the goal of analyzing the RUL prediction methods applied and submitted in that competition.

In the next sections, the methodologies applied to the dataset are described, following the pipeline proposed in the Chapter 4, and illustrated in Figure 4.1.

All the parameters, values and techniques were implemented based on the authors suggestions.

5.1.1 Dataset Structure

The Turbofan dataset is composed of four datasets with identical structure that were simulated over different combinations of Operational settings and Fault Modes. Each dataset is made of data from 21 sensors regarding pressures, temperatures and velocities of different components of the Turbofan engine. These sensors values capture the evolution of a fault and the goal is to predict the number of cycles remaining before total system failure, i.e., the RUL. As each dataset contains several aircraft trajectories, a RUL value should be predicted for each one, as a different failure was inserted in each trajectory.

Another dataset, the PHM08 Challenge Dataset, was also used for validating the developed approaches. This dataset is identical in structure and meaning to the Turbofan dataset, but was created specifically to be used for the PHM08 Challenge Competition, as a metric to rank the different methodologies submitted. Thus, this dataset was used for comparing the results obtained in this work, with the results achieved in the competition.

Table 5.1, represents the structure of the two datasets.

As can be observed, each dataset was simulated over different number of Operational settings combinations and Fault Modes, which influences the complexity in the RUL prediction. The number of trajectories in the training dataset and in the testing dataset is also different for each dataset.

Table 5.1: Datasets Structure and Details

| Datasets | | Train Trajectories | Test Trajectories | Operational Settings | Fault Modes |
|------------------------|--------------|--------------------|-------------------|----------------------|-------------|
| Turbofan | FD001 | 100 | 100 | 1 | 1 |
| | FD002 | 260 | 259 | 6 | 1 |
| | FD003 | 100 | 100 | 1 | 2 |
| | FD004 | 248 | 248 | 6 | 2 |
| PHM08 Challenge | | 218 | 218 | 6 | 2 |

Each dataset has a *train* dataset and a *test* dataset. The PHM08 Challenge Dataset provides also a *final_test* dataset for validation, which was used to rank the participants.

Each dataset contains data from several flight trajectories represented by time series. In the start of each time series the engine operates in normal conditions, i.e., without any failure, then at some point of the time series a fault is generated and evolves. In the *train* datasets the fault evolves until a maximum threshold is reached where the turbofan engine cannot operate anymore (total failure), this marks the end of the time series. In the *test* datasets the fault evolves and the time series ends before the threshold is reached. The objective here is to predict, based on the sensors data, the number of cycles remaining until the total failure of the engine, i.e., the RUL, for each test trajectory.

The data is provided in text files containing 26 columns and each row represents one operational cycle.

The information in each column is the following:

- 1) Unit number
- 2) Time, in cycles or hours
- 3) Operational setting 1
- 4) Operational setting 2
- 5) Operational setting 3
- 6) Sensor measurement 1
- 7) Sensor measurement 2
- ...
- 26) Sensor measurement 21

For each row there is information about: *Unit number*, that is, the identifier of the trajectory; *Time*, represented in operational cycles (or hours); 3 different *Operational settings* and 21 *Sensors measurements*.

The three Operational settings, also referred as Operational regimes, refers to: *Altitude*; *Mach Number* and *TRA* and they have an impact in the engine performance. The *Altitude*, as the name suggests, refers to the plane altitude in each moment of the flight, the *Mach Number* is the ratio of the aircraft speed and the speed of sound in the surrounding environment [167] and the *TRA*, which stands for *Throttle Resolver Angle*, is defined as the angular deflection of the pilot's power lever and it is used to control the thrust applied to the aircraft engine [168].

As illustrated in the dataset structure, each operational cycle is influenced by one of the six different combinations of three settings generated by the C-MAPSS model. The combinations applied in the PHM 2008 Challenge Dataset are presented in Table 5.2.

Table 5.2: Operational Settings

| Altitude (K ft.) | Mach Number | TRA (°) |
|------------------|-------------|---------|
| 0 | 0 | 100 |
| 10 | 0.25 | 20 |
| 20 | 0.70 | 0 |
| 25 | 0.62 | 80 |
| 35 | 0.84 | 60 |
| 42 | 0.84 | 40 |

In Figure 5.2 the combinations of the three Operational settings values are represented. As can be observed, there is no relation between each of the Operational settings points as these are dispersed in the 3D space without any direct correlation amongst them. As such, and because they have a significant influence in the engine performance, the degradation behavior of the failures should be different and independent according to the Operational setting in use. Therefore, this will need to be taken into account, when developing the methods for the RUL prediction.

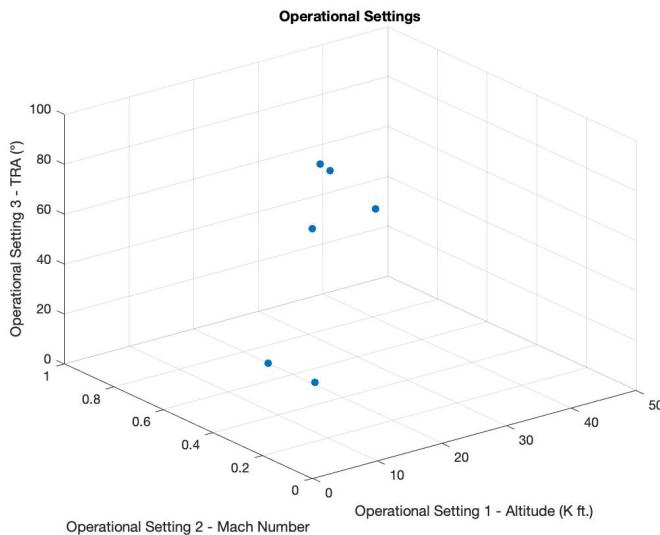


Figure 5.2: Operational settings relation

The details regarding the 21 sensors are described in Table 5.3.

5.1.2 Preprocessing of Turbofan Engine data

As mentioned before, the data presented in the dataset is synthetic as it does not correspond to real aircraft sensors data. Therefore, the presence of outliers and embedded noise still occurs, but does not have significant impact in the obtained results.

Nevertheless, due to the existence of some noise in the sensors data, its impact was attenuated and filtered in the steps of feature selection and also in the filtering of the RUL values obtained, by removing the extreme values.

Table 5.3: Sensors description

| Description | Units |
|---------------------------------|---------|
| Total temperature at fan inlet | °R |
| Total temperature at LPC outlet | °R |
| Total temperature at HPC outlet | °R |
| Total temperature at LPT outlet | °R |
| Pressure at fan inlet | psia |
| Total pressure in bypass-duct | psia |
| Total pressure at HPC outlet | psia |
| Physical fan speed | rpm |
| Physical core speed | rpm |
| Engine pressure ratio (P50/P2) | – |
| Static pressure at HPC outlet | psia |
| Ratio of fuel flow to Ps30 | pps/psi |
| Corrected fan speed | rpm |
| Corrected core speed | rpm |
| Bypass Ratio | – |
| Burner fuel-air ratio | – |
| Bleed Enthalpy | – |
| Demanded fan speed | rpm |
| Demanded corrected fan speed | rpm |
| HPT coolant bleed | lbm/s |
| LPT coolant bleed | lbm/s |

Regarding the flights categorization, the flights (trajectories) labeling is performed in the original dataset. By analyzing the flights labeling, no fluctuations were found in the flight ids, also the labeling was clear and chronologically ordered. Thus no extra preprocessing, with regards to the flight categorization, was required.

5.1.3 Feature Selection

Regarding the feature selection, the approach followed by the authors suggest that, by graphically analyzing the behavior of the sensors values it is possible to identify the relevant features.

If a clear and continuous trend was found for a specific sensor, it should be selected. Otherwise, if some inconsistent trends were displayed it might indicate that that sensor is not useful in predicting the RULs and so it is discarded.

Figure 5.3 illustrates an example of a feature that is relevant, as it shows a clear positive trend. Opposingly, Figure 5.4 illustrates irrelevant features, as different trends are represented in the sensor values.

After this analysis, a set of relevant features is obtained and used during the training of the model. The set of sensors used in this technique (and in the other 2 approaches) are: 2, 3, 4, 7, 11, 12 and 15.

Besides this method of feature selection, a Kruskal Wallis method was also implemented in order to identify the most relevant features using a statistical test, then a comparison of the results was made.

The Kruskal Wallis statistic test analyzes whether the different samples follow the same distribution function, or not, by comparing the mean values [41].

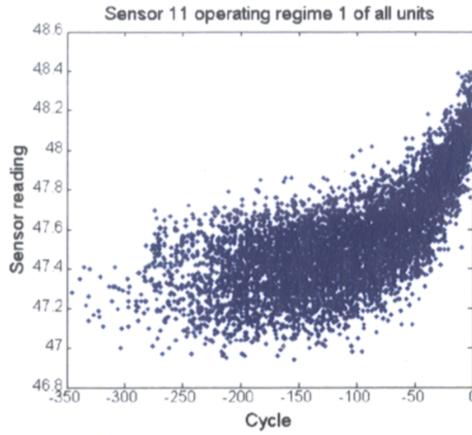


Figure 5.3: Relevant Sensor (from *A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems* [17])

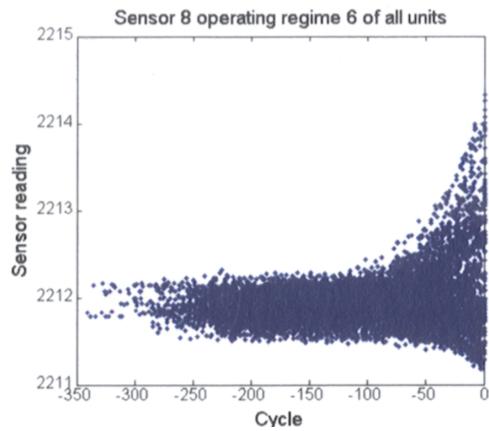


Figure 5.4: Irrelevant Sensor (from *A similarity-based prognostics approach for Remaining Useful Life estimation of engineered systems* [17])

Then, the more discriminative features are selected.

Comparing the results from the graphical analysis and from the Kruskal Wallis test, no major differences were found, as the resulting features set is similar in both approaches.

5.1.4 HI computation

Respecting the HI computation, the same technique was used in the three approaches. Due to the data characteristics the approach followed by the authors is based on a linear regression for modeling the HI evolution over time.

In the authors approach, the selected features are fused in order to produce the HI values, which reflect the system health condition. These are calculated through linear regression models, more specifically, a regression model for each operational regime is generated. The models are fitted using the training data specific to each regime.

The linear regression formulation used is the following:

$$y = a + \sum_{i=1}^N \beta_i x_i + \varepsilon \quad (5.1)$$

where $x = (x_1, x_2, \dots, x_N)$ is a feature vector of dimension N , $(a, \beta_1, \beta_2, \dots, \beta_N)$ are the model parameters and ε refers to the noise term. The y is a health indicator and is binary: it has the value of 0 if, in the training unit, the cycle is near the end of life; and it has the value of 1 if, in the training unit, the condition of the units is new. The remaining cycles are not in the extremes of the units' life and so they are not considered for the fitting of the models. The y values can be calculated, using the following formulation:

$$y = \begin{cases} 0, & C_i^{adj} > C_{max} \\ 1, & C_i^{adj} < C_{min} \end{cases} \quad (5.2)$$

where i is the number of the cycle in the training unit. Following the authors suggestion, the C_{max} has the value of -5 and C_{min} the value of -300.

Using this approach, for each regime, a model was generated, from which the HI is calculated, based on the sensors values.

With regards to the HI computation, the **Neural Network-based approach** incorporates an extra step, which is the training of a Multilayer Perceptron (MLP) for the computation of the HI values in the test trajectories. In this network, the input corresponds to the sensors data and the target to the HI values computed for the training dataset, based on the regressions obtained previously. Figure 5.5 represents the network architecture.

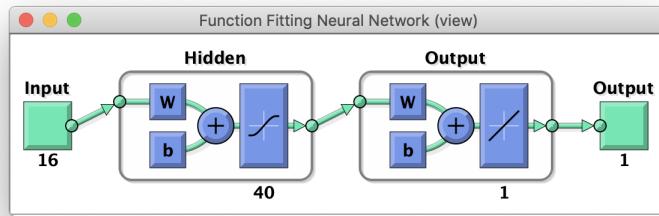


Figure 5.5: MLP Architecture

After training the network, the HI values are obtained for each test trajectory, by simulating the network with the respective sensors data as input.

5.1.5 RUL prediction

Simple State of the Art machine learning techniques were reproduced as preliminary experiments for the RUL computation. Specifically, three different approaches were implemented: a **Similarity base approach**, **Neural Network-based approach** and **Extrapolation-based approach**.

Similarity Based Approach:

In this approach the euclidean distance is used to assess which degradation behavior presented in the training units, better fits the degradation behavior of each test unit.

First, the degradation behavior of each training trajectory is modeled, using the following formulation:

$$y = a(e^{bC^{adj}+c} - e^c) + \varepsilon \quad (5.3)$$

where C^{adj} is the cycle index, (a, b, c) are the model parameters of the training unit i , y is the HI and ε is the noise term.

This exponential regression model is generated for each training unit, thus at the end, a library of models that represent the different degradation patterns in each of the training trajectories is obtained.

Figure 5.6 shows the curves corresponding to the exponential models obtained for the first 10 training units generated.

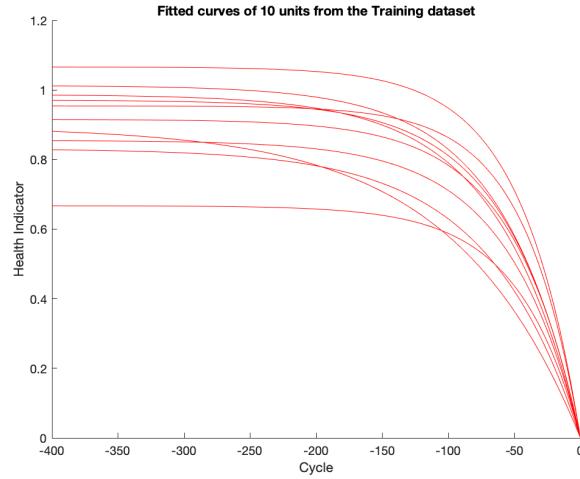


Figure 5.6: HI curves produced for the first 10 training units

Next, in order to compute the RUL, the degradation behaviour of each training unit is compared with each test unit, using the euclidean distance.

For each test unit, the most similar training trajectories, that is, with lower distance compared to the test units, comprise a pool of possible RULs. Figure 5.7 represents the training trajectories, whose distance to the test trajectory 13 is the lowest.

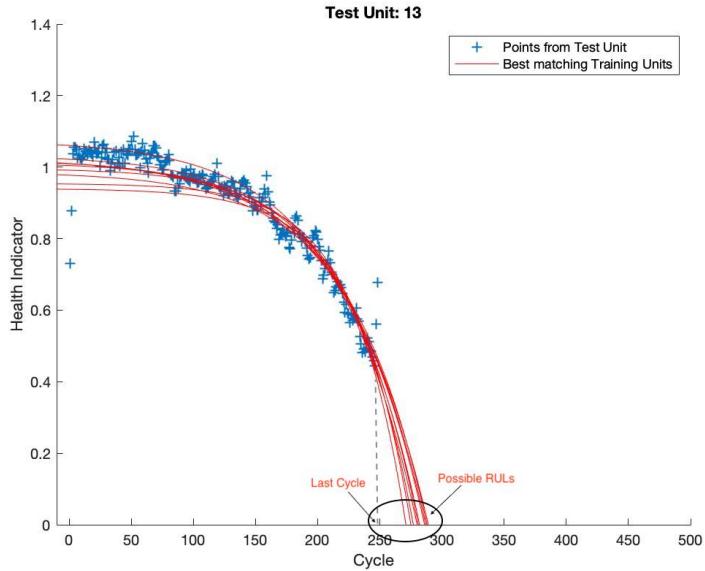


Figure 5.7: Assessment of the most similar training trajectories with the test trajectory 13

Finally for the RUL determination, a weighted average is performed, based on the pool of RULs:

$$RUL = (13/23).min_i(RUL_i) + (10/23).max_i(RUL_i) \quad (5.4)$$

This formulation, suggested by the author, aims to give more importance and weight to the earlier predictions rather than the later predictions.

For a more detailed explanation regarding the approach followed, please access Appendix B.1.

Neural Network Based Approach and Extrapolation Based Approach:

In both the **Neural Network-based** and **Extrapolation-based** approaches, the RUL is computed in an identical way. The RUL is obtained by performing an extrapolation of HI values computed in the previous step, until the HI value reaches the value of 0. When the HI value is 0, it represents the total inoperability of the system due to the failure evolution. Thus, the RUL value corresponds to the difference between the cycle where the HI value is 0 and the last cycle presented on that trajectory history.

Figure 5.8 and Figure 5.9 illustrates the extrapolation of the HI values of the test trajectory 13, computed using the **Neural Network-based** and **Extrapolation-based** approaches, respectively. Also, in these Figures, the RUL values obtained are indicated.

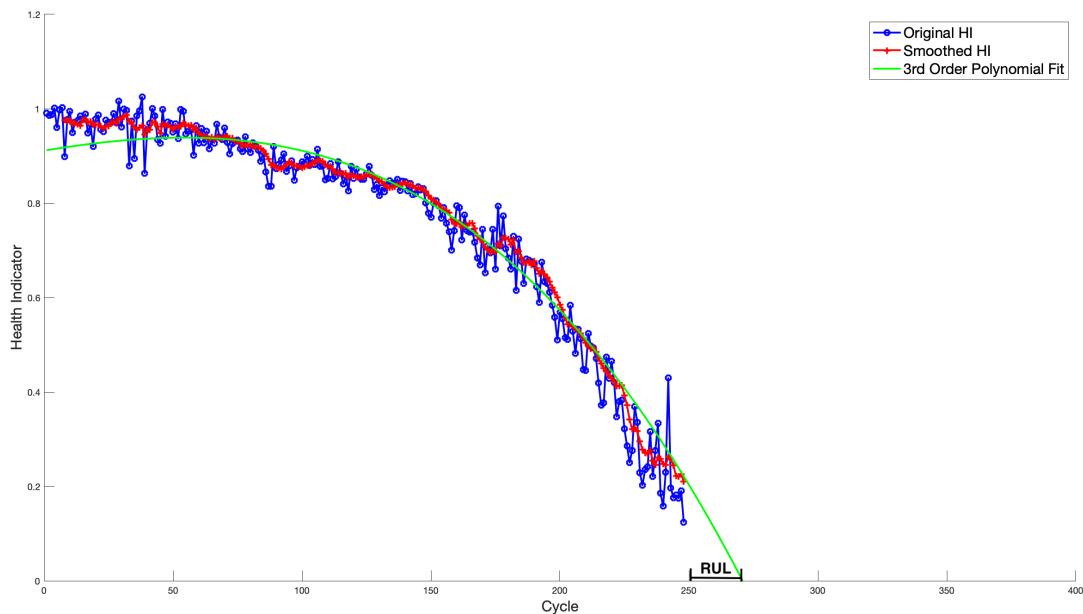


Figure 5.8: RUL estimation using a NN approach for Test Unit 13

As observed, although the extrapolation method is the same, due to the different HI values of each approach, the RULs obtained were also different.

For a more detailed explanation regarding the **Neural Network Based Approach**, please access Appendix B.2.

The description, regarding the **Extrapolation Based Approach** is present in Appendix B.3.

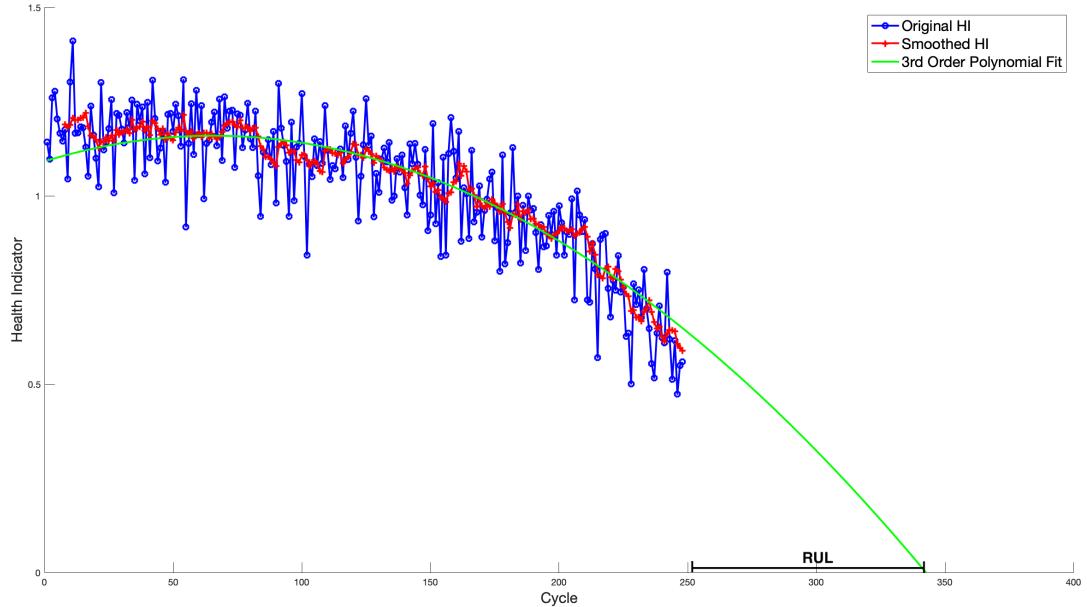


Figure 5.9: RUL estimation using an Extrapolation approach for the Test Unit 13

5.1.6 Results obtained

The Table 5.4 compares the different approaches, in terms of Root Mean Squared Error (RMSE). As the ground truth RULs for the PHM08 Challenge Dataset were not provided, the RMSE value was not computed for that dataset.

Table 5.4: RMSE Error

| | Turbofan Dataset | | | | PHM08 Challenge Dataset |
|------------------------|------------------|-------|-------|-------|-------------------------|
| | FD001 | FD002 | FD003 | FD004 | |
| Similarity Approach | 19.87 | 22.65 | 21.40 | 22.75 | - |
| NN Approach | 25.63 | 31.16 | 26.58 | 32.92 | - |
| Extrapolation Approach | 25.15 | 38.77 | 55.18 | 43.85 | - |

The Table 5.5 compares the different approaches, in terms of Score Error. This metric was created by the PHM08 Challenge Competition. The results achieved for the PHM08 Challenge Dataset were obtained through the submission of the results in the link provided by the Competition for the evaluation of the results.

The Score Error does not have a physical or particular meaning, it is simply a value to be used in order to compare the different approaches performance.

Table 5.5: Score Error

| | Turbofan Dataset | | | | PHM08 Challenge Dataset |
|------------------------|------------------|----------|-----------|----------|-------------------------|
| | FD001 | FD002 | FD003 | FD004 | |
| Similarity Approach | 559.08 | 3979.52 | 1805.32 | 2813.10 | 1036.56 |
| NN Approach | 701.29 | 18387.85 | 5215.90 | 82043.89 | 9628.34 |
| Extrapolation Approach | 1477.37 | 1.166e+6 | 3.557e+12 | 2.06e+06 | 32527.17 |

The analysis of the results of RMSE and Score Error presented in tables 5.4 and 5.5, allows one to conclude that the **Similarity-based approach** is the method that more accurately estimates the RUL for the testing dataset.

The second best approach is **Neural Network-based**, and the approach with the least accurate results is the **Extrapolation-based approach**.

In terms of dataset, the FD001 has better results. This was already expected as the data from this dataset was generated with just one failure embedded and only one operational regime in use. These facts simplified the detection of the failure and the prediction of the respective RUL.

The **Similarity-based approach** provided good results regarding the RUL prediction, which was expected, given that the approach followed was the one with the best score in the PHM 2008 Challenge Competition.

The results of the **Neural Network-based approach** were below the expected. Some reasons for not obtaining better results might be an important preprocessing step disregarded or not well executed, or a wrong decision regarding the choice of the order of the regression that better described the degradation behavior of the test instances.

Other possible reasons are the poor definition of the MLP architecture, namely the number of neurons and layers and the prognostic method used, in this case an extrapolation.

The **Extrapolation-based approach** also produced poor results. This was expected due to the fact that the RUL calculation was based on a simple regression. In particular, this type of approach can drastically weaken results when the dataset size is small, which was the case of some of the aircraft trajectories presented in the dataset.

This way, the use of an extrapolation for predicting the RUL value in the **Neural Network-based** and **Extrapolation-based approaches**, may not be the best option. When the degradation is represented by irregular trends and the quantity of data is low, generally, the extrapolation technique is not able to represent correctly the degradation behavior over time. This should be one of the main causes for the results obtained in both approaches.

Overall, it was expected that both the **Similarity-based** and **Neural Network-based** approaches would yield better results due to the techniques used for the HI estimation and RUL prediction. The **Extrapolation-based approach** was expected to provide worse results due to the simple methodology assumed for RUL prediction.

Comparing the Score error obtained for the PHM08 Challenge Dataset with the scores achieved by the authors, the **Similarity approach** was relatively closer to the expected, as the authors score was 512.12 and the score obtained in this work was 1036.56. The reasons for this difference might be the way the moving average was applied, including the considered window length, or the definition of the r value in the equation B.4, which influences the overall RUL calculation.

Nevertheless, the **Similarity-based approach** obtained a score of 1036.56, which, according to the benchmark performed in 2014, is within the best 5 results obtained using this dataset.

Figure 5.10 represents the best 10 results obtained using the PHM08 Challenge Dataset.

Although the data from the NASA's dataset is synthetic and simplistic compared to a real case scenario, it is valuable for experimenting and exploring different methods for the HI and RUL estimation.

| Rank on #5T | Competition Score | MSE | FPR (%) | FNR (%) | MAPE (%) | MAE | Corr. Score | Std. Dev. | MAD | MdAD |
|-------------|-------------------|--------|---------|---------|----------|-------|-------------|-----------|-------|-------|
| 1 | 512.12 | 152.71 | 56.35 | 43.65 | 15.81 | 8.67 | 0.96 | 0.64 | 8.68 | 5.69 |
| 2 | 740.31 | 224.79 | 57.73 | 38.12 | 18.92 | 10.77 | 0.94 | 0.76 | 10.72 | 7.00 |
| 3 | 873.36 | 265.62 | 53.59 | 28.18 | 19.19 | 11.47 | 0.93 | 0.81 | 11.75 | 8.00 |
| 4 | 1218.43 | 269.68 | 51.93 | 44.20 | 20.15 | 11.87 | 0.93 | 0.85 | 12.05 | 8.00 |
| 5 | 1218.76 | 331.30 | 50.55 | 49.45 | 33.14 | 13.81 | 0.91 | 0.95 | 14.03 | 10.87 |
| 6 | 1232.27 | 334.52 | 42.27 | 57.73 | 32.90 | 14.14 | 0.91 | 0.96 | 14.28 | 10.37 |
| 7 | 1568.98 | 394.46 | 50.55 | 47.24 | 36.75 | 15.37 | 0.89 | 1.03 | 15.48 | 12.00 |
| 8 | 1645.77 | 330.02 | 47.24 | 50.28 | 30.00 | 13.47 | 0.91 | 0.95 | 13.59 | 10.00 |
| 9 | 1816.60 | 359.97 | 50.28 | 49.17 | 26.47 | 13.82 | 0.90 | 0.99 | 14.07 | 9.75 |
| 10 | 1839.06 | 377.01 | 45.86 | 53.59 | 27.72 | 14.31 | 0.89 | 1.02 | 14.43 | 9.10 |

Figure 5.10: Table with the best 10 results until 2014 (from *Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets* [163])

5.2 Brakes System

One of the first datasets made available was the Brakes dataset. This dataset was considered a simple case scenario for the RUL estimation, due to its smaller size and low complexity of the brakes system, which made the understanding of the system and the interpretation the dataset easier. Hence, it was used as a case study, where some preliminary work was performed, with the goal of predicting the RUL based on the brakes sensors data.

Similarly to the previous case study, the next sections describe the approach for computing the RUL, in alignment with the pipeline proposed in Chapter 4, illustrated in Figure 4.1.

5.2.1 Dataset Structure

The brakes dataset contains historical data from a specific time range retrieved from the brakes sensors localized in the landing gear of 13 Boeing 787 airplanes. In addition, data regarding the performed removals (parts replacement) in the airplanes is also provided. Thus, two different csv's files were provided, the **Brakes sensors data** csv and the **Removals data** csv.

Brakes sensors data:

This csv file contains the historical data retrieved from 8 different positions of the brakes of different airplanes. Figure 5.11 illustrates a sample of this dataset.

As observed the dataset contains the following fields:

- **Tail No:** Anonymized identifier of the airplane
- **Leg Date:** Date of the flight
- **Departure:** Anonymized identifier of the departure location
- **Arrival:** Anonymized identifier of the arrival location
- **Report Date:** Date of when the data was reported
- **1 - 8:** Health percentage of the 8 brakes positions.

| Tail No | Leg Number | Departure Date | Departure Time | Arrival | Report Date | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-----------|------------------|----------------|----------------|--------------|-------------|----|----|----|----|----|----|---|---|
| 1 PH-013 | 03/01/2019 22:41 | TTT | AGH | 04-Jan-19 14 | 45 | 37 | 97 | 43 | 43 | 48 | 35 | | |
| 2 PH-013 | 03/01/2019 11:54 | AGH | TTT | 03-Jan-19 14 | 45 | 37 | 97 | 43 | 43 | 46 | 36 | | |
| 3 PH-013 | 02/01/2019 23:37 | TRH | AGH | 03-Jan-19 14 | 45 | 37 | 97 | 44 | 43 | 48 | 36 | | |
| 4 PH-013 | 02/01/2019 21:04 | FDH | TRH | 02-Jan-19 14 | 45 | 37 | 97 | 44 | 43 | 49 | 36 | | |
| 5 PH-013 | 02/01/2019 09:53 | AGH | FDH | 02-Jan-19 14 | 45 | 37 | 97 | 44 | 43 | 49 | 36 | | |
| 6 PH-013 | 01/01/2019 03:53 | AGH | AGH | 01-Jan-19 14 | 45 | 37 | 97 | 44 | 44 | 49 | 36 | | |
| 7 PH-013 | 31/12/2018 03:53 | AGH | AGH | 01-Jan-19 14 | 45 | 37 | 97 | 44 | 44 | 49 | 36 | | |
| 8 PH-013 | 31/12/2018 16:23 | AGH | FJU | 01-Jan-19 14 | 45 | 37 | 97 | 44 | 43 | 49 | 36 | | |
| 9 PH-013 | 31/12/2018 03:25 | GHI | AGH | 31-Dec-18 14 | 45 | 38 | 97 | 44 | 44 | 49 | 36 | | |
| 10 PH-013 | 30/12/2018 15:16 | AGH | GHI | 31-Dec-18 14 | 45 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 11 PH-013 | 30/12/2018 01:50 | JUC | AGH | 30-Dec-18 15 | 45 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 12 PH-013 | 29/12/2018 16:25 | AGH | JUC | 30-Dec-18 15 | 45 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 13 PH-013 | 28/12/2018 23:45 | HUH | AGH | 29-Dec-18 15 | 45 | 39 | 98 | 44 | 44 | 49 | 36 | | |
| 14 PH-013 | 28/12/2018 09:53 | AGH | HUH | 28-Dec-18 15 | 45 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 15 PH-013 | 27/12/2018 22:49 | UAB | AGH | 28-Dec-18 15 | 46 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 16 PH-013 | 27/12/2018 13:24 | AGH | UAB | 27-Dec-18 15 | 46 | 38 | 98 | 44 | 44 | 49 | 36 | | |
| 17 PH-013 | 27/12/2018 13:24 | AGH | GHI | 27-Dec-18 15 | 46 | 38 | 98 | 44 | 44 | 49 | 37 | | |
| 18 PH-013 | 27/12/2018 15:59 | AGH | GHI | 27-Dec-18 15 | 46 | 38 | 98 | 45 | 44 | 49 | 37 | | |
| 19 PH-013 | 24/12/2018 07:31 | TBL | AGH | 24-Dec-18 15 | 46 | 38 | 98 | 45 | 45 | 50 | 37 | | |
| 20 PH-013 | 23/12/2018 20:04 | AGH | TRL | 24-Dec-18 15 | 46 | 38 | 98 | 45 | 45 | 50 | 37 | | |
| 21 PH-013 | 23/12/2018 05:18 | CHC | AGH | 23-Dec-18 16 | 46 | 39 | 98 | 45 | 45 | 50 | 37 | | |
| 22 PH-013 | 22/12/2018 16:25 | AGH | CHC | 23-Dec-18 15 | 46 | 38 | 98 | 45 | 45 | 50 | 37 | | |
| 23 PH-013 | 21/12/2018 01:50 | JUC | AGH | 22-Dec-18 15 | 46 | 39 | 98 | 45 | 45 | 50 | 37 | | |
| 24 PH-013 | 21/12/2018 13:48 | AGH | JUC | 22-Dec-18 15 | 46 | 39 | 98 | 45 | 45 | 50 | 37 | | |
| 25 PH-013 | 21/12/2018 02:41 | JPU | AGH | 21-Dec-18 16 | 46 | 39 | 98 | 45 | 45 | 50 | 37 | | |
| 26 PH-013 | 20/12/2018 16:07 | AGH | JPU | 21-Dec-18 16 | 46 | 39 | 98 | 45 | 45 | 50 | 37 | | |
| 27 PH-013 | 19/12/2018 13:55 | JUC | AGH | 20-Dec-18 16 | 46 | 39 | 99 | 45 | 45 | 50 | 38 | | |
| 28 PH-013 | 19/12/2018 00:29 | HUH | AGH | 19-Dec-18 16 | 46 | 39 | 99 | 45 | 45 | 50 | 38 | | |
| 29 PH-013 | 19/12/2018 00:29 | HUH | AGH | 19-Dec-18 16 | 46 | 39 | 99 | 45 | 45 | 50 | 38 | | |
| 30 PH-013 | 19/12/2018 00:29 | HUH | AGH | 19-Dec-18 16 | 46 | 39 | 99 | 45 | 45 | 50 | 38 | | |

Figure 5.11: Brakes dataset sample

In this dataset, each row corresponds to the information regarding one flight, thus the sensors values obtained during the flights were not provided, only the sensors values at the end of the flight. As a consequence, the dataset size is significantly reduced.

Removals data:

Besides the sensors data, data regarding the performed removals was also made available in the **Removals data** csv file. Each row of this dataset contains a removal that was performed due to a failure or degradation in the brakes positions.

This type of information is particularly important, as it allows the computation of degradation patterns that are used, as a base, for making future predictions regarding the brakes degradation and thus the computation of the RUL.

Figure 5.12 shows a sample of the removals dataset.

| /Users/danielazevedo/Desktop/dados_KLM_breaks/RemovalData_share.csv | | | | | | | | | | |
|---|--------|--------|--------|----------|----------|--------|---------|--|--|--|
| Rem/Inst Date | Weekno | IN/OUT | AC Reg | Position | Rem.Code | ATA | Airline | | | |
| 2 21-Sep-17 | 38 | OUT | PH013 | 1 | 3 | 324501 | KLM | | | |
| 3 25-Oct-17 | 43 | OUT | PH013 | 2 | 3 | 324501 | KLM | | | |
| 4 20-Sep-17 | 38 | OUT | PH013 | 3 | 3 | 324501 | KLM | | | |
| 5 2-May-17 | 18 | OUT | PH013 | 4 | 3 | 324501 | KLM | | | |
| 6 14-Dec-18 | 50 | OUT | PH013 | 4 | 3 | 324501 | KLM | | | |
| 7 25-Oct-17 | 43 | OUT | PH013 | 5 | 3 | 324501 | KLM | | | |
| 8 2-Oct-17 | 40 | OUT | PH013 | 6 | 3 | 324501 | KLM | | | |
| 9 6-Nov-17 | 45 | OUT | PH013 | 7 | 3 | 324501 | KLM | | | |
| 10 16-Sep-17 | 37 | OUT | PH013 | 8 | 3 | 324501 | KLM | | | |
| 11 5-Sep-17 | 36 | OUT | PH004 | 1 | 3 | 324501 | KLM | | | |
| 12 7-Sep-17 | 36 | OUT | PH004 | 2 | 3 | 324501 | KLM | | | |
| 13 7-Sep-18 | 36 | OUT | PH004 | 2 | 2 | 324501 | KLM | | | |
| 14 5-Sep-17 | 36 | OUT | PH004 | 3 | 3 | 324501 | KLM | | | |
| 15 11-Oct-17 | 41 | OUT | PH004 | 4 | 3 | 324501 | KLM | | | |
| 16 31-Aug-17 | 35 | OUT | PH004 | 5 | 3 | 324501 | KLM | | | |
| 17 18-Sep-17 | 38 | OUT | PH004 | 6 | 3 | 324501 | KLM | | | |
| 18 5-Dec-18 | 49 | OUT | PH004 | 6 | 3 | 324501 | KLM | | | |
| 19 28-Oct-17 | 43 | OUT | PH004 | 7 | 3 | 324501 | KLM | | | |
| 20 8-Jan-18 | 2 | OUT | PH004 | 7 | 3 | 324501 | KLM | | | |
| 21 24-Oct-17 | 43 | OUT | PH004 | 8 | 3 | 324501 | KLM | | | |
| 22 8-Jan-18 | 2 | OUT | PH003 | 1 | 1 | 324501 | KLM | | | |
| 23 23-Jan-18 | 4 | OUT | PH003 | 2 | 3 | 324501 | KLM | | | |
| 24 25-Jun-18 | 27 | OUT | PH003 | 2 | 3 | 324501 | KLM | | | |
| 25 18-Apr-18 | 16 | OUT | PH003 | 3 | 3 | 324501 | KLM | | | |
| 26 9-Mar-18 | 10 | OUT | PH003 | 4 | 3 | 324501 | KLM | | | |

Figure 5.12: Removals dataset sample

As represented in Figure 5.12, the dataset contains the following fields:

- **Rem/Inst Date:** Date of the removal
- **Weekno:** Number of the week where the removal was performed
- **IN/OUT:** Type of the removal: ‘IN’ when the removal corresponds to the introduction of a new part; ‘OUT’ when a wastage part is removed from the airplane
- **AC Reg:** Anonymized identifier of the airplane
- **Position:** Brake position that is being intervened
- **Rem-Code:** Code for the removal motive
- **ATA:** ATA Code of the component that is being intervened
- **Airline:** Airline company responsible for the removal

5.2.2 Preprocessing of Brakes data

As mentioned before, the brakes dataset corresponds to a simple case scenario, where the dataset already provides the essential information needed, in a compact and clear organization.

Also, the sensors data provided regards the end of the flight, thus, no real time data from the sensors is provided. In addition, as the dataset features were manipulated, that is, they do not correspond to the raw values retrieved from the sensors, the existence of noisy data is low.

Due to these reasons, no extra preprocessing was performed on this dataset, regarding the filtering of possible noisy data.

In relation to the correct flight categorization, as each line corresponds to a different flight, the information of each flight is well identified.

5.2.3 Feature Selection

In this dataset, there was no need to perform feature selection due to the features relevancy for the diagnostic and prediction of the system health condition.

As each feature reflects the health percentage of a different brake position, and each brake position has equal relevancy for the system condition diagnosis, all the features were considered to be valuable and relevant for the computation of the HI and, consequently, the RUL.

5.2.4 HI computation

Regarding the HI computation, it was not necessary to apply any specific method, as the features already provided this information. As referred before, each feature indicates the health percentage of a different brake position, hence, these values correspond to the HI.

As such, 100% means that the position is fully healthy, and 0% means that it needs to be replaced due to failure or degradation.

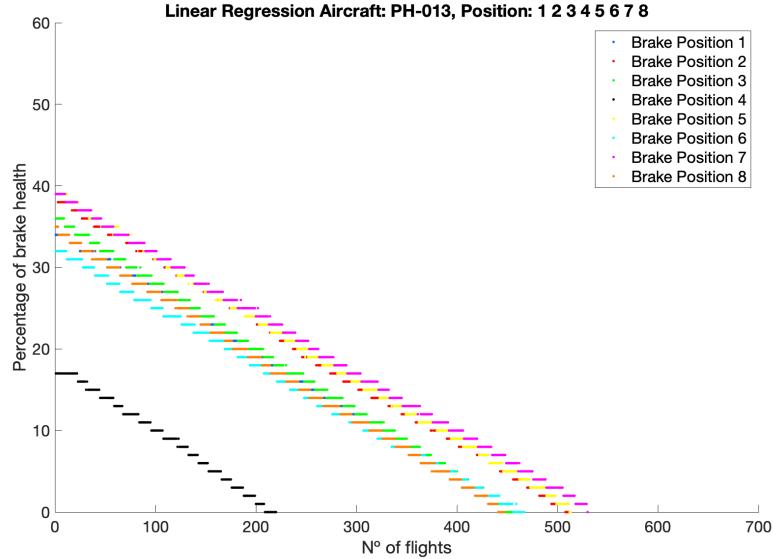


Figure 5.13: Degradation trajectories of the 8 brakes positions for aircraft PH-013

Figure 5.13 reflects the evolution of the degradation of the 8 brakes' positions for aircraft PH-013. As can be concluded, the HI values are decreasing over time, until they reach values closer to 0. This is when the removals are performed.

5.2.5 RUL prediction

For the brakes dataset, a simple method for the RUL computation is developed, based on a Linear Regression.

Using the degradation trajectory, a Linear Regression is applied in order to model the degradation evolution over time. Furthermore, using the obtained regression it is possible to predict the future degradation behavior and consequently, compute the RUL value.

Then, using the RUL information, the estimated removal time is compared to the real removal instant in order to conclude whether the removal should have occurred earlier or later than the actual removal.

Figure 5.14 shows an example of the prediction of the removal instant for a certain degradation trajectory of aircraft PH-013. Using this Figure it is possible to perform a comparison between the predicted and real removal time for the degradation trajectory.

The points represented in the Figure correspond to the health percentage of a particular degradation trajectory regarding the brake position 1 of the aircraft and were extracted directly from the dataset. As indicated, the dark blue points represent the diagnostic points, and are used to fit the linear regression. The light blue points represent the ground truth, these are used to evaluate if the linear regression accurately reproduces or not the degradation evolution. As illustrated, the linear extrapolation correctly represents the health percentage evolution. It is also possible to observe that the real removal time was when the HI was about 7%.

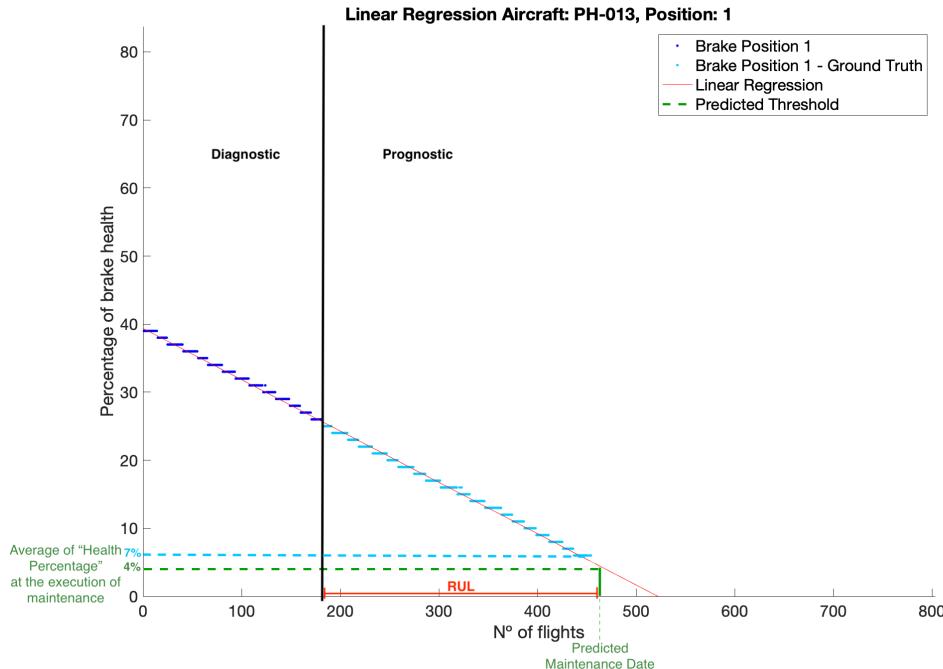


Figure 5.14: RUL prognostic

The predicted removal time was calculated by observing the number of flights remaining when the HI was 4%. The 4% corresponds to the average of the HI at the removal instant of all brakes degradation trajectories in the dataset.

This way, the RUL corresponds to the number of flights since the end of the diagnostic phase to the predicted removal time.

As observed in the Figure, the predicted removal time was a few flights after the actual replacement, which means that, in this particular case scenario and according to the prediction, the brake position 1 could have operated a few more flights. As such, the RUL estimation was close to the ground truth, which confirms that a linear regression is useful for modeling the degradation behavior of the brake positions.

Using this simple approach, it was possible to create a model for representing the brakes degradation and then predict its future behavior in order to compute the RUL.

Nevertheless, this approach can still be significantly improved. Contrary to what appears in Figure 5.14, the use of a Linear Regression to represent the brakes degradation is not the best choice, as, according to the State of the Art, generally, the components degradation do not follow a linear behaviour, but a non linear or exponential behaviour [17]. Also, no data preprocessing (like scaling, feature reduction or outlier detection) was deemed valuable, nevertheless, it may help improving the accuracy of the obtained results. Lastly, the health condition provided in the dataset features was computed based on the necessity of replacement and not based on the real health condition. This means that, for instance, the 0% reflects that the brake needs replacement very soon and not that it is inoperable due to failure or degradation. Thus, the formulation and meaning of the health condition (dataset features) also contributes for the discrepancy in the results.

Chapter 6

Diagnosis and Prognosis on the Air Bleed System

After performing an experimental work with the Brakes dataset and Turbofan dataset, the work focused on the analysis of the Air Bleed system of Boeing 747.

The Air Bleed dataset is one of the 13 systems contemplated in the ReMAP proposal, as such, the data was provided within the ReMAP context.

Due to the complex nature of the Air Bleed system and its respective data, the work developed in this dataset corresponds to the main contribution of this Thesis.

The Air Bleed system is one of the important systems present in the airplanes. In simple terms, the bleed system is responsible for processing the high pressure and temperature of the air that is bled from the turbofan engine and conduct it, through valves and regulators, to other parts of the aircraft where that air is needed. Some common operations where the processed bleed air is used are: cabin pressurization, air conditioning, engines anti-icing, etc [169].

Based on the data provided regarding the Boeing 747 Air Bleed system, the goal is to diagnose the HI of the system over time, which is influenced by possible degradation, wastage or failures. Using the diagnostic component, predictions regarding the future degradation are made, which result in the RUL computation.

In the following Sections, the PHM approach developed for the diagnosis and prognosis of the Air Bleed system health condition is explained, following the established pipeline, represented in Figure 4.1.

6.1 Air Bleed data analytics

In this Section the study and description of the Air Bleed data is performed, as well as, the analysis of the data quality and relevancy to this work. In addition, the techniques of preprocessing and feature selection applied in the data are explained next.

All the techniques implemented and the respective results were generated using the Python language. Furthermore, the Jupyter notebook [170] was used for a more easy and clear analysis of the provided data.

6.1.1 Dataset structure

The Air Bleed dataset provided, contains historical data from 20 Boeing 747 airplanes.

Besides the data retrieved from the aircraft sensors, the information regarding the generated Flight Deck Events (FDE) and the removals that were executed in these specific airplanes, was also made available.

Thus, three different datasets were provided, each with different type of information:

- **Sensors data dataset:** This dataset contains raw data extracted directly from a set of sensors localized in the Air Bleed system.
- **FDE dataset:** This dataset contains information regarding the FDE, that were automatically triggered by the system during the flights.
- **Removals dataset:** This dataset contains information regarding the removals (parts replacement) that were performed by the maintenance team, due to inability to operate normally.

Sensor data Dataset

The sensors historical data was provided in 20 csv files. Each csv file incorporates data from a different aircraft. In total, historical data from 20 Boeing 747 airplanes was provided.

For each aircraft, there is data retrieved from 4 different Air Bleed systems of the plane, each with 5 sensors whose data was provided in 5 anonymized features. The data timestamp ranges between 2015 and 2017 and the sampling rate is 1 second. As such each csv file, specific of a different aircraft, has a size of about 4 GB.

All the 20 csv files have the same structure and organization. Each csv file has 30 different fields (columns), which, due to privacy issues, are anonymized. The csv columns are the following:

- **H51394154:** Index value of the each flight, which corresponds to a counter that is reset in the beginning of each flight.
- **Timestamp:** Datetime of each data row, as the name suggests.
- **H20790882:** Flight phase of each data row. In the Boeing 747, from where the data is retrieved, there are 14 different flight phases, which have the following meaning:

| | |
|------------------|---------------------|
| 1. Power on | 8. Cruise |
| 2. Engine start | 9. Descent |
| 3. Taxi out | 10. Approach |
| 4. Unknown | 11. Rollout |
| 5. Take-off roll | 12. Taxi in |
| 6. Initial climb | 13. Unknown |
| 7. Climb | 14. Engine shutdown |
- **H90548828 - H68446583:** This set of columns contains the data from the 5 sensors, regarding the 4 different systems. The full description of their meaning is presented in Table 6.1

- **H13768180:** File date, that is, the date of the creation of file containing the data extracted from the sensors. This information is not used, as it is not relevant for this work.
- **H62778170:** Tail Number, that is the plane ID. As each file contains data from one specific plane, this column values are constant through each file.
- **H64936356:** Identifier of the file were the data was recorded. This information is not used, as it is not relevant for this work.
- **H73670103:** Time values. Its meaning is not clear for us, neither to the data provider. Also, there are significant number of Nan's and thus the data regarding this column was no used.
- **H22124930:** Date values. Similarly to the previous column, this column data was also not used due to uncertainty regarding its meaning and meaningful number of Nan's.
- **H55759866:** Flight number (flight id). Nevertheless, there are significant number of missing values.
- **custom_id:** Flight number (flight id), that resulted from an initial data pre-processing performed by other ReMAP partner.

Table 6.1: Sensors Description

| Column Name | Sensor | Air Bleed System |
|-------------|-----------------|------------------|
| H90548828 | Shaft 1 | 1 |
| H38450254 | Shaft 1 | 2 |
| H32780567 | Shaft 1 | 3 |
| H35950744 | Shaft 1 | 4 |
| H67308013 | Shaft 2 | 1 |
| H81690022 | Shaft 2 | 2 |
| H97597396 | Shaft 2 | 3 |
| H24264101 | Shaft 2 | 4 |
| H45316075 | Oil Temperature | 1 |
| H05924399 | Oil Temperature | 2 |
| H69613721 | Oil Temperature | 3 |
| H94813608 | Oil Temperature | 4 |
| H53329349 | Air Pressure | 1 |
| H53294016 | Air Pressure | 2 |
| H65502179 | Air Pressure | 3 |
| H92021026 | Air Pressure | 4 |
| H65983355 | Air Temperature | 1 |
| H83395790 | Air Temperature | 2 |
| H34527078 | Air Temperature | 3 |
| H68446583 | Air Temperature | 4 |

FDE file

Regarding the data provided from the Air Bleed system, the information of the FDE that were triggered during flight is also provided. As mentioned before, the FDE correspond to automatic messages generated by the system due to some problem or anomaly detected. These messages have considerable importance for this work, as they might be an indicator of a possible failure, thus, this information is valuable for the diagnosis and prediction of the HI of the system, and its validation.

The FDEs that occurred in 15 different airplanes were provided in a separate csv file. The csv file has the following structure:

- **Message Code:** Identifier of the problem detected in the Air Bleed system.
- **Message Text:** Text that describes the detected problem. It is associated with a specific Message Code. Moreover, in the text, it also indicates the number/identifier of bleed system of the aircraft where the FDE was triggered.
- **Tail#:** Anonymized tail number of the aircraft where the the problem was detected.
- **Leg Date:** Datetime when the identified problem was detected.
- **Flight Phase:** Flight phase when the problem was detected. The coding system used is the following:
 - **AL:** Approach Land
 - **AP:** Approach
 - **CL:** Climb
 - **DC:** Descent
 - **ER:** Enroute Cruise
 - **ES:** Engine Start
 - **FL:** Flare
 - **GA:** Go Around
 - **IC:** Initial Climb
 - **LT:** Leg Transition
 - **MT:** Maintenance
 - **PF:** Preflight
 - **PO:** Power On
 - **RO:** Rollout
 - **SD:** Engine Shutdown
 - **TA:** Taxi-Out
 - **TI:** - Taxi- in
 - **TO:** - Take off
- **ATA Code:** ATA code of the part of the system where the problem was detected.
- **Fault Code:** Code of the identified fault. In comparison to the first column (Message Code), the Fault Code may be more specific regarding the anomaly detected.
- **Fault Text:** Fault description. Each Fault Code has a particular Fault Text.
- **Fault Report Code - Monitor Code:** Report of the maintenance operations. Nevertheless, this column is always empty and so it does not add any value to the annotations or this work.
- **Priority:** Priority of each FDE, specified in the following levels.
 - **High:** Needs immediate analysis and corrective action. The system condition either:
 1. Contains non-dispatch relief item
 2. Has highest historical costs

3. Significantly affects operations
- **Medium:** Needs near-term analysis and possibly corrective action. The system condition either:
 1. Limits dispatch relief
 2. Has an impact on operations
 3. High cost
- **Low:** Needs routine analysis and possibly corrective actions. It represents abnormal behavior, that may either:
 1. Limit dispatch relief
 2. Have an impact on operations
 3. Have the potential to escalate to a high-cost FDE

Removals file

The information regarding the parts replacement performed in the aircraft by the maintenance team is provided in a separate dataset.

The removals presented in this dataset are from 22 different airplanes. This dataset with the removals information, is structured in the following way:

- **Name:** Aircraft part that is replaced in the removal.
- **Rem/Inst Date:** Datetime of the removal action.
- **IN/OUT:** Type of the removal performed: 'IN' when the removal corresponds to the introduction of a new part; 'OUT' when a wastage part is removed from the airplane.
- **AC Reg:** Anonymized tail number of the airplane where the removal was executed.
- **Position:** Number/Identifier of bleed system id where the removal was performed.
- **Rem.Code:** Reason for the removal, expressed in a coding system. The coding system used is:
 - 1: Removal performed due to reaching a time-limit (scheduled maintenance).
 - 2: Removal performed due to a cross-exchange.
 - 3: Removal performed due to a failure detected (unscheduled removal).
 - 4: Removal performed due to a shop check.
- **TSI Hours:** Number of hours of flight since the previous removal performed on the same component.
- **TSI Cycles:** Number of cycles of flight since the previous removal performed on the same component.
- **ATA:** ATA code of the part of the system where the removal was executed.
- **Airline:** Airline responsible for the data.
- **FF:** Code of the health condition of the removed aircraft part or component.

The coding system used is the following:

- Y/J: Fault found in the aircraft part.
- N: No fault found in the aircraft part.
- O: No information regarding the presence of faults in the aircraft part.

Relation and Mapping of the 3 sources of information

By the description above of the different datasets of data, it is perceptible that complementary information is provided by the 3 sources of data. This way, the correct mapping between the different sources of information is useful in order to combine the different data with the objective of applying all the relevant information in this work.

As a result of the analysis regarding the structure of the 3 datasets, the following criteria were identified for the matching of the information between the different datasets:

- **Aircraft Tail number and number/identifier of the Bleed System:** As all the three datasets have information regarding the tail number and identifier of the bleed system involved, this criteria is significantly important to cross information between the different types of datasets.
- **Time:** The second criteria is the time. All the 3 datasets have information regarding the time when the different data was recorded or executed. This way, knowing already that it refers the same aircraft and same system, is possible to establish a timeline of the different events.

Using these criteria it is possible to successfully cross and map information from the 3 sources of information (sensors data, FDEs and removals) and establish an authentic timeline with the events occurred.

During the analysis of the different data provided, certain problems were found. These correspond to potential obstacles that, besides turning the data analysis more challenging and demanding, may also, have a negative impact on the algorithms performance. Different problems were found in the three files provided, particularly:

- Problems found in the sensors data:
 - Noise in the data.
 - Missing data regarding the flights.
 - Flights overlapping.
 - Flight Index was incorrect.
 - The meaning of some dataset fields was not clear.
 - Some flights contained too few hours, which means they shouldn't be considered.
- Problems found in the FDE:
 - Replacement time of each FDE is unknown, that is, the time between the FDE alert and the actual resolution is not known.
 - There may be some dependencies between different FDE messages. This relation is not clear, thus some expert knowledge regarding the system components behaviour is required.

- Problems found in the removals:
 - Uncertainty regarding the faulty behaviour of the reported removals i.e. not knowing if there was in fact a failure in the component replaced.
 - Absence of information regarding the health condition of the components used in the replacement.

Some of these problems were solved in the preprocessing stage while others were avoided, thereby having no impact on the analysis.

6.1.2 Preprocessing of Air Bleed data

Due to the different problems previously mentioned, some data preprocessing steps are performed. The goal of the preprocessing is to prepare the data for the application of the methodologies that diagnose and predict the system degradation over time. With that objective in mind, the most restrictive problems found were analyzed and handled in order to decrease their impact in the methodologies performance.

In order to perform a more accurate and correct analysis of the sensors data, the 14 phases of each flight in the dataset were combined in a set of 5 aggregated phases, that are grouped in the following way:

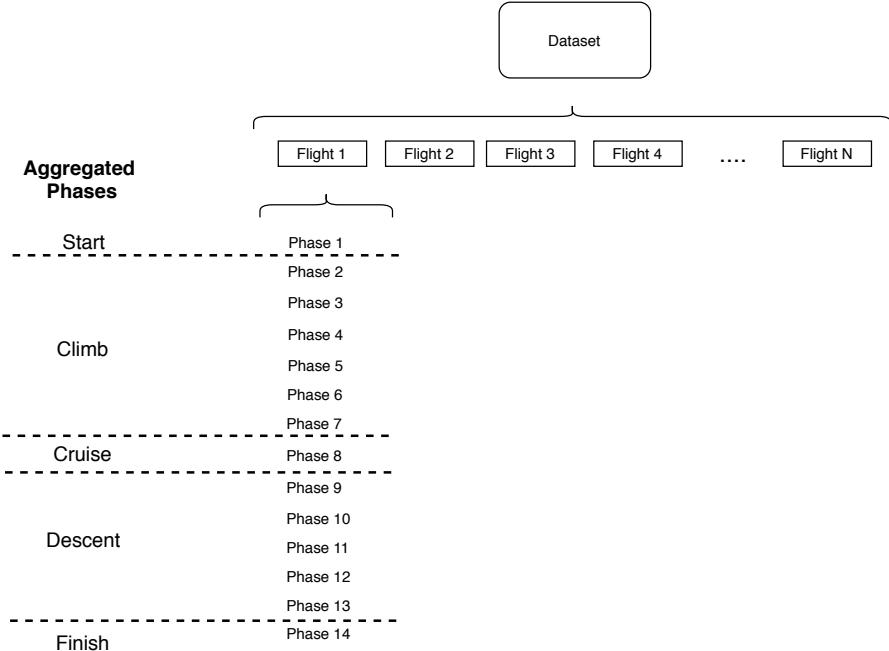


Figure 6.1: Aggregated phases Organization

The phases division was performed with the goal to correctly isolate the different data patterns presented in the sensors during one flight. Figure 6.2 exemplifies the data patterns isolation of the sensor H38450254 for a particular flight.

As observed, each aggregated phase embodies different data patterns. By analyzing the data by each aggregated phase, the analysis should be more correct and accurate as the data range and variance is restricted. Henceforth, all the analysis and preprocess applied to the data were made considering this aggregated phases division.

The performed data preprocessing, which is presented next, include the detection and adjustment of outliers and the correct identification and labeling of the different flights presented in the data.

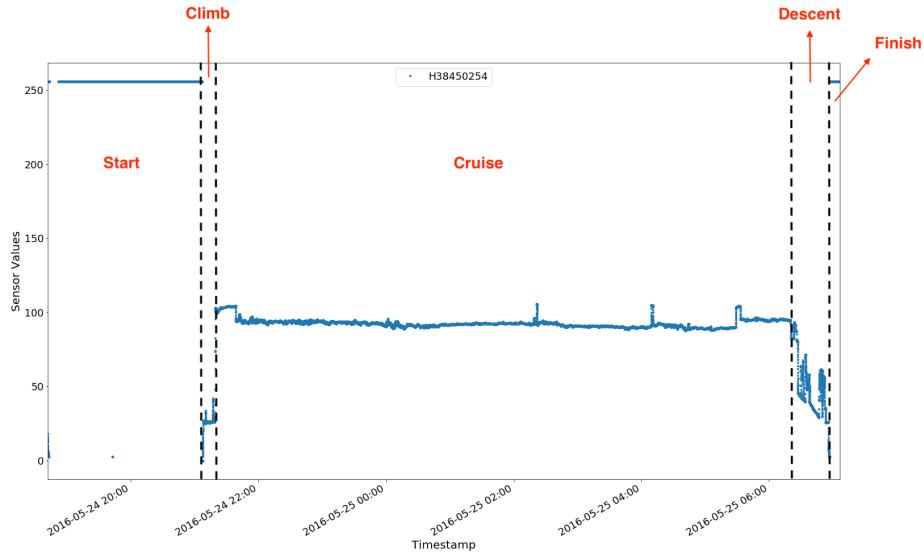


Figure 6.2: Aggregated phases division in one flight

Detection and accommodation of outliers

In order to reduce the noise embedded in the sensors data, a technique for detection and accommodation of outliers was developed. The algorithm is described in Section 4.2.1.

As there are clear data patterns in each sensor that are repeated in each flight, a sliding window approach was applied in order to detect outliers in each window based on the interval $[\mu - 3\sigma, \mu + 3\sigma]$, where the μ and σ represent the mean and standard deviation of the windowed values, respectively. If the sensor values are outside this interval they are considered an outlier.

The interval $[\mu - 3\sigma, \mu + 3\sigma]$ was defined for detecting the outliers, as, assuming a normal distribution of the sensors data, the established interval includes 99.73% of the values. Thus, it was determined that if the sensors values was outside this range it was labeled as an outlier and adjusted for the interval $[\mu - 2\sigma, \mu + 2\sigma]$.

The window length used was 20 minutes of data. The criteria for the choice was based on the criteria of trial and error and respective graphical analysis. Different window lengths were tested, like 5, 10 and 60 minutes, but from the graphical analysis it was observed that the window length of 20 minutes was the one which, visually, eliminates more of the irregular and out of bounds values.

Figures 6.3 and 6.4 illustrate the detection and adjustment of outliers for a specific time interval regarding Air Bleed system 2. Only the sensors H53294016 and H83395790 are presented as these are the ones with higher presence of outliers.

The green points in the Figures represent the detected outliers that are adjusted in the interval $[\mu - 2\sigma, \mu + 2\sigma]$.

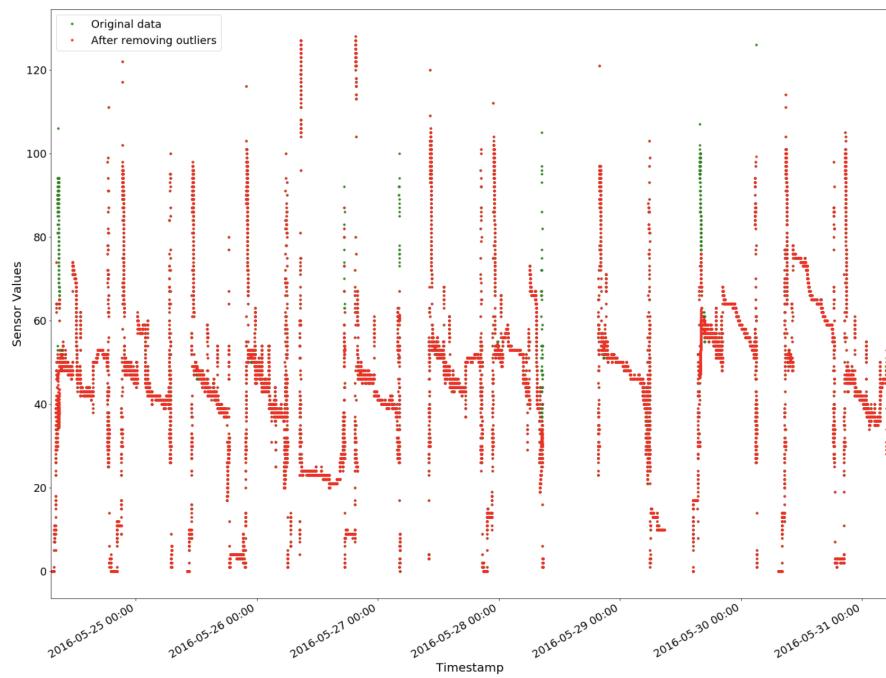


Figure 6.3: Outlier detection in sensor H53294016

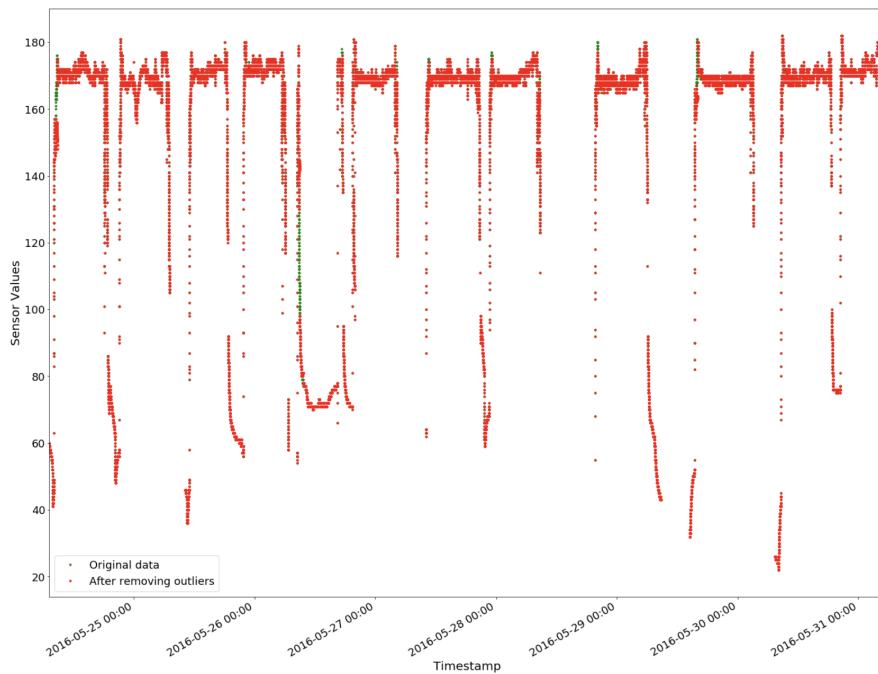


Figure 6.4: Outlier detection in sensor H83395790

In pseudo code, the implementation of the algorithm for the outliers detection and accommodation is illustrated in Figure 6.5:

```

Result: Outlier Detection and Accommodation
for each Sensor do
    for each Flight do
        for each Aggregated_Phase do
            window_size = 5 minutes
            #Definition of the initial sliding window
            s_window = first starting window of the considered interval
            #Definition of the last sliding window
            last_window = last window of the considered interval

            while s_window not last_window do
                μ = average of the values in the s_window
                σ = standard deviation of the values in the s_window
                next_value = first value out of the s_window

                if next_value > μ + 3σ :
                    #Outlier
                    # Data accommodation
                    next_value = μ + 2σ
                elif next_value < μ - 3σ :
                    #Outlier
                    # Data accommodation
                    next_value = μ - 2σ
                else:
                    #Not considered and outlier
                    Shift s_window one unit forward
            end
        end
    end

```

Figure 6.5: Pseudo code of outliers detection and accommodation algorithm

Identification and correct division of the different flights

Another important point in the application of the machine learning methodologies, is the correct labeling of the flights presented in the data, that is, the identification of the data corresponding to each flight.

With that goal in mind, column H55759866 was analysed to check its relevance for this point. By analysing this column data, it was realized that it contains a significant percentage of missing values, which prevents an accurate data flights division.

Another possibility found for the flights division, was using the column custom_id, created by another ReMAP partner. However, after performing a data analysis some problems in the correct identification of continuous flights were found.

Figure 6.6 illustrates the original flight phase values for a specific time interval, and Figure 6.7 illustrates the flight division performed by another ReMAP partner.

Although the isolated flights are correctly identified, continuous flights are labeled as only one flight, which does not correspond to the truth.

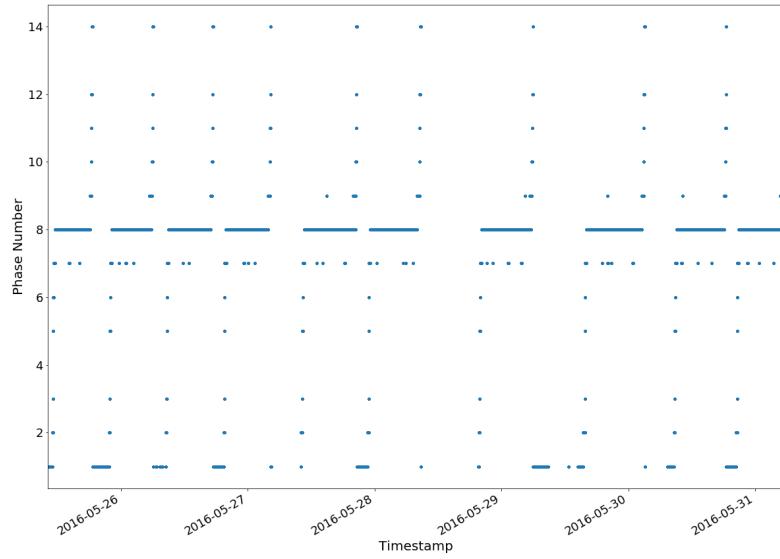


Figure 6.6: Original data

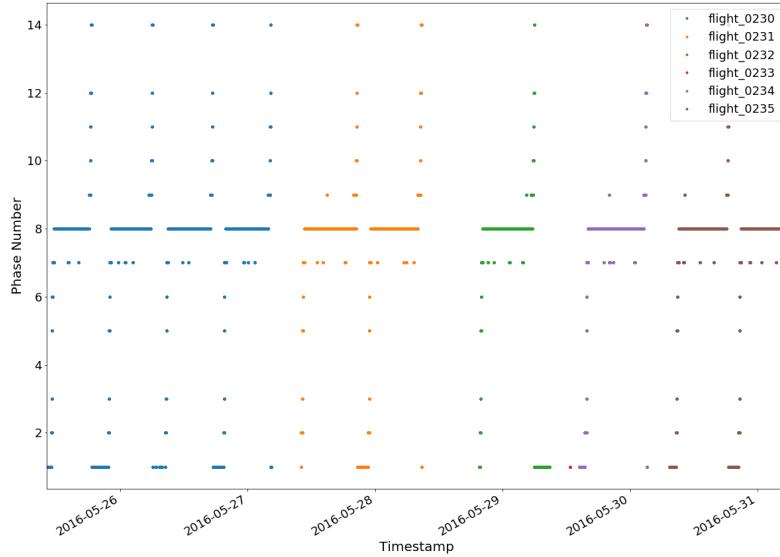


Figure 6.7: Categorization using partner's flight_id

One last possible way to correctly divide the flights data is using the first column, H51394154, which corresponds to the flight index of each flight. Nevertheless, during the data analysis certain inconsistencies, regarding this column, were identified. Figure 6.8 illustrates a small sample of the dataset. As the data is temporally ordered (according to the timestamp column) the column H51394154 should correspond to a counter that would be consecutively and contiguously increasing over time.

However, as observed in the highlighted regions, the values in the column H51394154 are not ordered sequentially.

| | H51394154 | timestamp | H20790882 | H90548828 | H38450254 | H32780567 | H35950744 | H67308013 | H81690022 | H97597396 | H24264101 | I |
|---------|-----------|---------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|---|
| 1228745 | 6305 | 2015-02-02 12:02:29 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228746 | 6306 | 2015-02-02 12:02:30 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228747 | 6307 | 2015-02-02 12:02:31 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228644 | 6204 | 2015-03-01 12:00:48 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228645 | 6205 | 2015-03-01 12:00:49 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228646 | 6206 | 2015-03-01 12:00:50 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228647 | 6207 | 2015-03-01 12:00:51 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228656 | 6216 | 2015-03-01 12:01:00 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228657 | 6217 | 2015-03-01 12:01:01 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228658 | 6218 | 2015-03-01 12:01:02 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228659 | 6219 | 2015-03-01 12:01:03 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228668 | 6228 | 2015-03-01 12:01:12 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228669 | 6229 | 2015-03-01 12:01:13 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228670 | 6230 | 2015-03-01 12:01:14 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228671 | 6231 | 2015-03-01 12:01:15 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |
| 1228748 | 6308 | 2015-03-02 12:02:32 | 1.0 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | 255.9 | |

Figure 6.8: Data inconsistencies found in the column H51394154

On the grounds that a proper way to accurately identify the data corresponding to each flight was not found, a methodology to correctly identify and categorize the flights was developed. Due to the importance of this task for the proper data analysis, different approaches were developed and compared, namely:

Approach 1 - Flight labeling based on the increasing phase number

The first simple and elementary approach was to consider that, according to the data, a flight ended when the phase number (column H20790882) of consecutive data rows decreased. Ideally, this would only happen when the phase transits from 14 to 1, which implies the beginning of a new flight. However, analysing the obtained results and the data, it was concluded that the phase number of a flight is not linearly increasing, as there are small variations between consecutive data rows. Figure 6.9 illustrates the evolution of the phase number of a particular flight, where it is possible to observe small fluctuations in the phase number of consecutive rows, specially between the phases 7 and 8.

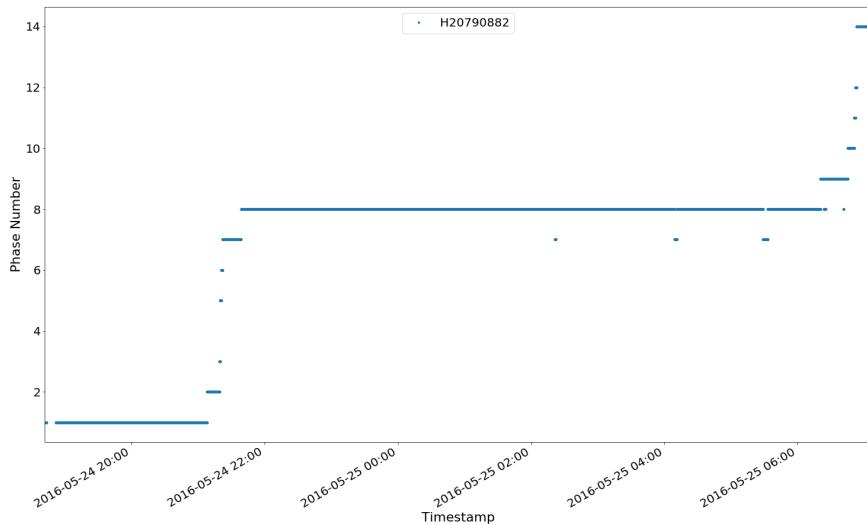


Figure 6.9: Evolution of the phase number of a particular flight

As such, this approach did not correctly identify the different flights, hence it was not used for the flight categorization.

Approach 2 - Flight labeling based on the phase number transition $14 \rightarrow 1$

The second approach was to consider only the phase transition $14 \rightarrow 1$ for the flight intermission, as, theoretically, it was expected to represent the end of a flight and the beginning of a new one. Nonetheless, by analysing the sensors patterns of each flight, it was noticed that at the end of certain flights there was a small portion of data with phase number 1 which still belonged to the current flight and thus the transition $14 \rightarrow 1$ did not corresponded to the break in flight.

Figures 6.10 and 6.11 reflect a case scenario where the flight identification was executed based on this approach. In these Figures, the different colors differentiate the different flights according to the performed labeling. As is observed, at the end of certain flights, like the flights 2, 3 and 4, there are some data points that still belong to the current flight, but were already included in the next flight, which made the flight labeling incorrect.

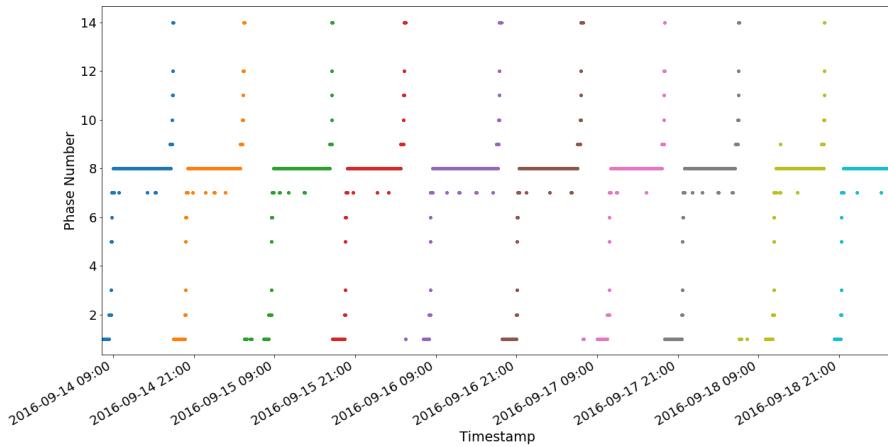


Figure 6.10: H20790882 - Phase Number

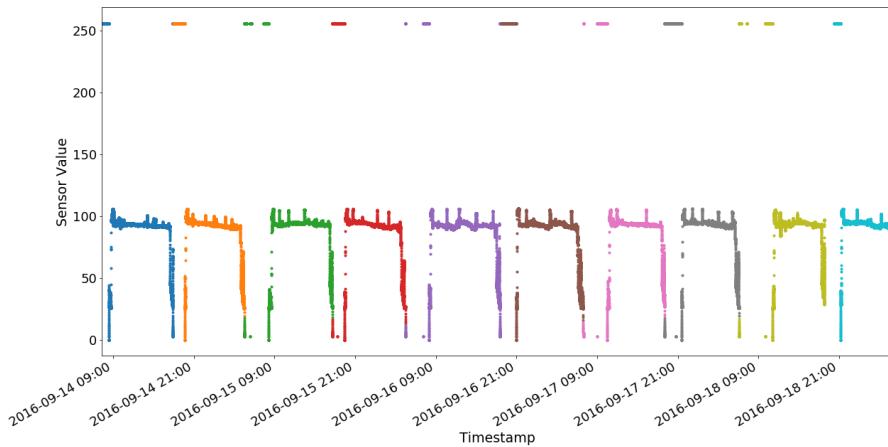


Figure 6.11: H38450254 - Sensor 1

As the flights identification was imprecise, this approach was not used for the flights categorization.

Approach 3 - Flight labeling based on the data rows timestamp

The next studied approach for the flight categorization was to consider that the break in flight occurred when there was a certain time difference between the timestamp of two consecutive data rows.

This approach was made on the assumption that a flight would not start immediately after the previous one had ended. Nevertheless, when analysing the data in more detail, it was possible to observe that certain flights occurred continuously in time, that is, without any time interval between them.

Figure 6.12 illustrates an example where the consecutive flights did not have any time interval separating them, and thus they were labelled as the same flight (same color), although incorrectly.

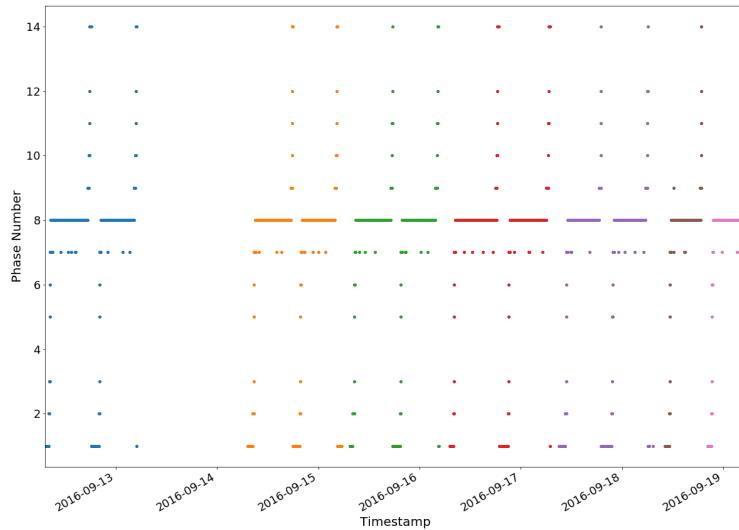


Figure 6.12: H20790882 - Phase Number

As the flights labeling of continuous flights was incorrect, this approach was not followed.

Approach 4 - Flight labeling based on the data rows timestamp and phase transition combined

The fourth considered approach for the flight categorization concerns the combination of two previous studied criteria, the time interval between data rows and the phase transition $14 \rightarrow 1$.

In a first iteration of the data, the time criteria is used to categorize the different flights. This means that, when the difference between the timestamp of two consecutive data rows was higher than a specific threshold (ex: 1 hour), a break in flight was performed.

Next, with the objective of improving the precision of the flight labeling, a second iteration was operated on the previous labeling. The goal is to identify ‘sub flights’ within the same flight (according to the previous labeling) in order to detect and handle the case scenarios similar to the one illustrated in Figure 6.12. In this second iteration, the phase transition ($14 \rightarrow 1$) was used. This implies that, in the same flight, when a transition from phase 14 to phase 1 occurred a new flight begins, and the flight labeling is adjusted.

Figure 6.13 illustrates the application of this approach to the same time interval as Figure 6.12. As can be concluded, the labeling imprecision presented in Figure 6.12 was solved.

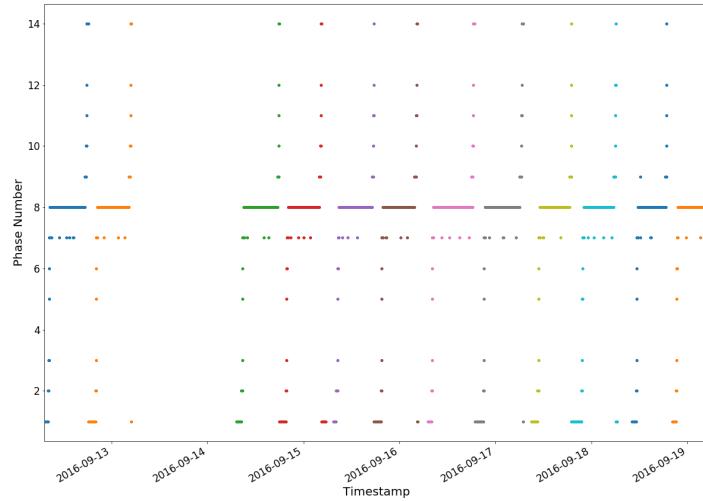


Figure 6.13: H20790882 - Phase Number

Due to the good results obtained in the flight identification and labeling, this approach was considered valid and is used for the flights categorization from this point onwards.

In a general perspective, Figure 6.14 illustrates the pseudo code of the algorithm explained before, for the correct flight identification and labeling.

Result: Flight Categorization - Labelling data according to the flights

```

#Flight categorization considering the Time criteria
for each row in Dataset do
    time_threshold = 1 hour
    cycles_threshold = 5000
    if current_timestamp - previous_timestamp > time_threshold :
        #End of current flight
        if number_flight_cycles < cycles_threshold :
            #Flight is too short
            #Time interval will no be considered
        else:
            #Registration the new flight
            #Labelling of the flight data
        #Start of the new flight in the next iteration
    end

    #Search for subflights inside each flight
    for each Flight do
        for each row in the Flight do
            if previous_phase == 14 and current_phase == 1 :
                #Identification of a subflight
                #Relabelling of the flight data accordingly
        end
    end

```

Figure 6.14: Pseudo Code of the flight labeling algorithm

Elimination of ‘small’ flights

After performing the flight categorization, it was noticed that some flights had a small duration regarding normal standards. After a discussion with the other partners regarding this fact, it was understood that these specific cases correspond to noise subsequent to possible maintenance tasks, flight tests or periods of time that the plane was logging data without any specific purpose.

As such, a specific threshold was considered for filtering the flights. The considered threshold is 5 hours. As the data was extracted from Boeing 747 that generally perform flights with duration of about 10 hours, the threshold defined is 5 hours. When the duration of the flight was inferior to the considered threshold, that specific time interval of data was not considered for analysis, as it does not correspond to a real flight.

Figure 6.15 represents the flight labeling for a specific time interval without any filtering and Figure 6.16 represents the flight labeling for the same time interval after performing the mentioned filtering.

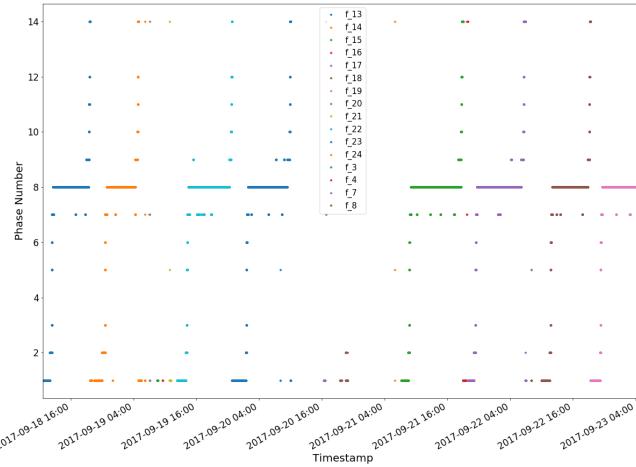


Figure 6.15: Original Flights Labeling

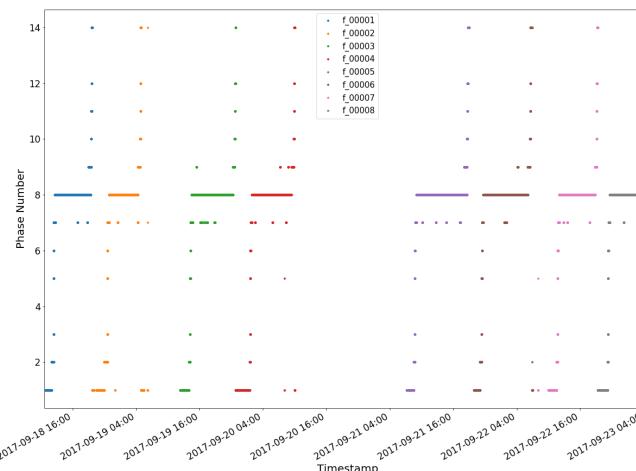


Figure 6.16: Flights Labeling after filtering

As can be observed, significant number of flights were discarded as they correspond to noise and not to real flights.

The application of the enunciated preprocessing techniques is expected to handle and solve certain problems detected previously, and with that, improve the usability of the data format, so that the models' accuracy can be higher.

6.1.3 Feature Selection

Regarding the feature selection, due to the data characteristics and organization and the small number of features, a simple correlation method, using the Pearson coefficient, is applied.

According to the provided dataset description, the 20 features of the dataset correspond to data from 5 sensors of 4 different Air Bleed systems, thus, a correlation analysis is interesting to be applied in order to confirm which features correspond to each of the 4 systems. It is expected that the features that correspond to the same sensor of the 4 systems have a high correlation.

Analysing the correlation heatmap of the 20 features of the aircraft 04388298, the following patterns are obtained:

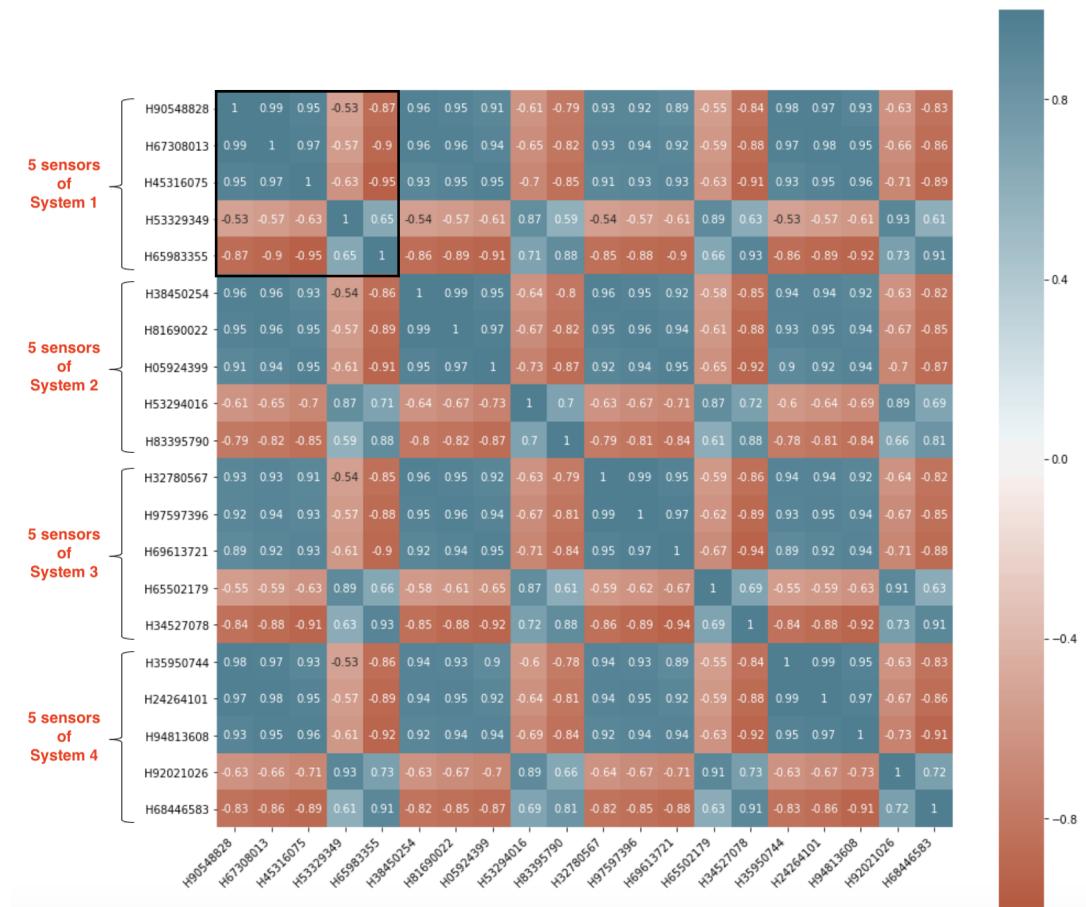


Figure 6.17: Correlation heatmap of the 20 features of aircraft 04388298

As can be observed, there are clear color patterns, represented by a square of blocks of length 5, that are repeated 4 times in the horizontal and in the vertical. Henceforth, the conclusion is that each 5 consecutive features (sensors) are related to a different Air Bleed system.

Next, in order to confirm that the features of each system correspond to the same sensors, Figure 6.18 and Figure 6.19 were generated.

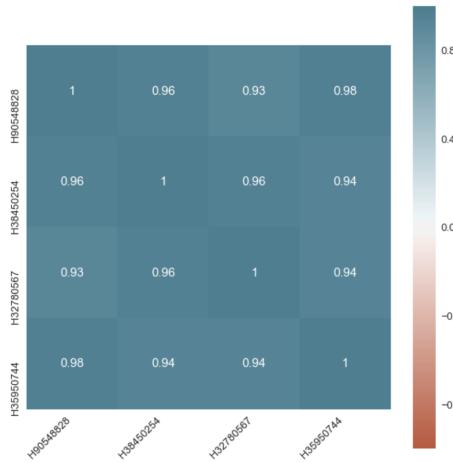


Figure 6.18: Correlation of the first feature of each system

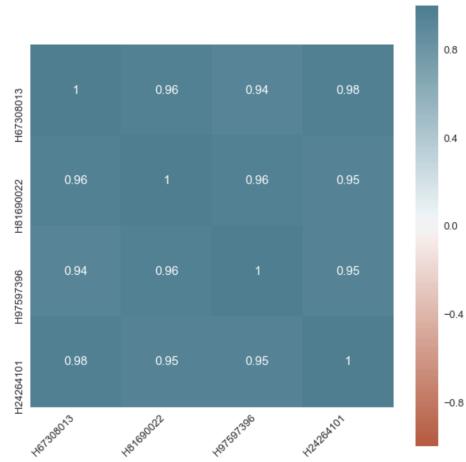


Figure 6.19: Correlation of the third feature of each system

Due to the high correlation illustrated in Figures 6.18 and 6.19, it is concluded that the features in each system, in fact, corresponds to the output of identical sensors.

Moreover, the redundancy presented in the 5 sensors of each system was also analysed. Figure 6.20 illustrates the correlation between the 5 features of system 1 of aircraft 04388298.

As observed by the correlation values, the first three sensors have a high correlation between them. In particular, the first two (Shaft 1 and Shaft 2) are extremely correlated, as their correlation value is around 0.99. This was already expected as the data patterns of the first two sensors are identical, due to the fact that they represent the same type of variable. Hence, these two features are redundant, and so, one of them can be removed. As the second feature has a higher correlation with the others, the second feature of each system should be removed, which corresponds to the columns H67308013, H81690022, H97597396 and H24264101.

6.2 Diagnosis of the system health condition

Based on the dataset provided and the lack of expert knowledge regarding the system operation and sensors impact on the overall system health condition, a data driven approach was followed for the diagnosis of the system health condition. Furthermore, a new formulation for the HI is proposed, which focuses on the analysis of the raw data retrieved from the 5 sensors of each Air Bleed system, in order to detect situations of extra or lesser degradation in the system.

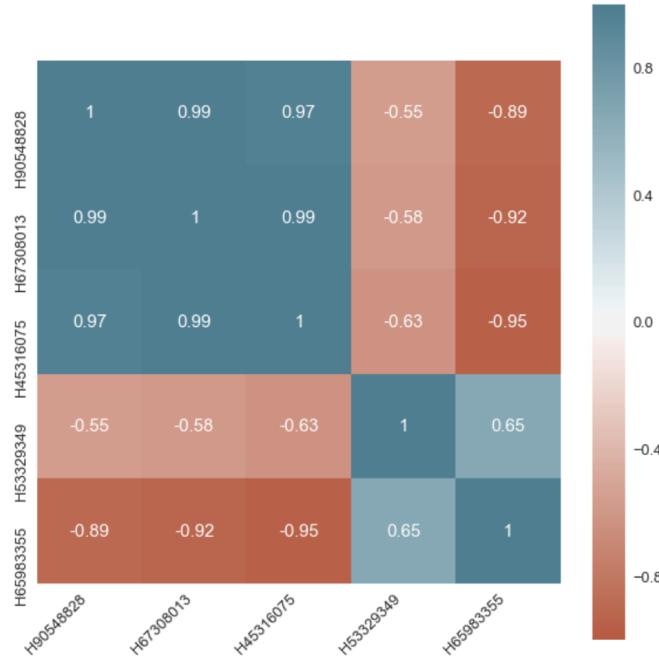


Figure 6.20: Correlation heat map of the 5 features of system 1

6.2.1 HI Formulation

As the HI aims to reflect the system condition, it is formulated as a combination of different variables that can positively or negatively affect the overall system health condition. The most basic and direct variable that affects the HI value is the duration (flight hours), i.e. the higher the number of flights performed, the more affected the system condition is. Other variables such as, the sensors impact on the system health condition, the deviation and variance in the sensors data, the flight conditions, the flight destiny and the meteorological conditions may also influence the system degradation, and consequently, the HI value. Each variable has an assigned weight that represents its impact on the HI computation.

Taking into account the provided data, physical information and knowledge obtained from the Air Bleed system, the HI (which reflects the system degradation) is calculated based on the analysis of the **deviation and variance** of the sensors data. The other mentioned variables that may affect the system health condition are not going to be considered in this work due to the absence of the required data and lack of expert knowledge.

This way, the HI is formulated in the following way:

$$HI = \sum_{k=1}^n \sum_{j=1}^p duration_{j,k} * (1 + (\alpha_{1,j,k} + \alpha_{2,j,k})) \quad (6.1)$$

where n is the number of flights, p is the number of aggregated phases in one flight and $duration_{j,k}$ is the duration of the aggregated phase j of flight k . The considered coefficients have the following meaning:

- α_1 : Coefficient that reflects the **deviation** in the sensors values. The values deviation is analyzed using the **mean** feature of the sensor values.

- α_2 : Coefficient that reflects the **variance** in the sensors values. The values variance is analyzed using the **standard deviation** feature of the sensor values.

Furthermore the α_1 and α_2 values are defined based on the following rules:

- α_1 : This coefficient reflects whether the sensor values are too deviated from the normal and standard values.
 - $\alpha_1 > 0$: In the case where the sensors values are **too deviated** (positively or negatively) from the normal values, it is considered a situation of **extra degradation**.
 - $\alpha_1 = 0$: If the sensors values are **within the normal range**, it is considered a situation of **normal degradation**.
- α_2 : This coefficient reflects whether the variance/oscillation in the sensor values is too significant, considering the standard and normal values of variation.
 - $\alpha_2 > 0$: In the case where the sensors values variation is **much higher than the normal situations**, it is considered a situation of **extra degradation**.
 - $\alpha_2 < 0$: If the variation values of the sensors is **much lower than normal**, it is considered a situation of **lesser degradation**.
 - $\alpha_2 = 0$: Otherwise, if the sensor values variance is **within the normal variance**, it is considered a situation of **normal degradation**.

Thus, taking in consideration the HI formulation (equation 6.1):

- If the sum of the *alpha* values is greater than 0 it will cause an extra increase the HI.
- If the sum of the *alpha* values is equal to 0, it will have no impact in the HI value, as it is a situation of normal degradation defined only by the *duration*.
- Otherwise, if the the sum of the *alpha* values is lower than 0, it will attenuate the HI increase.

This way, the HI value corresponds to the number of flight hours likely to be flown based on the deviation and the variance of the sensors values.

HI algorithm

With the objective of correctly assessing the health condition of the overall system, the analysis regarding the degradation is performed for each of the 5 sensors. In each sensor, the flights are analyzed individually, and in each flight the analysis is performed on each of the 5 aggregated phases.

This means that an α_1 and an α_2 are determined for each aggregated phase, of each flight, regarding each of the 5 sensors.

For a better understanding of the algorithm, Figure 6.21 illustrates a general flowchart of the algorithm developed for the assessment of the degradation through the analysis of the sensors raw data.

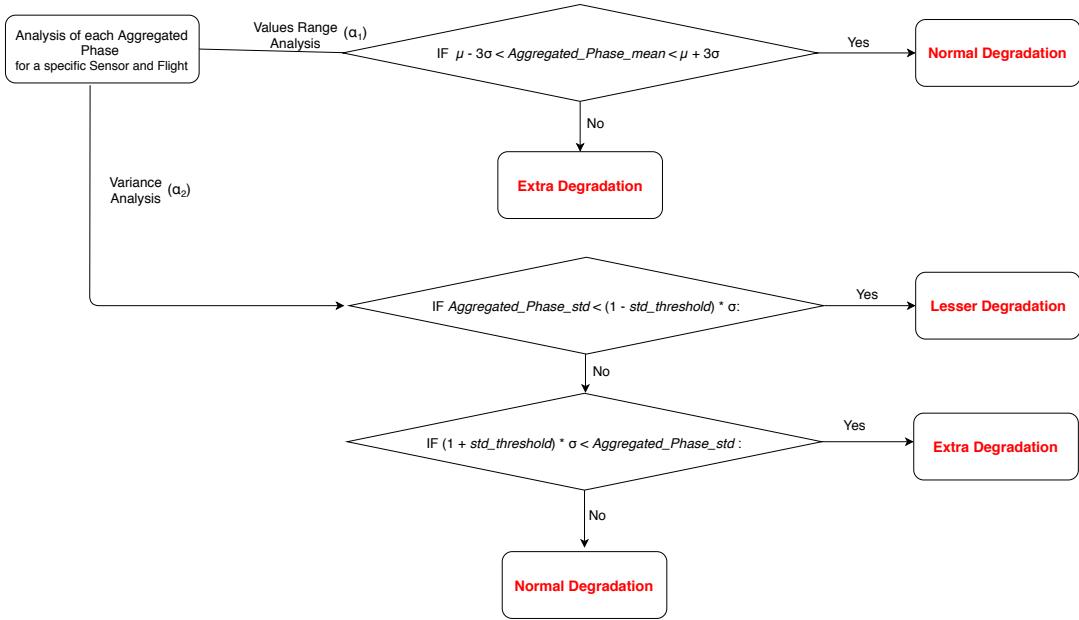


Figure 6.21: General representation of the developed algorithm

Following the representation in Figure 6.21, there are important parameters that need further explanation, namely:

- μ and σ : The μ and σ are used as the baseline values for assessing the degradation present in the sensors. They correspond to the average of the mean and standard deviation, respectively, of the sensor values considered typical and common.

These variables are calculated for each specific sensor and flight phase being analyzed. The μ and σ are calculated based on a data sample considered normal, that is composed by a set of flights that occurred without any FDE registered and with no visible anomalies or inconsistencies in the sensor values. An example of the sensor data considered normal is presented in Appendix C.

A general formulation of the μ is the following:

$$\mu_{p_j}^{s_k} = \sum_{m=1}^n \frac{\bar{s}_k f_m^{p_j}}{n} \quad (6.2)$$

where n is the number of flights, p_j is the considered aggregated phase, \bar{s}_k is the average of the values of sensors k and f_m the considered flight, in each iteration.

Similarly, the σ is represented by the following expression:

$$\sigma_{p_j}^{s_k} = \sqrt{\sum_{m=1}^n \frac{\text{std}(s_k f_m^{p_j})}{n}} \quad (6.3)$$

where n is the number of flights, p_j is the considered aggregated phase, $\text{std}s_k$ is the standard deviation of the values of sensors k and f_m the considered flight, in each iteration.

The purpose of calculating and using these values is to have a base for comparison, in order to conclude whether or not the sensors values are similar to the normal case scenario.

- **Aggregated_Phase_mean** and **Aggregated_Phase_std**: These two variables correspond to the mean and standard deviation values, respectively, of a specific flight, flight phase and the sensor being analyzed.

These are the variables that are compared with standard values (μ and σ), in order to conclude whether the time interval being analyzed corresponds to a situation of extra, less or normal degradation.

- **std_threshold**: The *std_threshold* corresponds to the threshold that determines the boundaries of the degradation condition based on the variance criteria (reflected by α_2). Figure 6.22 illustrates its meaning.

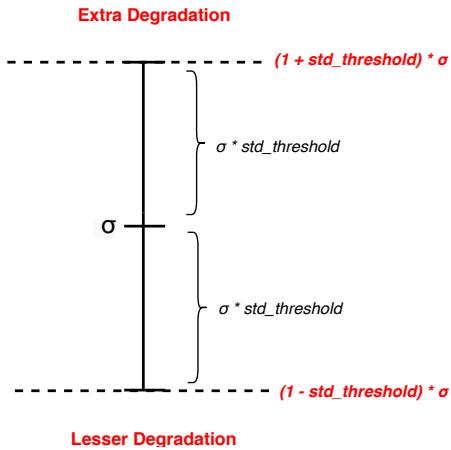


Figure 6.22: Representation of the *std_threshold* meaning

The *std_threshold* value used is **40%**, that is, 0.4. This value was obtained by experimenting and testing different values and evaluating the respective results. The validation criteria for selecting the most appropriate *std_threshold* value was the relationship between the increase in the degradation and the occurrence of a FDE. Theoretically, if a FDE occurred during a flight, it should reflect an extra increase in the degradation. As such, the *std_threshold* of 0.4 provided better results regarding the reflection of the HI increase when a FDE was triggered.

Regarding the Values Range Analysis (α_1), the boundaries defined for identifying the situations of Extra Degradation are $\mu \pm 3\sigma$. These were chosen based on the approach by Y. Wang et al. [101], where the $\mu + 3\sigma$ threshold was used for determining the beginning point of the degradation. Although the type of signals and sensors are different, it was considered a valid approach to test and, due to the interesting results obtained, the $\mu \pm 3\sigma$ threshold is used during this work.

HI meaning and α_1 and α_2 computation:

In an ideal situation, that is, without any incident that significantly increases the damage caused in the system, the HI is represented by the line present in Figure 6.23.

As can be observed, the HI is expressed in flight hours, and it represents the **number of hours of flight expected to be flown, based on the sensors data**. The blue vertical lines correspond to the end of a flight and thus the beginning of the next. This way, each flight corresponds to the time interval between two consecutive vertical lines.

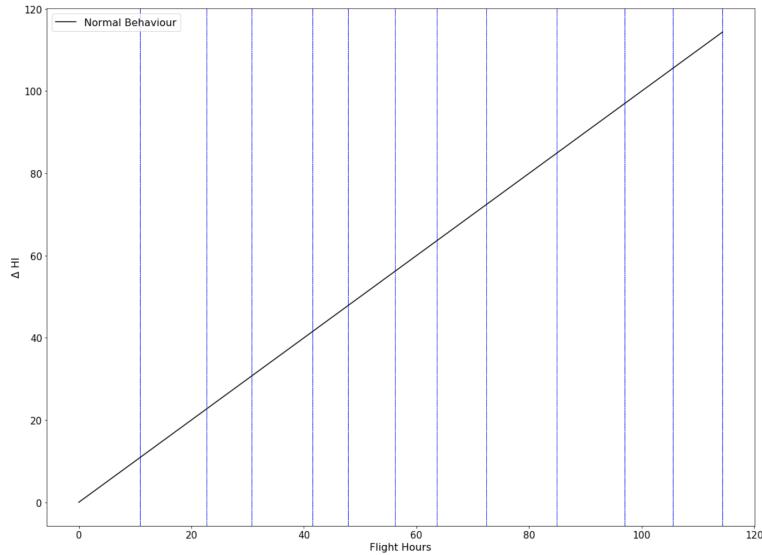


Figure 6.23: Case scenario with an ideal behaviour of HI

In a normal and ideal scenario (represented in Figure 6.23) where there is no FDE recorded and no adverse conditions during the flights, the HI is equal to the actual flight hours completed by the airplane, as illustrated in Figure 6.23. In this situation, and according to the HI formulation (equation 6.1), the alpha have no influence, thus ($\alpha_1 + \alpha_2 = 0$).

Nevertheless, the ideal scenario does not happen frequently, as there are external variables and conditions that affect the system condition, thus provoking extra damage.

Figure 6.24 illustrates the HI evolution for a specific time interval, retrieved from the system 1 of the aircraft 04388298.

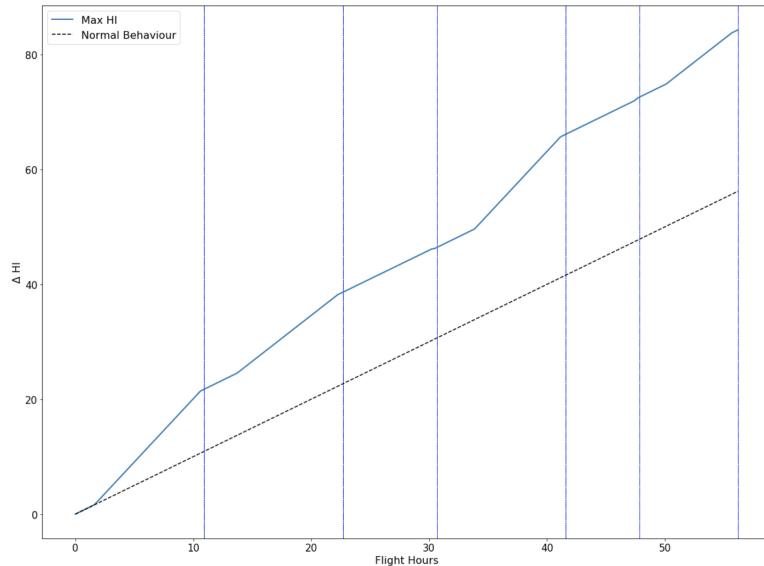


Figure 6.24: HI evolution in a real case scenario

As observed, there are moments where the HI value increases more significantly during some flights and others where the HI value increases at a lower rate during other flights. These HI fluctuations are due to the α_1 and α_2 values.

At the end of the 6 flights, the HI is about 94 flight hours. This means that, although only approximately 56 flight hours have been flown, the damage caused in the system is equal to 94 flight hours.

The values used for α_1 and α_2 are important as they represent the increase and decrease rate of the HI in each point.

The values used, in this work, for α_1 and α_2 are:

- α_1 :
 - $\alpha_1 = 0.6$ - If the sensor values **deviation** from the normal is **significant**.
 - $\alpha_1 = 0$ - If the sensors values are within the **normal range**.
- α_2 :
 - $\alpha_2 = 0.6$ - If the sensor values **variation** from normal is **significant**.
 - $\alpha_2 = 0$ - If the variation of the sensors values are within the **normal range**.
 - $\alpha_2 = -0.6$ - If the sensor values **variation** is below the normal values.

The selected values for the α_1 and α_2 (-0.6, 0 and 0.6) were chosen without any knowledge about the system operation or its age evolution. Since, there is no information regarding the ground truth of the HI no special tuning could be performed for α_1 and α_2 . Therefore, the choice of α_1 and α_2 focuses on interpretability of the values, rather than explainability.

For instance, if the sensors values regarding one flight presented low deviation of the values regarding the standard values and a high variation regarding the normal values, α_1 is 0 (normal degradation) and α_2 is 0.6 (extra degradation), respectively. Thus, the HI will increase by 0.6 ($\alpha_1 + \alpha_2$) during this flight.

Combination of the HI of the different sensors

Following the approach explained for the HI computation, an HI curve is obtained based on the data of each one of the 5 sensors from the Air Bleed system.

Figure 6.25 illustrates the 5 HI curves, that reflects the HI evolution according to the raw data of the 5 sensors, for a specific time interval.

With the objective of creating one curve that reflects the general health condition (HI) of the system, the 5 HI curves are combined into one.

Due to the absence of expert knowledge regarding the impact of each sensors in the Air Bleed system, it is assumed that all of them have the same importance and therefore all have the same weight in the combined HI curve. Hence, in each point, the combined curve is obtained by considering the maximum value of the sum of α_1 and α_2 of the 5 curves. This way, for each flight, the highest degradation presented in the 5 sensors is considered and comprised in the curve that represents the HI of the entire system.

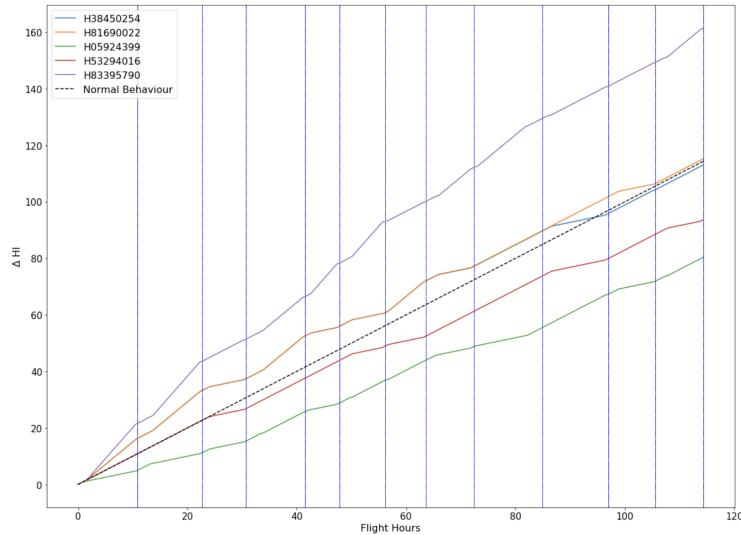


Figure 6.25: HI evolution in each of the 5 sensors from the Air Bleed system

Figure 6.26 illustrates the curve that reflects the HI of the overall system, which resulted from the combination of the HI curves of the 5 sensors.

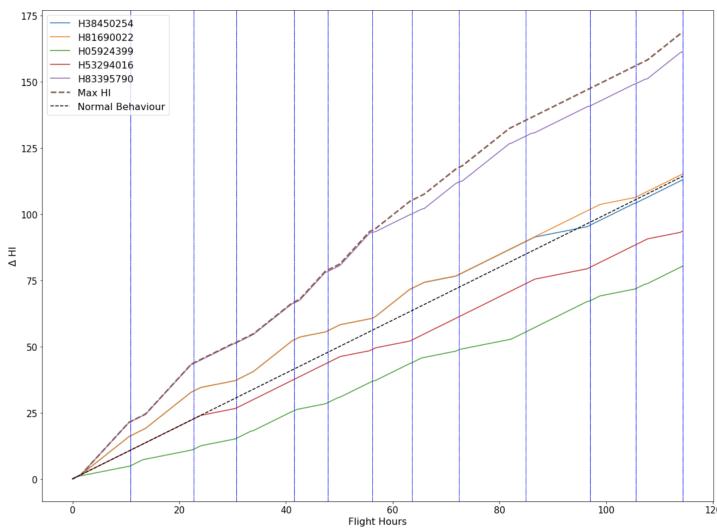


Figure 6.26: HI evolution in each of the 5 sensors from the Air Bleed system

As can be observed, the HI curve, which represents the system condition, results from the maximum of the slope of the 5 curves (representing the 5 sensors), regarding each flight.

For a more accurate and realistic combination of the HI from the 5 sensors, more specific knowledge regarding the impact of the sensors in the Bleed system is required.

6.2.2 Results of the HI algorithm

In this section the HI algorithm is applied to different time intervals. The aim is to present and discuss the applicability and correctness of the HI obtained in real case scenarios.

The chosen time intervals were selected from the Air Bleed data regarding different aircraft and Air Bleed system. The considered intervals correspond to different moments where the sensors data is of good quality. This means that the presence of missing data, in particular, incomplete flights or simultaneous flights is low. Also, all the illustrated results are obtained after executing the previous steps in the considered pipeline (in Figure 4.1).

The validation regarding the correctness of the HI obtained is performed by the presence of the FDEs. As the FDEs reflect some anomaly or fault in the system, it is expected that the HI increases significantly during the flights when the FDEs occurred.

For a better understanding and discussion of the obtained results, in each time interval the information regarding the FDEs and removals performed in each specific interval is illustrated in the HI graphs. The FDEs are illustrated by the “FDE” word in the respective flight and the removals are represented by a purple vertical line between the end of a flight and the beginning of a new one, with the word “Removal” associated.

As mentioned before, the different blue vertical lines determines the end of a flight and the beginning of a next one.

It is also worth mentioning that, in the presented graphs, the values on the y axis are expressed in ΔHI . This is due to the fact that they reflect the relative HI and not the absolute HI as there is no information of the moment where the respective degradation life cycle started. Ideally the HI is 0 when the component is new, in this instant a new degradation life cycle is initiated, as in this cases this information is unknown the ΔHI is represented instead of the HI.

Time Interval 1

The time interval 1 has the following characteristics:

- **Aircraft Tail Number:** 04388298
- **Air Bleed System:** System 2
- **First Instant:** 2016-05-24 6:0:0
- **Second Instant:** 2016-06-01 17:0:0
- **Duration (Flight Hours):** 115 hours
- **Number of Flights:** 12 flights
- **ΔHI :** 169 flight hours

Figure 6.27 reflects the HI evolution for the time interval 1.

Observing the HI graph regarding this time interval there was the occurrence of three FDE and two removals. This time interval is interesting in order to analyze the relation between the 3 types of data: sensors data, FDE and removals.

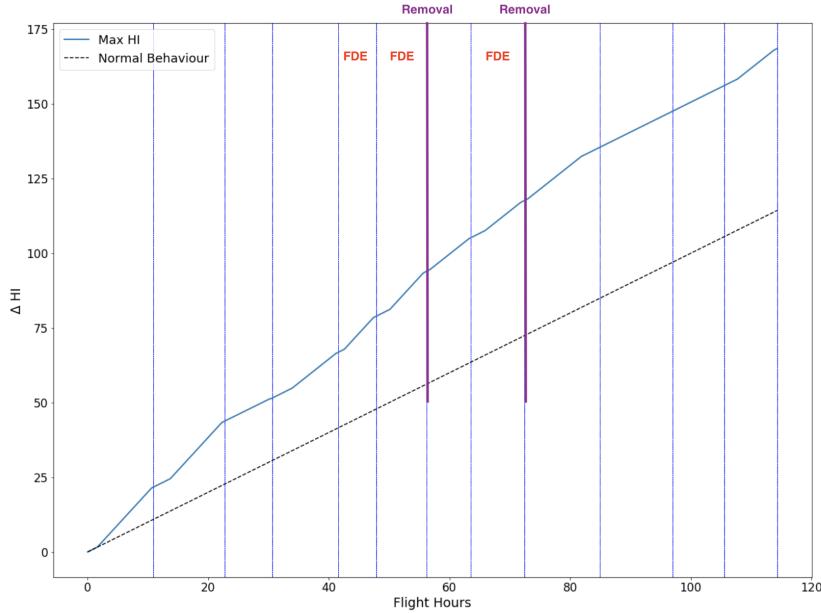


Figure 6.27: HI of Time Interval 1

In a general perspective, the HI is increasing over time, with some fluctuations in the HI slope. Observing the flights in the middle zone of the graph (from flight 5 until flight 9) there is a continuous increase in the HI slope. This is sustained and justified by the presence of FDEs in this specific zone. Thus it is possible to conclude that during these specific flights, there was some anomaly in the deviation or variance of the sensors data that increased the HI and also triggered 3 FDEs.

Analyzing the removals time, is possible to observe that the first removal was executed between 2 FDEs. Due to the presence of FDEs before and after the first removal and the fact that the HI did not decrease its slope after the removal, one can conclude that the first removal did not solve the problem identified by the 2 previous FDEs. Furthermore, a second removal of the same component was performed later which also suggests that the first one was not successful. After the second removal the FDEs were not generated in the next flights and the HI slightly reduces its slope which indicates that the second removal effectively solved the problem raised before.

By analyzing the other flights of the time interval 1, namely flights 1 and 2, an increase in the HI is also observed. As there are no FDE triggered in these flights, the identified anomalies in the sensors data may be contributing to the wear of the system, which may make it more prone to failure and contribute to the trigger of the next FDE (three flights ahead). Another possible reasons for the presence of fluctuations in flights where no FDE was raised are the presence of noise or logging errors, which distorts the truth expressed by the sensors data regarding its degradation or the inadequacy of the parameters chosen for the assessment of the degradation in the HI algorithm.

Time Interval 2

The next interval, time interval 2, has the following characteristics:

- **Aircraft Tail Number:** 10876262
- **Air Bleed System:** System 2
- **First Instant:** 2016-5-20 14:28:0
- **Second Instant:** 2016-5-29 4:32:0
- **Duration (Flight Hours):** 115 hours
- **Number of Flights:** 10 flights
- **ΔHI :** 154 flight hours

The HI evolution during this time interval is displayed in Figure 6.28.

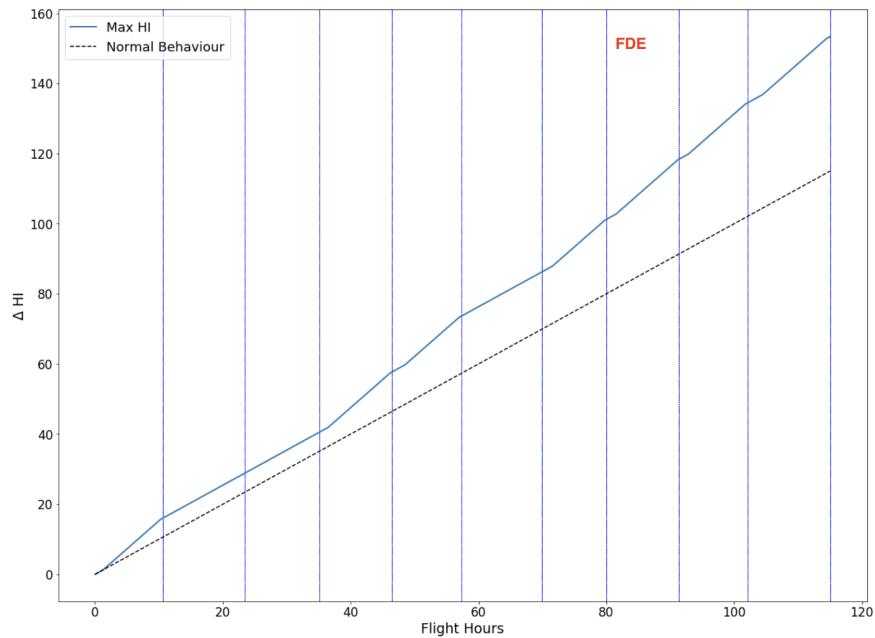


Figure 6.28: HI of Time Interval 2

In this time interval there was the occurrence of one FDE and no removals.

In a general perspective it is observed that the HI increase in each flight is not significant. This might indicate that the system condition in this specific interval is relatively new and so the degradation rate is lower.

In the first flights, the degradation rate is low, which indicate that the data patterns of the flights are regular and without significant anomalies regarding the data range.

Nevertheless, in the last 4 flights it is possible to see a raise in the HI values. Also, analyzing the time when the FDE was generated is possible to establish a correlation. The trigger of the FDE was a reflection of some anomaly in the system, which was also reflected in the sensors data, thus it causes the increase of the HI values in the last flights.

Time Interval 3

The next interval, time interval 3, has the following characteristics:

- **Aircraft Tail Number:** 9229765
- **Air Bleed System:** System 3
- **First Instant:** 2016-12-25 0:0:0
- **Second Instant:** 2017-1-19 9:0:0
- **Duration (Flight Hours):** 254 hours
- **Number of Flights:** 28 flights
- ΔHI : 436 flight hours

This time interval is longer than the others. A longer time interval was used so that the relation between the different sources of data could be present and explained during the discussion. Figure 6.29 reflects the HI evolution for time interval 3.

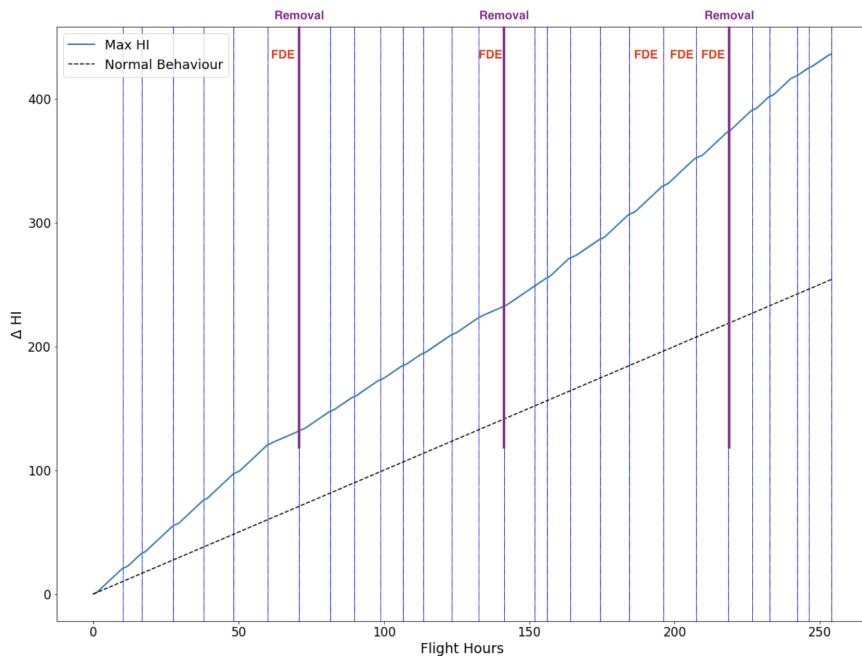


Figure 6.29: HI of Time Interval 3

This interval allows the correlation and understanding of the relation between the 3 data sources on wider scale, and its impact in the HI evolution, similarly to Time Interval 1.

From a general perspective it is clear that the HI slope varies between each removal performed, as during each interval between removals the HI reflects a different trend. This variation in the HI slopes makes sense as it demonstrates that the performed removals truly altered the system, which caused a change in the HI behavior.

However, the removals don't always contribute to a decreasing in the HI slope. For instance, before the second removal the HI slope of the flights is smaller than in the flights after the removal.

This means that the removal did not solve the problem, as the deterioration of the system continued. As a consequence, a next removal needed to be performed, as the previous did not solve the problem.

Another interesting conclusion is the impact of the FDEs on the removals. As observed in this example, a great number of the removals are executed due to the trigger of FDEs. This means that the FDEs most often reflect a real problem that requires the intervention of the maintenance team. Thereby, the FDEs can also be used as a warning to understand whether or not a specific removal solved the problem. If the removal effectively solved the problem it is expected that the occurrence of FDEs after the removal stops. Figure 6.29 exhibits this relation.

Time Interval 4

The time interval 4 is the last case scenario presented and has the following characteristics:

- **Aircraft Tail Number:** 04388298
- **Air Bleed System:** System 4
- **First Instant:** 2017-06-13 15:0:0
- **Second Instant:** 2017-06-26 10:0:0
- **Duration (Flight Hours):** 199 hours
- **Number of Flights:** 20 flights
- **ΔHI :** 217 flight hours

The HI evolution is reflected in Figure 6.30.

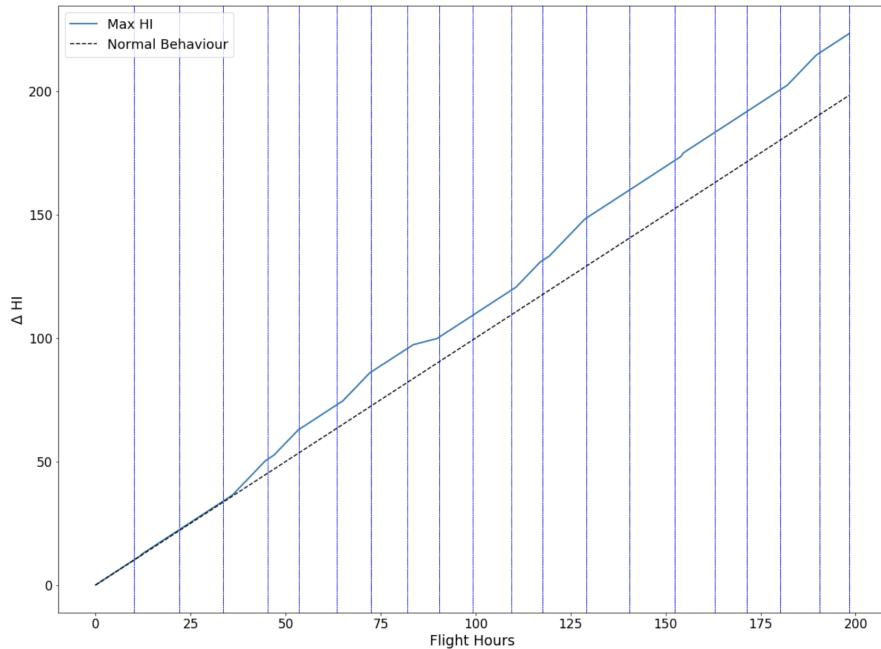


Figure 6.30: HI of Time Interval 4

This time interval reflects a different situation in comparison to the other time intervals.

In this specific time interval there are no FDEs recorded nor removals performed.

Thus, this time interval reflects a situation with normal degradation patterns where the mean and standard deviation of the sensors values are within the normal and common range. To validate these facts a screenshot of the sensor values can be accessed in Appendix C.

This way it is expected that the sensors describe regular and repeated data patterns regarding the performed flights during this time interval.

In case this type of degradation behavior continues over time it is expected that the HI value increases at a normal rate, contributing to a low and regular increase in the HI values.

Furthermore, time intervals similar to this (regarding the degradation evolution over time) are used for the training of the HI algorithm. As they are considered intervals with normal degradation patterns, they are used as the base for computing the typical values of the mean and standard deviation of the five sensors (μ and σ). Based on these calculated values, the cases with extra, normal and lower degradation are identified. This is fully detailed in Section 6.2.1.

Overall, by analyzing the results for these four time intervals, interesting results were obtained and some conclusions can be made. It was clear that in most of the cases the presence of the FDE was related with an increase in the HI values. This was the expected behavior and helps validating the approach followed for the computation of the HI.

Nonetheless, there were also cases where the increase in the HI did not match the trigger of a FDE. Three main reasons were found to justify this fact, firstly the sensors data might be noisy which mislead the truth regarding the degradation reflected in the sensors.

Secondly, that the increase in the HI does not trigger an FDE but may lead and contribute to the occurrence of one in a near future.

Thirdly, the anomalies regarding the deviations in the mean and standard deviation value may not indicate a situation of extra degradation and thus no real FDE should be triggered.

Another interesting conclusion is the relation between the removals and the HI trend. Specifically in time interval 3, it was observed that the executed removals changed the HI trend in the flights after the removals. However, the removals did not always succeed in solving the problem as confirmed by the increase of the HI after the removal and the occurrence of a posterior removal.

Lastly, it was also clear that HI values increased more significantly in the time intervals with more triggered FDEs. This confirms the relation between the presence of FDEs and the increase of the HI values.

6.3 Prognosis of the system health condition

The last step of the pipeline is the prediction of the system's future degradation behavior, which is reflected in the RUL value calculated.

The RUL information is the most interesting and valuable for the maintenance team as it gives a prediction of the component's useful life remaining, that is, the number of flight hours in which that component can still operate without any fail.

Given the data provided, and the representation of HI previously formulated, the approach followed for the RUL prediction is a similarity-based approach, identical to the one developed with NASA's dataset presented in Section 5.1.5.

Although the HI representations are different, as in NASA's approach the HI is expressed as a percentage of health condition and in the Air Bleed system the HI is expressed in flight hours, the approach is still valid to be used in both cases.

Other approaches for the RUL computation were considered, as for example an extrapolation method, but as the information regarding the maximum value of the HI in the current scenario (Air Bleed system) was missing, this approach was not considered valid.

Thus, the approach used for predicting the RUL is a similarity-based approach, in which, the Euclidean distance is used to assess which degradation presented in the training trajectories better describes the degradation presented in the test trajectories. Then, based on the most similar trajectories the RUL is computed. Instead of a single value for the RUL, an interval for the RUL value is predicted with the goal of representing the uncertainty associated with the prediction.

6.3.1 Identification of the Removal Intervals

The first step in the RUL interval computation is the definition of the removal intervals, which correspond to the training trajectories.

The correct identification of the different trajectories (removal intervals) is important in order to create a set of base trajectories used in the similarity-based method for the prediction of the RUL.

A removal interval corresponds to the time interval between 2 consecutive removals executed on the same Air Bleed system, same aircraft and with the objective of resolving the same problem. This way each removal interval identified represents an entire life cycle of a specific component, that is, from the moment it was new (first removal) to the moment it was replaced due to degradation or failure (second removal).

Nonetheless, not all removals presented in the removals file can be used for limiting the removal intervals. In order to assure that the removal intervals effectively correspond to a entire component life cycle, the first removal needs to have resolved the problem detected previously, so that a new life cycle can be initiated.

This information regarding the removals effect and impact in the detected problems is not provided in the removals file.

Thus, in order to identify which removals, in fact, solve a detected problem, the FDE information is used. For this work, and in alignment with the other ReMAP partners opinion, it is considered that if there is the occurrence of FDEs just prior to the removal and then, after the removal, no more FDEs were triggered, then is possible to conclude that that removal, in fact, solve an existing problem. This way a filter of the removals that actually solved an existing fault can be performed.

Figure 6.31 illustrates a theoretical scenario which correspond to the detection of a removal interval.

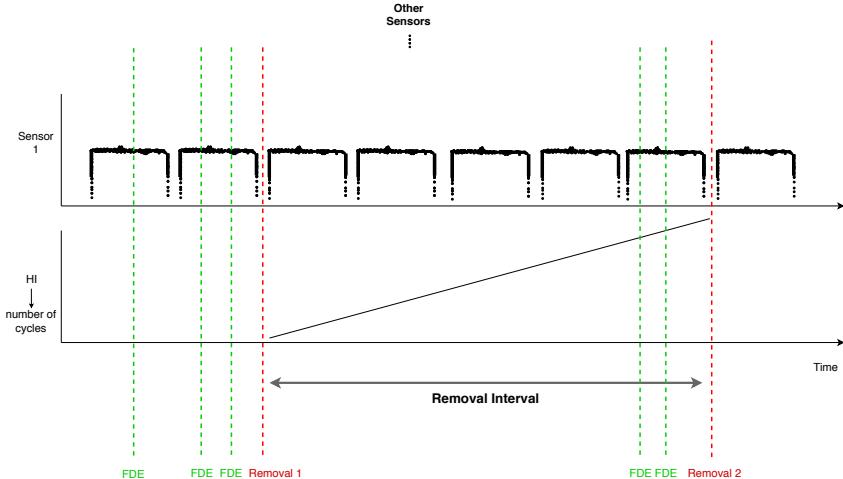


Figure 6.31: Removal Interval example

As observed in the Figure, in order to correctly identify the removal intervals, the combination of the sensors data, FDE and removals data is required. The rules used for crossing the different data sources are described in Section 6.1.1.

After analyzing and crossing the data, 8 time intervals were found in the Air Bleed data, which are represented in the following Table 6.2. In order to calculate the degradation (HI) in each trajectory found, the proposed algorithm for the HI computation is used.

Table 6.2: Removal Intervals

| | Aircraft | System | Problem | First Removal | Second Removal | Number of Days | Number of Months | Number of Flights | HI |
|--------------------|----------|--------|------------|---------------|----------------|----------------|------------------|-------------------|------|
| Removal Interval 1 | 4388298 | 2 | CNTRL PRV | 28/05/2016 | 11/07/2016 | 44 | 1,47 | 49 | 800 |
| Removal Interval 2 | 47304516 | 4 | CNTRL PRV | 14/01/2017 | 21/04/2017 | 97 | 3,23 | 116 | 1000 |
| Removal Interval 3 | 92229765 | 3 | CNTRL PRV | 17/01/2017 | 15/4/2016 | 88 | 2,93 | 106 | 1400 |
| Removal Interval 4 | 88022223 | 3 | CNTRL HPBV | 24/08/2017 | 12/10/2017 | 49 | 1,63 | 69 | 800 |
| Removal Interval 5 | 73646573 | 4 | CNTRL PRV | 27/10/2016 | 19/01/2017 | 84 | 2,8 | 126 | 1100 |
| Removal Interval 6 | 47304516 | 1 | VLV HPSOV | 27/08/2017 | 12/11/2017 | 77 | 2,57 | 102 | 1100 |
| Removal Interval 7 | 10416261 | 2 | CNTRL HPBV | 24/07/2016 | 12/09/2016 | 50 | 1,67 | 67 | 1000 |
| Removal Interval 8 | 4388298 | 1 | CNTRL PRV | 26/12/2015 | 19/02/2016 | 55 | 1,83 | 64 | 600 |

Table 6.2 demonstrates several aspects regarding the removal intervals identified. As can be observed, even within the removal intervals of the same problem, the duration of the interval, the number of flights and the HI varies moderately.

Figure 6.32 represents an example of a Removal Interval (Removal Interval 1), where the different HI fluctuations over time can be observed.

6.3.2 Computation of the RUL interval

After defining the training trajectories (removal intervals) the RUL interval is computed following 3 steps:

1. Compare the Similarities between the Test trajectory and the Training trajectories.

The **Test trajectory**, T_{test} , corresponds to the trajectory whose future degradation is to be predicted.

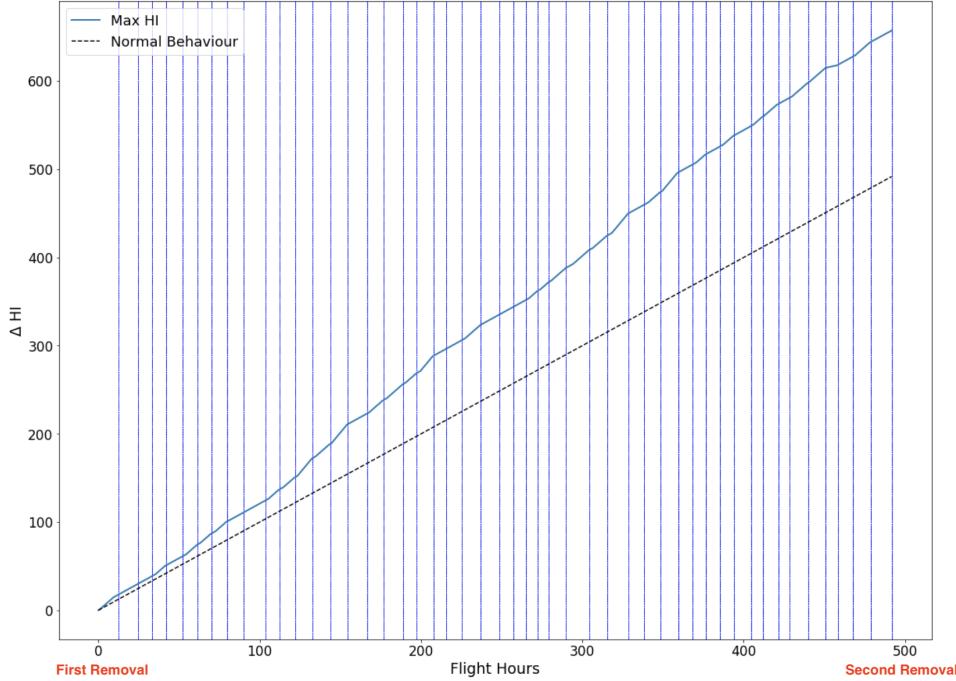


Figure 6.32: Removal Interval 1

The **Training trajectories**, $T_i, i \in [1, 8]$, correspond to the 8 trajectories identified and illustrated in Table 6.2.

The similarity criteria used is a weighted Euclidean distance. The weights are used to give more or less relevance to each flight in the calculated distance.

The formulation for the calculation of the distance between the Test trajectory and each of the Training trajectories is the following:

$$euc_distance(T_{test}, T_i) = \sqrt{\sum_{k=1}^n \omega_k (T_{test,k} - T_{i,k})^2} \quad (6.4)$$

where n is the number of considered flights, i is the Training trajectory being analyzed and ω is an array of the flight weights.

The objective is to give most weight to the more recent flights and less weight to the older flights, this way similarities between recent flights of different trajectories have more impact than similarities between past flights.

2. Select the most similar Training trajectories to the Test trajectory.

The criteria used for selecting the most similar trajectories, is the weighted Euclidean distance, formulated previously.

Based on the calculated distances, the trajectories used for computing the RUL interval of the test trajectory are selected based on the combination of two criteria:

- **Specific distance threshold:** By specifying a fixed threshold, the calculated trajectories distances exceeding that threshold are discarded, otherwise, if the trajectories distance are within the defined threshold, these are selected.
- **Minimum distance:** The second criteria used is considering the trajectories with the minimum distance to the test trajectories.

Figure 6.33 illustrates an example of the most similar trajectories to a test trajectory. In this case, the test trajectory corresponds to Removal Interval 6.

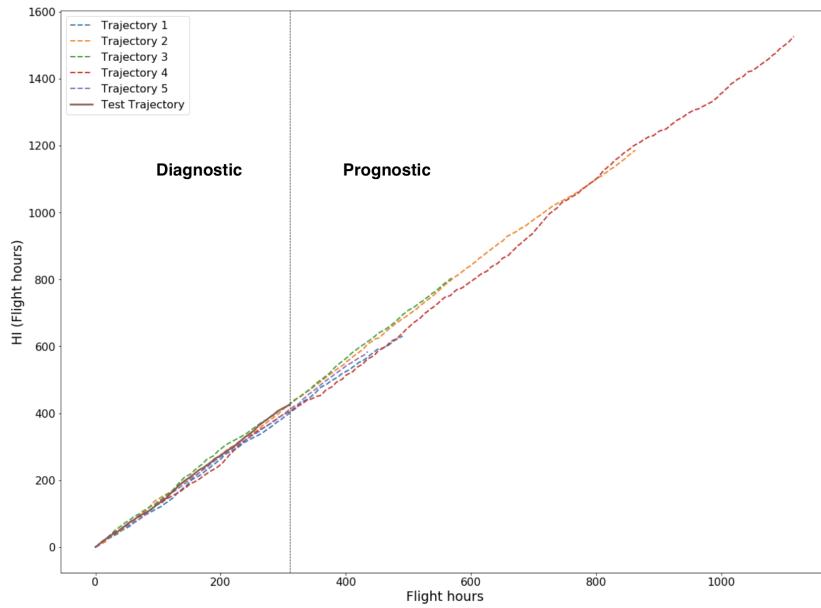


Figure 6.33: RUL prediction

3. Calculation of the RUL interval, using the minimum and maximum RUL of the selected trajectories.

Based on the RUL of the most similar trajectories, the RUL interval for the Test trajectory, T_{test} , is defined by the minimum RUL and maximum RUL of the selected trajectories.

Figure 6.34 illustrates an example of the RUL prediction over time, based on the selected trajectories illustrated in Figure 6.33.

As observed in the Figure 6.34, the upper boundary of the RUL interval is calculated using the maximum RUL at each point and the lower boundary is calculated using the minimum RUL at each point.

The RUL interval is predicted at the end of each flight, using a sliding window approach that is increasing over time. This means that each point of the graph reflects the RUL interval prediction based on the previous time range plus one more flight added to the sliding window.

6.3.3 Results of the RUL algorithm

This Section presents some results of the application of the RUL algorithm for different time intervals in order to discuss its applicability to real case scenarios.

In order to compare the predicted RUL with the ground truth, trajectories from the removal intervals identified are used as test trajectories. The remaining 7 trajectories are used as training trajectories for predicting the RUL of the test trajectory. In particular, Removal Interval 3 and Removal Interval 1 are used as test trajectories, to demonstrate the applicability of the RUL algorithm proposed.

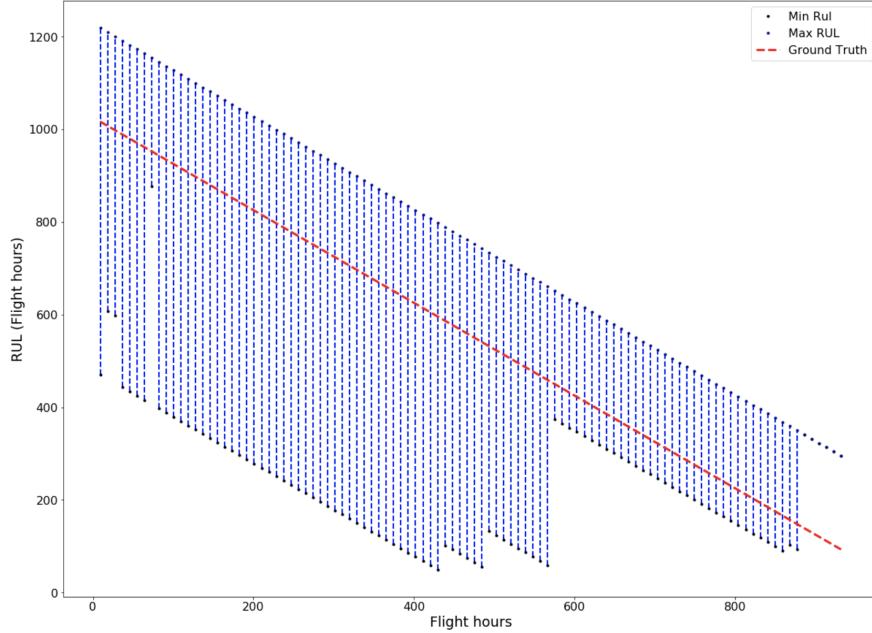


Figure 6.34: RUL prediction

The identified removal intervals are illustrated in Table 6.2.

As mentioned before the upper boundary is defined by the maximum RUL of the most similar trajectories and the lower boundary defined by the minimum RUL of the most similar trajectories. The red line corresponds to the ground truth.

Removal Interval 3

This removal interval has the following characteristics:

- **Test Trajectory:** Removal Interval 3
- **Training Trajectories:** Other 7 Removal Intervals

The RUL estimated for Removal Interval 3 is illustrated in Figure 6.35

Analyzing the ground truth line in the Figure it is possible to observe that it is encompassed in the predicted RUL interval. Nevertheless, there is some variance associated with the RUL interval defined, this may be due to the fact that a small set of training trajectories was used, which contain different lengths. The diverse lengths in the training trajectories are also reflected in the graph of Figure 6.35.

The fluctuations in the upper and lower boundaries of the RUL are due to a change of the most similar trajectories. Some of these changes, specifically in the later flights, are due to the end of the training trajectories, thus other larger trajectories are used for defining the RUL interval.

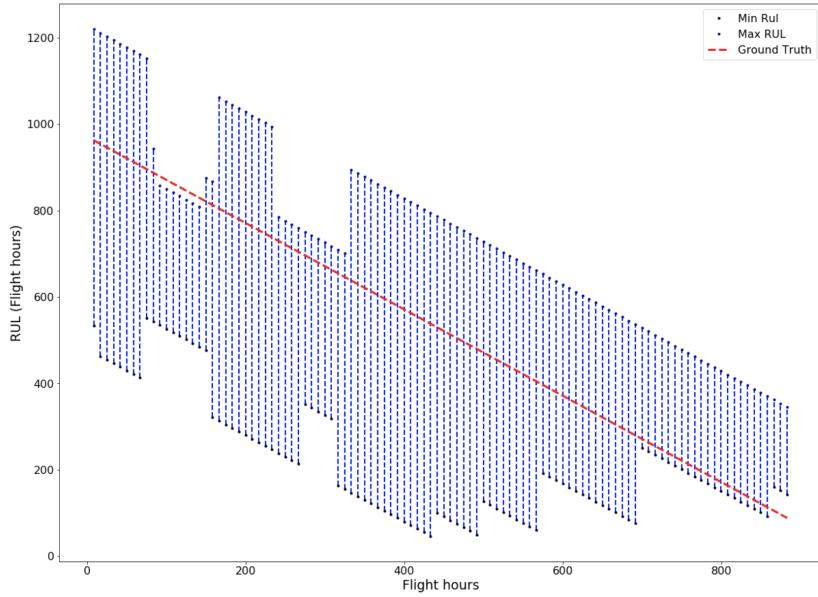


Figure 6.35: RUL estimated for Removal Interval 3

Removal Interval 1

This removal interval has the following characteristics:

- **Test Trajectory:** Removal Interval 1
- **Training Trajectories:** Other 7 Removal Intervals

The RUL estimated for Removal Interval 1 is illustrated in Figure 6.36

In this removal interval, by analyzing the upper and lower RUL boundaries it is clear that the set of the most similar training trajectories doesn't change much over time as the boundaries have a regular decreasing behavior. Also, similar to the previous case scenario, the high variance in the RUL intervals is obvious.

The ground truth, represented by the red line, is closer to the lower boundary which indicates that the similar trajectories with the lowest RUL are identical to the test trajectory, with regards to the degradation behavior.

Overall, observing the results obtained for the RUL estimation based on a similarity-based approach, the results are not the most interesting regarding the precision in the RUL prediction but are promising in a way that they provide a correct idea and approximation of the true RUL of the test trajectories.

One obvious point of the results obtained is the high variance in the RUL intervals obtained. This fact is easily justified by two main reasons.

The first is the fact that the set of the training trajectories is small and thus it is more difficult to find trajectories with similar behavior to the test trajectories.

The second, and most important one, is the fact that the initial health condition of the replacing parts is not known. This means that in the First Removal the replacing part may be new or used, and this influences the length of the respective trajectory.

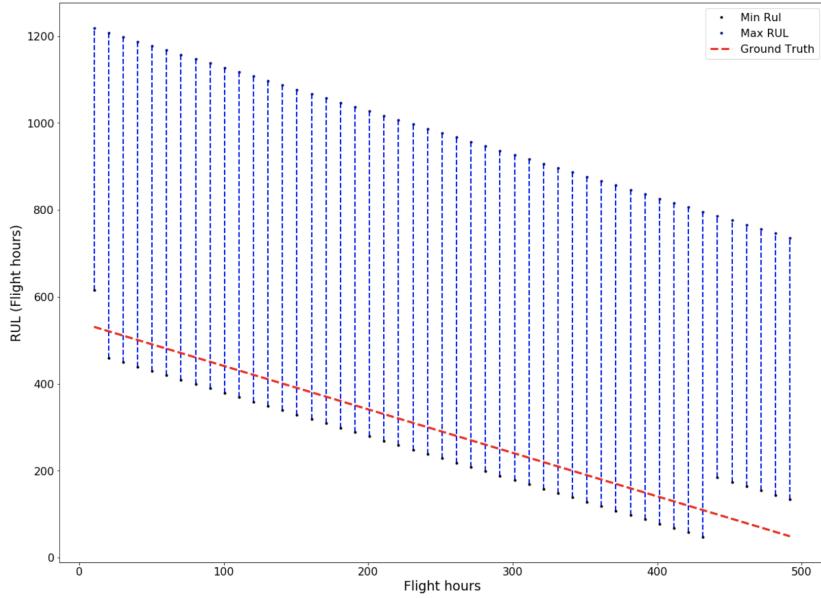


Figure 6.36: RUL estimated for Removal Interval 1

For instance, if the replacement is new the trajectory length is longer, as there is no initial degradation. Opposingly, if the replacing part is used, it already has some degradation embedded, thus the trajectory length is shorter.

Still and all, the approach for the RUL was developed mainly as a proof of concept. As the set of training trajectories is too short, no profound or precise analysis was able to be performed. The aim was to build a technique for the RUL estimation that completes the PHM methodology developed in this work and already provided interesting results to be explored in future work. Based on these points, the developed technique for the RUL estimation achieved the expectations and will be explored in future work.

6.4 Conclusions regarding the Diagnosis and Prognosis

Regarding the Diagnosis, expressed by the HI value, a singular and original approach was used. Analyzing the State of the Art methods enclosed in the HI computation, no direct approach was found suitable to be used in this work. Although the HI concept has already been objective of significant study, most of it corresponds to specific subject areas applied to private datasets. Also, the data specifications and structure, which included three different sources (sensors data, FDEs and removals) and the fact that the sensors data are anonymized contribute to not finding relevant work applied in this particular context of aeronautics.

The HI formulation corresponded to a novel approach for diagnosing the system health condition, as no similar approach was found in the literature.

The proposed HI formulation took into consideration the different factors that impact the system condition, like the sensors impact in the system condition, the variations in the sensors data and the flight conditions. Therefore, and based on the data provided, the HI was calculated through the analysis of time domain features of the raw data extracted directly from the sensors.

Due to the lack of expert knowledge regarding the operation of an Air Bleed system, the feedback from the ReMAP partners, specially from the aircraft engineers, was very important and relevant due to their expertise in the aircraft industry and in the specific system analyzed, the Air Bleed system.

In terms of results, the examples discussed in Section 6.2.2 showed an interesting relation between the presence of FDEs and the increase in HI in the near flights. Also the presence of removals was reflected by changes in the HI trend which proves that removals effectively cause changes in the system behavior. Although these positive aspects help validating the proposed formulation for the HI, there are other time intervals where the HI wasn't the expected, this divergence may be the result of numerous causes, such as noisy data, logging errors or inadequacy of the parameters used in the HI implementation. Moreover, more knowledge and know-how about the impact of each sensor in the health condition of the overall system, would contribute to a more accurate and correct combination of the sensors HI behavior for determining the general Air Bleed system condition.

Also regarding the HI computation, the choice or determination of the parameters to use, for example α_1 and α_2 , can still be improved and grounded by well-founded facts, instead of using a trial and error approach. Nevertheless, this does not compromise the validation of the proposed HI formulation.

Concerning the Prognosis component, the RUL computation approach used was adapted from an approach developed and applied in the Turbofan engine.

Due to the good results obtained previously, a similarity-based approach was used to estimate an interval for the RUL values. Although this approach correspond to the State of the Art and was already used in the Turbofan dataset, it was considered suitable to also be used in the Air Bleed dataset. Moreover, this approach is adequate for representing and predicting the degradation behavior of the aircraft systems, as it deals properly with the non linearity expressed in the degradation patterns.

The results obtained and presented in Section 6.3.3 were promising, although associated with a big variance interval. This is a direct consequence of the fact that a small set of training trajectories was used and also due the uncertainty of the health condition of the replacing parts in the removals. Nonetheless, with a bigger set of trajectories and more information regarding the condition of the replacing parts, this approach seems to be valid for the prognosis of the future degradation behavior of the trajectories.

This page is intentionally left blank.

Chapter 7

Conclusion

As emphasized numerous times in the document, this work aimed to create a PHM methodology, based on machine learning and CBM fundamentals, able to detect, diagnose and predict the health condition of specific aircraft systems. The fact that usually the system's degradation is not characterized by a linear behavior and it is affected by several variables (internal and external) difficult to assess or control, makes these tasks significantly more challenging. Thus and so, this work relevancy is sustained by its importance and potential impact in the maintenance routines and also by the complexity in the tasks involved.

In the initial part of this work, the study of the current approaches for performing aircraft maintenance and the identification of problems involved in those approaches served as an introduction to the aeronautical field. Furthermore, it helped the understanding of some important concepts like flight cycles, HI and RUL and stimulated the study of the different State of the Art methodologies that can be applied in PHM systems.

The exploratory case studies (Turbofan engine and Brakes), although they do not correspond to the main contribution of this work, served as preliminary work where different approaches for the HI and RUL computation were experimented and tested.

In particular, using the Turbofan datasets, a common formulation of the HI was used, where the HI represented the percentage of health condition of the subsystem in analysis. Regarding the RUL estimation methods, three State of the Art approaches were implemented: a Similarity-based, an Extrapolation-based and a Neural Network-based approach.

With respect to the Brakes dataset, the HI information was already given in the dataset features. The method for RUL estimation was a Linear Regression, as it described properly the HI evolution over time.

The main contribution in this work was the one performed over the Air Bleed dataset. This dataset comprised raw data retrieved from the sensors of the Bleed system of Boeing 747, thus it corresponds to a real and complex scenario, from which the analysis and results obtained are more valuable, relevant and suitable for being used in a real world context. For the Air Bleed dataset, a new formulation for the HI was created, in which, based on the analysis of the time domain features (mean and standard deviation) of the sensors values, the degradation condition of the system was extracted.

Regarding the RUL computation, the similarity approach developed in the Turbofan dataset was suitable and adaptable to be used in the Air Bleed dataset. Thus, the adaptation of the similarity-based technique developed in the case study of the NASA Turbofan dataset was performed for the RUL estimation in the Bleed system.

Following the conclusions affirmed in Section 6.4, the methods developed for the diagnosis and prognosis of the Air Bleed system overall obtained good and interesting results, that proves that the PHM approach applied in the Air Bleed system is valid to be applied in the aircraft field in a near future.

Nevertheless, the methods for diagnosis and prognosis will not replace or independently manage the aircraft maintenance routines. From a general point of view, the approach developed in this work is not intended to replace the expertise of the aircraft engineers, instead, the aim is to provide them with more relevant information regarding the system condition and its future behavior. The output provided of this work, that is the HI and RUL of particular aircraft system, is within the maintenance engineers scope and knowledge. Thus, combining their knowledge with the provided reports and documentation sustaining and validating the models accuracy, fundamentals and relevance, the aircraft engineers are capable of analyzing and interpreting them in order to better plan and execute the maintenance tasks that may be performed in order to improve aircraft maintenance.

From the point of view of the ReMAP project, this work will be inserted and combined in a more complete and robust platform that will comprise different components in order to build a more complete, accurate and solid solution to be presented as the final outcome of the project. In particular, the work provided by this Work Package (WP5) will serve as input for WP6, which is responsible for developing decision-based models capable for optimizing flight scheduling.

Concerning the interpretability and explainability of the developed work in the ReMAP project, the performed diagnosis and prognosis will be sustained and validated by documentation and reports. These will provide information regarding the diagnosis and prognosis (HI and RUL) meaning, its fundamentals and uncertainty levels, as well, as statistics regarding the methods' accuracy supported by auditable records. Furthermore, the obtained models will need to be explained in order for the data engineers to understand the relevance and validation of the obtained results. Also, the models created will be tested and validated in other aircraft systems to verify the model's accuracy and versatility.

Analysis of the scheduled planning

The proposed work planning for the last months, represented in a *Gantt Chart*, is illustrated in Figure 7.1.

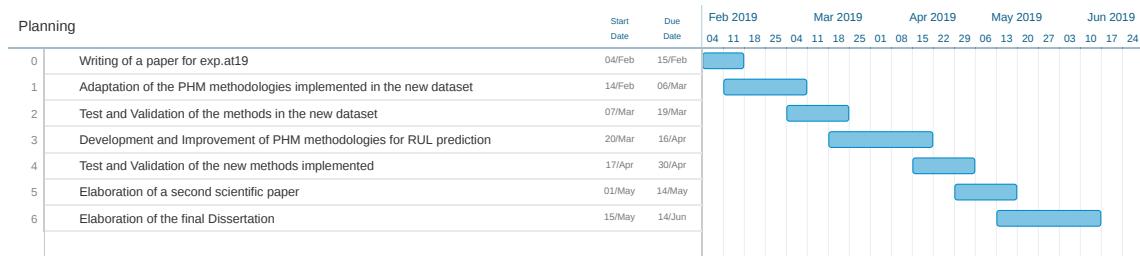


Figure 7.1: Proposed Work Planning

The *Gantt Chart* reflecting the actual work performed during the last part of this Thesis is illustrated in Figure 7.2.

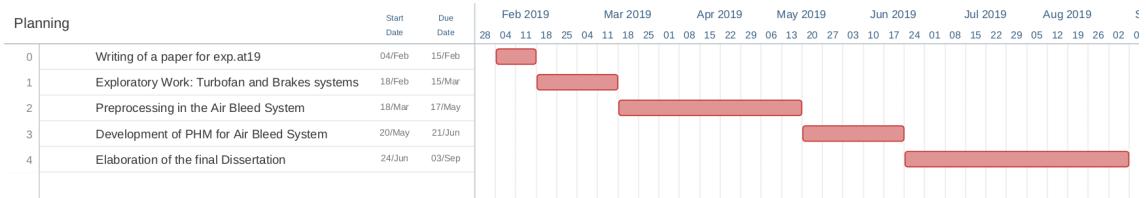


Figure 7.2: Actual Work Performed

Comparing the planning proposed (Figure 7.1) with the real one (Figure 7.2), there were some deviations.

The main reasons for these deviations were regarding the date of delivery of the data from the Air Bleed system.

After writing the paper for the exp.at'19 Conference (according to the planning), the work regarding the exploratory case studies was performed until the Air Bleed data became available.

The two case studies, although they correspond to different aircraft systems with different data structure and organization, were useful for testing different methods for the HI computation and the RUL estimation. Nevertheless, this preliminary work was not part of the main contribution developed in the Thesis.

After the Air Bleed data became available, all the focus shifted to this system, as it is a more complex and challenging case scenario. Due to the fact that the data characteristics were different from the previous datasets worked on, no direct application of the HI methods was made from the previous work. Thus, a new formulation was created and tested based on the provided information from the Bleed system.

The delay regarding the delivery of the Air Bleed data caused a delay in the next planned tasks. Nevertheless, although the delays, the initial planning was successfully completed.

Risks Assessment

In the intermediate part of this work, the identification of risks that could have an impact on this work was performed. The three identified risks and expected impact are represented in the following table:

Table 7.1: Impact & Probability Analysis

| Impact \ Probability | 1 - Minimal | 2 - Minor | 3 - Moderate | 4 - Significant | 5 - Severe |
|----------------------|-------------|-----------|--------------|-----------------|------------|
| 5 - Very High | | | | | |
| 4 - High | | | Risk 2 | | |
| 3 - Medium | | | | | |
| 2 - Low | | | | Risk 1 | Risk 3 |
| 1 - Very Low | | | | | |

Based on the work evolution, the identified risks had the following impact:

1. Risk 1 - Unavailability of data:

As a ReMAP partner provided the expected data, that is, the Air Bleed system dataset, this risk hadn't significant impact in this work.

Nevertheless, as the data arrival was later than what was expected, some arrangements and prioritization were performed according to the established plan. Thus, this risk did not have much relevancy and is not considered a risk for the future work.

2. Risk 2 - High execution time of the methods used:

This risk had an impact, specially in the later part of this work. The solution found was to setup and use a server which allowed the execution of parallel work and thus speeding up the execution time. Therefore, this risk was successfully mitigated.

3. Risk 3 - Lack of quality of the data in the new dataset:

After an initial analysis of the Air Bleed dataset, and according to the feedback provided by the ReMAP partners, this risk hadn't any impact on this work as the data reflected the system condition. Furthermore, the combination and correlation between the different data sources and the results obtained sustain this affirmation. This way, this risk did not materialize in this work, thus no mitigation action was required.

With regards to the impact of the identified risks on this work, they had no negative influence. **Risk 1** and **Risk 3** did not materialized during this work, and the **Risk 2**, which actually interfered with the work, was rapidly mitigated through a mitigation plan already prepared.

Future Work

The PHM methodology developed in this work achieved interesting results, that were positively received by the client, in this case a ReMAP partner.

Regarding future work, although the promising results, the HI formulation can be improved in order to take into account other factors, like the meteorological conditions and flight turbulence, that also have an impact in the system conditions. Also, better parameters tuning may be performed in order to more accurately detect the situations of extra, normal or lesser degradation.

Regarding the prognosis, different distance metrics, other than the Euclidean distance, may be suited and interesting to be applied. Additionally, more training trajectories should be identified and analyzed in order to extend the set of training trajectories.

In order to test the general applicability of the proposed PHM methodology for the diagnosis and prognosis of the system condition, it should be applied and tested in different aircraft systems. In the future, more data extracted from other aircraft systems will be provided within the ReMAP context. Particularly, data extracted from a new system, the Cabin Air Conditioning and Temperature Control System, is going to be provided soon, which will be used for testing and validating the approach proposed in this work.

This will help testify and verify if the HI formulation and its prediction are also valid in other systems, other than the Air Bleed system.

At the end of this work, the developed PHM methodology was successfully implemented. Using the data retrieved from the aircraft sensors it diagnoses and predicts the system health behavior. This included different tasks, as preprocessing, feature selection, HI computation and RUL estimation.

The results obtained were interesting and very positive and serve as a promising approach to be used in different aircraft scenarios, namely in aircraft maintenance.

Based on the results obtained in a real aircraft system (Air Bleed system) and the feedback gathered from the different partners of the project, the developed approach and, in particular the formulation of the HI, is suitable and valid for being used in the aircraft field for assessing and predicting the aircraft systems health condition.

From the point of view of the ReMAP project, this work fulfilled the intended goals and instigate the interest and approval of the other partners. Furthermore, the presented work and results also arouse appreciation and interest from ReMAP, due to the innovation and positive results obtained.

All the feedback stimulates and encourages the continuation of the work in the future.

This page is intentionally left blank.

Appendices

Appendix A

Contribution and involvement in the ReMAP

As mentioned before this work is enrolled in the ReMAP project, that aims to develop a new approach to perform aircraft maintenance, by following a CBM approach.

As this work is enclosed in the Work Package 5 (WP5), there were meetings between the partners responsible for WP5 every two weeks, where the UC also participates. In these meetings relevant aspects regarding WP5 work were discussed, as well as, follow ups and punctual presentations regarding the partners' work. UC, as an involved partner, made numerous interventions, as well as, presentations regarding this work evolution.

Furthermore, UC attended the general meetings between all the partners. These occur every 6 months, they took place in Coimbra (Portugal) and Patras (Greece). At these meetings, the purpose was to present the work progression over time and discuss certain topics of the projects. In addition, valuable feedback and interesting discussions took place at these meetings, with the partners also working with the provided data.

These discussions were important in order to have validation or suggestions of improvement regarding the approaches followed in this work. The partners' opinions were considered valuable and relevant due to their expert knowledge in the aircraft field.

In addition, in July, the partners responsible for the WP5 (including the UC) attended a meeting with the goal of discussing the work developed in the presence of aircraft engineers. This reunion was very important as it created an opportunity for discussing the developed approach in this work with the ReMAP aircraft engineers, who are experts in the analyzed system.

As a result, the HI formulation created was understood and positively received as an innovative approach by the aircraft engineers, for detecting the health condition of the Air Bleed System. This output was valuable as it served as a validation of the developed approach for the diagnosis of the HI and its prognosis, expressed in the RUL value.

Moreover, the reunions with specific partners, like Onera and Embraer, also turned out to be useful, due to the their practical context, as more attention was given to the details and a more closer analysis was performed regarding the implemented methodologies.

Some of these meetings occurred in LIIS (Laboratory of Industrial Informatics and Systems) in the Department of Informatics Engineering (DEI), where, apart from the ReMAP contribution, other contributions, in different contexts, were made, namely the

Appendix A

publication of a paper regarding the monitoring of the energy and water consumption in a domestic scenario, entitled “*Application for Monitoring and Prediction of Energy and Water Consumption in Domestic Cyber-Physical Systems*” [171].

Appendix B

RUL approaches applied in the Turbofan Engine dataset

This Chapter aims to present and detail the three approaches reproduced, according to the State of the Art, and applied in the Turbofan Engine dataset. Furthermore, the details regarding the choice of the techniques and parameters were made based in the authors suggestions.

B.1 Similarity based Prognostics Approach

This approach [17] was selected because it was the implementation that better predicted the RULs of the dataset in the PHM08 Challenge Competition.

This method has two main stages: training and testing. Figure B.1 shows the generic steps executed in each stage.

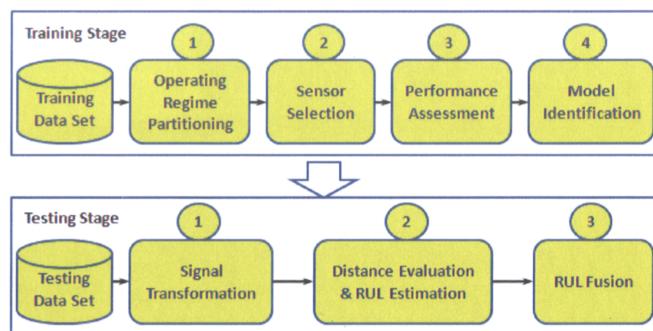


Figure B.1: Steps taken in each stage (from *A Similarity-Based Prognostics Approach for Engineered Systems*)

B.1.1 Training

Operation Regime Partitioning

In this step some data preprocessing was performed, before using the data.

As referred to before, the three Operational settings in each cycle can be different. Analyzing the overall sensor data, it is not possible to identify a clear trend without dividing sensor data by the different combinations of the Operational settings. Therefore, it was required to identify to which combination of Operational settings each cycle belongs to. This was achieved using the k-means method, as the number of combinations is known by the dataset description.

The cycle values C in the training dataset were also modified. The data from each training unit was rearranged so that the last unit cycle corresponded to 0, to accomplish this the following formula was applied to all training data: $C^{adj} = C - \text{Unit Life}$.

Sensor Selection

Next, feature selection was performed with the objective of choosing the most relevant features/sensors in estimating the RUL. This selection was made by observing the behavior of each of the 21 sensors over time. If a clear and continuous trend was found for a specific sensor, that one, should be selected. Otherwise, if some inconsistent trends were displayed it might indicate that that sensor is not useful in predicting the RULs and so it is discarded. At the end, a set of important features should be used during the training of the model. The set of sensors used in this technique (and in the other 2 approaches) were: 2, 3, 4, 7, 11, 12 and 15.

Performance Assessment

In the Performance Assessment step the features extracted from the data were fused in order to produce the HI values, which reflect the system health condition. These were calculated through linear regression models, more specifically, a regression model per each operational regime was generated. The models were fitted using filtered training data specific for each regime.

The linear regression formulation used was the following:

$$y = a + \sum_{i=1}^N \beta_i x_i + \varepsilon \quad (\text{B.1})$$

where $x = (x_1, x_2, \dots, x_N)$ is a feature vector of dimension N , $(a, \beta_1, \beta_2, \dots, \beta_N)$ are the model parameters and ε refers to the noise term. The y is a health indicator and is binary: it has the value of 0 if, in the training unit, the cycle is near the end of life; and it has the value of 1 if, in the training unit, the cycle corresponds to an early life of the units. The remaining cycles are not in the extremes of the units' life time and so they were not considered for the fitting of the models. The y values can be calculated, using the following

formulation:

$$y = \begin{cases} 0, & C_i^{adj} > C_{max} \\ 1, & C_i^{adj} < C_{min} \end{cases} \quad (\text{B.2})$$

where i is the number of the cycle in the training unit. Following the authors implementation, C_{max} has the value of -5 and C_{min} the value of -300.

Having now one regression model per each regime, a HI time series can be calculated for each unit according to the operational regime of each cycle. The HI time series, obtained during the implementation, for training unit 1 is illustrated in Figure B.2.

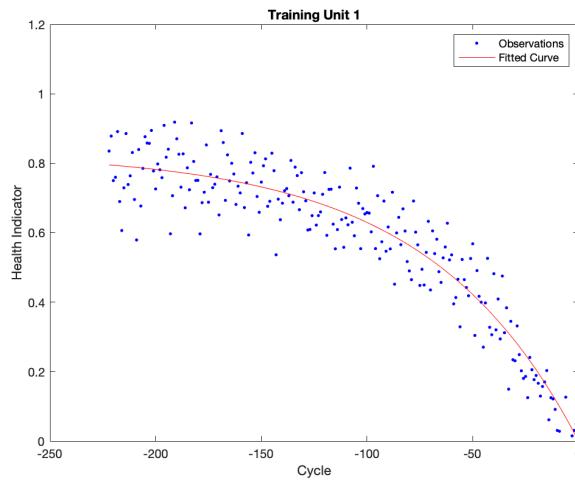


Figure B.2: HI time series obtained for training unit 1

Model Identification

In this step an exponential regression model, M_i , was created for each training unit i . These models simulate the different degradation behavior of each training unit and were fitted using the respective unit training data.

The exponential regression formulation used was the following:

$$y = a(e^{bC^{adj}+c} - e^c) + \varepsilon \quad (\text{B.3})$$

where C^{adj} is the cycle index, (a, b, c) are the model parameters of the training unit i , y is the HI and ε is the noise term. Figure 5.6 shows the degradation curves, obtained from the exponential models, regarding the first 10 training units.

The library of exponential models created in this step was then used to calculate the RULs of the test units.

B.1.2 Testing

In the testing stage, the set of models representing the degradation behaviors of the training units, were used to infer the RULs of each test unit.

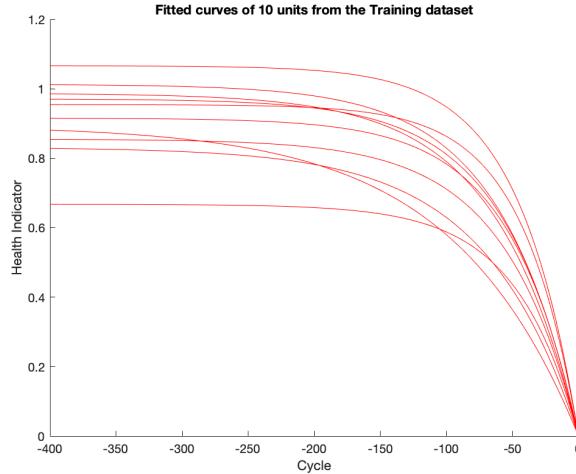


Figure B.3: HI curves produced for the first 10 training units

Signal Transformation

Before calculating the RULs, the test data was preprocessed, similarly to the preprocess applied to the training data.

For each unit in the testing data set, the selected sensors data were classified by operating regimes, transformed by the linear models for performance assessment obtained during training, and merged to obtain an HI sequence Y [17].

Distance Evaluation & RUL Estimation

In this step, a moving average method was applied to the HI series.

Next, the distance between a test unit and each of the training models, with respect to r consecutive cycles, was calculated applying an Euclidean Distance, as represented in following formula:

$$d(\tau, Y, M_i) = \sum_{j=1}^r (Y_j - f_i(-\tau - r + j))^2 / \sigma_i^2 \quad (\text{B.4})$$

where τ represents the number of cycles that sequence Y of the test unit is left shifted from the last cycle of model M_i , Y is the HI sequence of the test unit, f_i is the value of the exponential model of training unit i , r is the number of consecutive cycles analyzed from the test unit and the σ_i^2 is the variance associated with the fitting of model M_i .

In order to find the cycle from which the test unit fits the most in the training unit, the value of τ should be tested in the following interval: $0 \leq \tau \leq T_i - r$.

For each model M_i in the library, the RUL estimation for a test unit i is:

$$RUL_i = \arg \min_{\tau} D_i \quad (\text{B.5})$$

Where the distance score D_i regarding that training unit i is:

$$D_i = \min_{\tau} d_i(\tau, Y, M_i) \quad (\text{B.6})$$

As a result, a pool of RULs was obtained for each test unit.

RUL Fusion

The last step is to merge the pool of RULs in one final RUL for each test unit. It is advised by the authors to, first, do some filtering in the pool of RULs. As such, the authors presented some techniques for filtering:

- **Candidate Selection:** Some candidates of the RUL pool can be discarded, namely the ones whose distance score is 25% higher than the smallest distance score presented in the pool.
- **Outlier Removal:** The RULs particularly long or particularly short should be removed. With the purpose, the following constraint can be applied to the RULs:

$$Q_{0.5} - 3(Q_{0.5} - Q_{0.25}) < RUL_i < Q_{0.5} + 2(Q_{0.75} - Q_{0.5}) \quad (\text{B.7})$$

, in which $Q_{0.25}$, $Q_{0.5}$ and $Q_{0.75}$ are the first, second and third quartiles, respectively.

Finally for the RUL determination, the following weighted average was performed:

$$RUL = (13/23).min_i(RUL_i) + (10/23).max_i(RUL_i) \quad (\text{B.8})$$

Using this formula the author aimed to prioritize earlier predictions rather than later predictions.

B.2 Neural Network Based Approach

Following the same intention of comparing different approaches for the same dataset, a Neural Network based method was implemented.

The Neural Network consisted in a Multilayer Perceptron (MLP) and its design was inspired in an approach submitted for the competition [172].

B.2.1 Data preprocessing

Before training the MLP some preprocessing steps were performed, in accordance with the authors idea.

Data normalization

The data from the training and test datasets was normalized in order to uniformize the range of values. The formula applied was the following:

$$N(x^d) = \frac{x^d - \mu^d}{\sigma^d} \quad (\text{B.9})$$

where x^d corresponds to the original values for sensor d , and μ^d and σ^d represents the mean and standard deviation of the values for sensor d , respectively.

Data smoothing

In order to smooth the data from the datasets, a moving average was applied, with a window length of five. In Figure B.4 it is possible to see the difference between the original and the smoothing data.

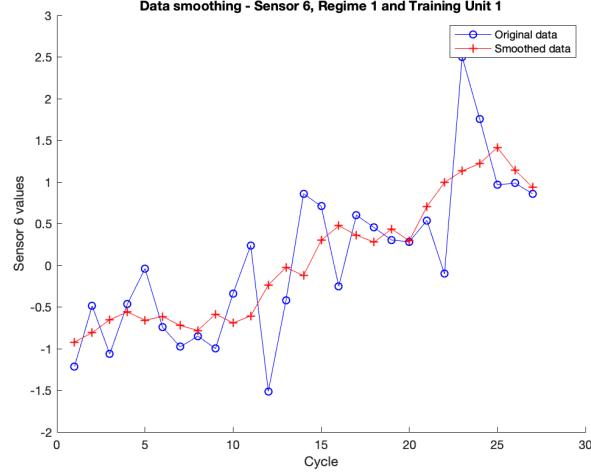


Figure B.4: Smoothing performed on Sensor 6 of training unit 1

Historical Run Indicator

The author suggested the addition of six features to the dataset, representing the total number of cycles spent in each operational regime since the beginning of the run. This could enhance the knowledge of the Neural Network. Figure B.5 illustrates the evolution of the run indicators in the training unit 1.

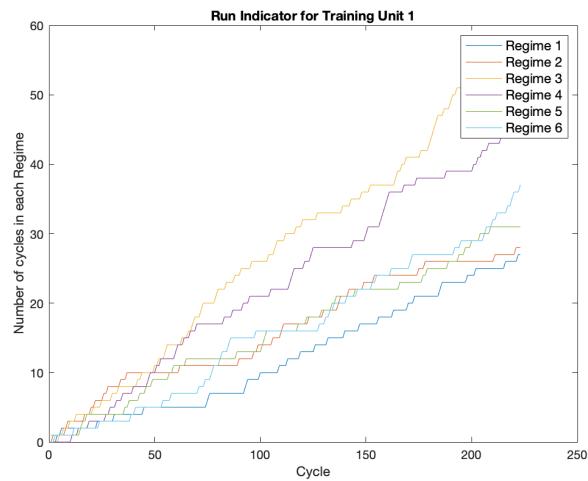


Figure B.5: Evolution of the Run Indicators for the 6 Operational Regimes

Calculation of the Health Index of the training dataset

Using the regression models created previously and the operating regime of each training cycle, a HI value was calculated for each cycle. In order to smooth the values due the presence of noise, a moving average with window length of 5 was used. The window length value was chosen by experimenting and testing different values and analyzing their impact and smoothness in the data.

In Figure B.6 it is possible to observe the HI sequence obtained for the training unit 1 and the respective smoothing.

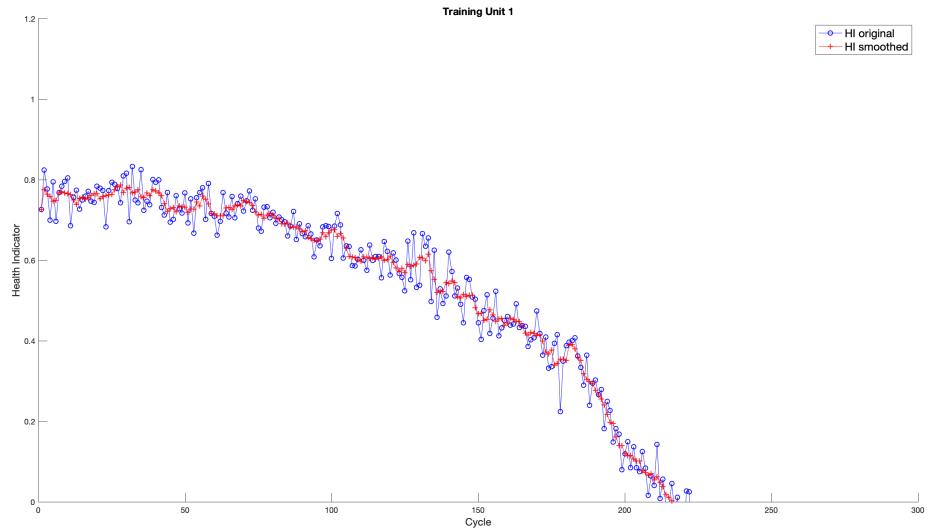


Figure B.6: HI sequence obtained for training unit 1

B.2.2 Training

In the training stage, using the regression models obtained in the previous implementation, for each regime, a MLP was trained using the training dataset. Its architecture was based in the authors suggestion.

The MLP was generated with 16 **inputs** and **two layers**: a hidden layer and an output layer. The hidden layer had 40 neurons and a *tan-sigmoid* as the activation function. The output layer used a *linear transfer* function as the activation function.

The training data, that is, the MLP *input*, had the following columns:

- 3 Operating Regimes
- 7 Sensors selected
- 6 Run Indicators

The MLP *target*, was the HI sequence calculated previously.

The overall MLP structure is illustrated in the Figure 5.5.

B.2.3 Testing

On the completion of the MLP training, the test dataset was simulated in the trained net. The output corresponded to the HI values of each cycle in the test data.

For each test unit, a moving average filter with window length of 7 was applied in order to smooth the HI values, reducing the noise associated. The reason of the choice of the window length was the same as before.

In order to calculate the RUL, the next step was extrapolating, using the most suited polynomial, the HI sequence for each test unit, until the point it reaches the 0. This point corresponds to End of Life of that unit. The order of the polynomial was chosen by graphic analysis.

The RUL for that unit corresponds to the difference between the cycle corresponding to the engine EoL and the last cycle presented on that unit test history.

Figure 5.8 shows the extrapolation achieved for the test unit 13.

B.3 Extrapolation based Prognostics Approach

In order to experiment other category of prognostics methodologies, an Extrapolation based method was implemented. The implementation is, again, divided in two main stages: Training and Testing.

B.3.1 Training

With the objective of comparing this approach with the one implemented previously, the training steps are identical. The training include **Operation Regime Partitioning**, **Sensor Selection** and **Performance Assessment**. In the end, a regression model per regime should be created and they should be the same as the ones generated in the previous implementation.

B.3.2 Testing

In the testing phase, three steps need to be achieved: **Signal Transformation**, **HIs calculation** and **RUL extrapolation**.

Signal Transformation

Before calculating the RULs by extrapolation, the test data was preprocessed, similarly to the preprocess applied to the training data. This preprocessing includes **Operation Regime Partitioning** and **Sensor Selection**.

HIs calculation

In this step the HI values were calculated regarding the test dataset. Based on the regression models created previously and on the operating regime of each cycle in the test dataset, a HI sequence was obtained for each test unit.

In order to remove noise, a moving average filter with window length of 7 was applied to the HI sequence of each unit. The length of the window determined was based on the attenuation in the HI values. With smaller windows, the smoothing was not significant and thus a window length of 7 was applied.

RUL Extrapolation

For each test unit, a polynomial fitting curve of the HI values was generated and the RUL regarding that test unit was extrapolated. The choice of the polynomial order that better fits the HI values was done, again, by analyzing the graph with the HI values and respective polynomial curves (from the 1st order to the 4th) and concluding which order was more appropriated to that specific values.

In Figure 5.9 the RUL extrapolation performed for test unit 13, is demonstrated. As can be observed, the RUL corresponds to the difference between the cycle corresponding to the End of Life (EoL) of that unit and the last cycle presented on that unit history.

Appendix C

Sensors values of a case scenario considered normal

The 5 Figures below illustrates the 5 sensors data for a interval considered normal, in terms of degradation. In this interval no FDE were triggered, no removals were performed and, as seen in the Figures, the sensors don't present significant anomalies and have regular patterns in each flight.

The plotted data corresponds to the time interval 2017/6/13 - 2017/6/26 of the Air Bleed system 4, regarding the aircraft 04388298.

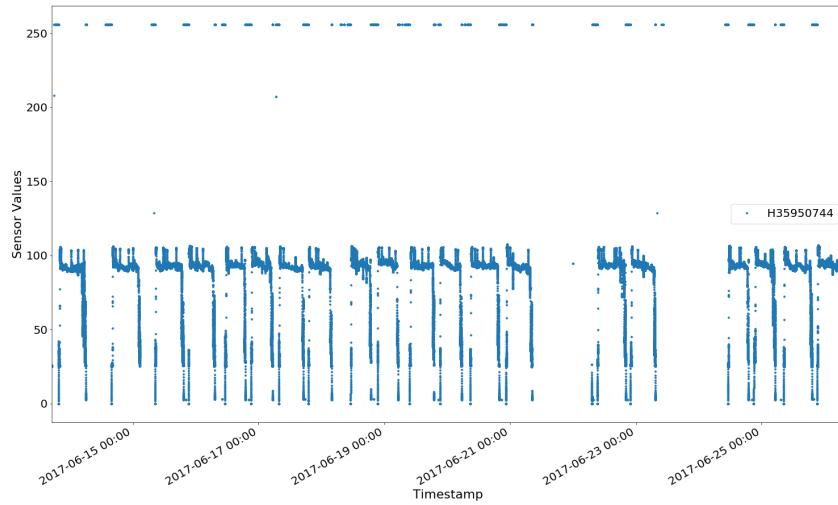


Figure C.1: Sensor values of Sensor 1 - column H35950744

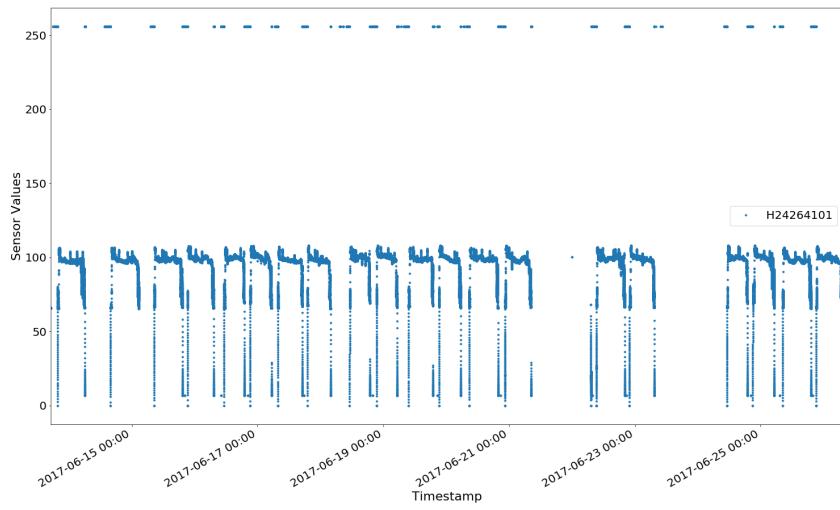


Figure C.2: Sensor values of Sensor 2 - column H24264101

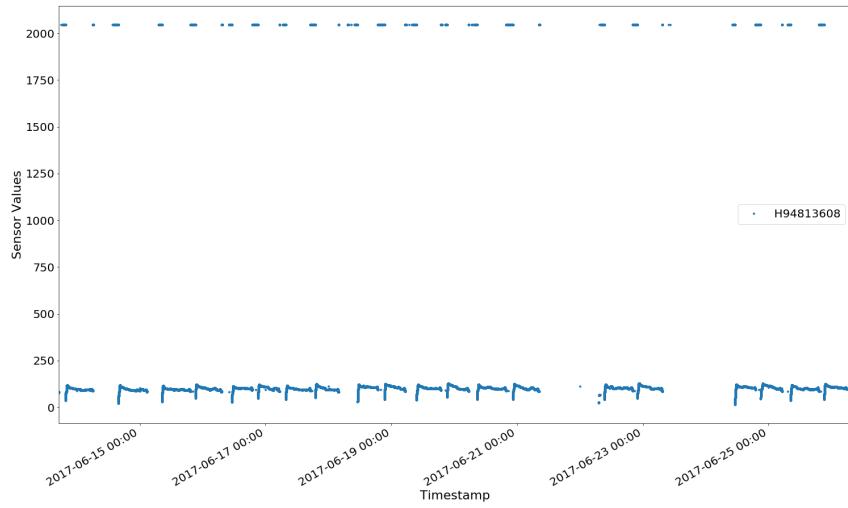


Figure C.3: Sensor values of Sensor 3 - column H94813608

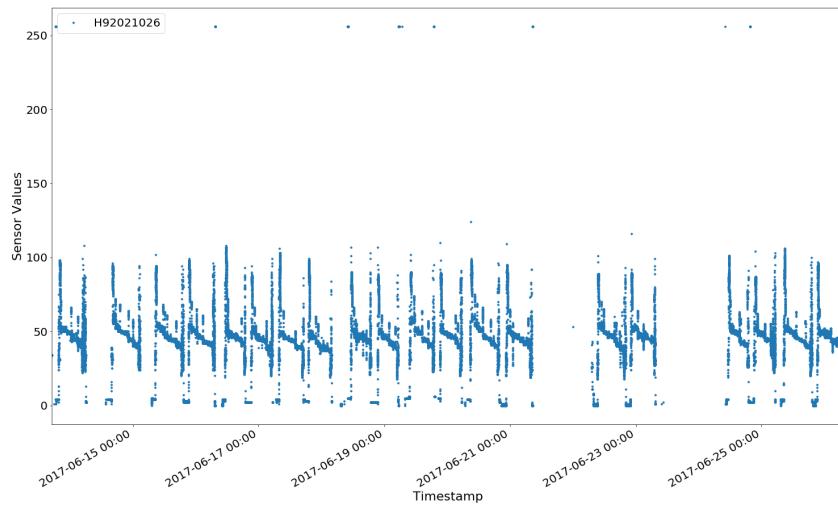


Figure C.4: Sensor values of Sensor 4 - column H92021026

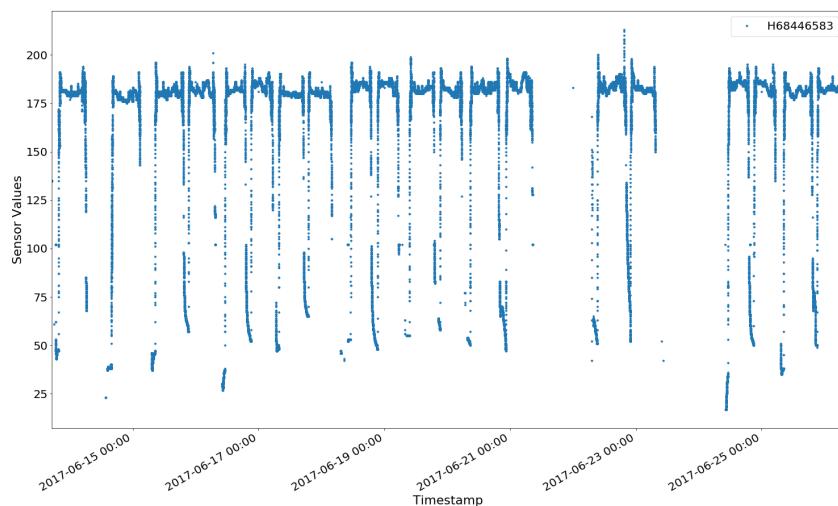


Figure C.5: Sensor values of Sensor 5 - column H68446583

This page is intentionally left blank.

References

- [1] ReMAP - Real-time Condition-based Maintenance for Adaptive Aircraft Maintenance Planning. <https://h2020-remap.eu/>. Accessed: 2019-5-26.
- [2] Tim Edwards. Personal correspondence with Tom Edwards. United Airlines.
- [3] National Research Council. *New Materials for Next-Generation Commercial Transports*. The National Academies Press, Washington, DC, 1996.
- [4] Premaratne Samaranayake and Senevi Kiridena. Aircraft maintenance planning and scheduling : An integrated framework. *Journal of Quality in Maintenance Engineering*, 18:432–453, 2012.
- [5] Serdar Dalkilic. Improving aircraft safety and reliability by aircraft maintenance technician training. *Engineering Failure Analysis*, 82(May):687–694, 2017.
- [6] Premaratne Samaranayake. Current Practices and Problem Areas in Aircraft Maintenance Planning and Scheduling Interfaced / Integrated System Perspective. *Asia-Pacific Industrial Engineering and Management Systems Conference*, 2006.
- [7] William Rankin, Rebecca Hibit, Jerry Allen, and Robert Sargent. Development and evaluation of the maintenance error decision aid (meda) process. *International Journal of Industrial Ergonomics*, 26(2):261 – 276, 2000.
- [8] Safety Report. *IATA - Health safety report 2017*. IATA, 2017.
- [9] Airbus Industry. *A Statistical Analysis of Commercial Aviation Accidents, 1958-2017*. Airbus, 2018.
- [10] R.C. Graeber and D.A Marx. Reducing human error in aviation maintenance operations. *Flight Safety Foundation 46th Annual International Air safety Seminar*, 1993.
- [11] James. Reason. *Managing the risks of organizational accidents / James Reason*. Ashgate Aldershot, 1997.
- [12] Online simulation of methods to predict the remaining useful lifetime of aircraft components. In *5th Experiment@International Conference (exp.at'19)*. IEEE, 2019.
- [13] Web-based tool for predicting the remaining useful lifetime of aircraft components. In *5th Experiment@International Conference (exp.at'19)*. IEEE, 2019.
- [14] International Journal of Online and Biomedical Engineering (iJOE). <https://www.online-journals.org/index.php/i-joe>. Accessed: 2019-08-20.
- [15] D. Azevedo, B. Ribeiro, and Alberto Cardoso. Benchmark of methods applied to PHM08 Challenge Dataset. Technical report, University of Coimbra, 2018.

- [16] Yaguo Lei, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, and Jing Lin. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mechanical Systems and Signal Processing*, 104:799–834, 2018.
- [17] T. Wang, Jianbo Yu, D. Siegel, and J. Lee. A similarity-based prognostics approach for remaining useful life estimation of engineered systems. In *2008 International Conference on Prognostics and Health Management*, pages 1–6, Oct 2008.
- [18] Naipeng Li, Yaguo Lei, Jing Lin, and Steven X. Ding. An Improved Exponential Model for Predicting Remaining Useful Life of Rolling Element Bearings. *IEEE Transactions on Industrial Electronics*, 62(12):7762–7773, 2015.
- [19] A. Saxena. Prognostics, the science of prediction. *Annual Conference of the PHM Society (PHM2010)*, 2010.
- [20] Jay Lee, Fangji Wu, Wenyu Zhao, Masoud Ghaffari, Linxia Liao, and David Siegel. Prognostics and health management design for rotary machinery systems - Reviews, methodology and applications. *Mechanical Systems and Signal Processing*, 42(1–2):314–334, 2014.
- [21] ATA Codes. <https://www.aviationmaintenancejobs.aero/aircraft-ata-chapters-list>. Accessed: 2019-04-21.
- [22] Flight Cycle definition. <https://www.lawinsider.com/dictionary/flight-cycle>. Accessed: 2019-07-12.
- [23] Valentine Goblet, Nicoletta Fala, and Karen Marais. Identifying phases of flight in general aviation operations. In *15th AIAA Aviation Technology, Integration, and Operations Conference*, 06 2015.
- [24] Flight Phase division. https://www.skybrary.aero/index.php/Flight_Phase_Taxonomy. Accessed: 2019-07-12.
- [25] Basim Al-Najjar and Imad Alsyouf. Selecting the most efficient maintenance approach using fuzzy multiple criteria decision making. *International Journal of Production Economics*, 84(1):85 – 100, 2003.
- [26] Jianhui Luo, M Namburu, Krishna Pattipati, Liu Qiao, M Kawamoto, and S Chigusa. Model-based prognostic techniques [maintenance applications]. In *Proceedings AUTOTESTCON 2003. IEEE Systems Readiness Technology Conference.*, pages 330 – 340, 10 2003.
- [27] Jong Ho Shin and Hong Bae Jun. On condition based maintenance policy. *Journal of Computational Design and Engineering*, 2(2):119–127, 2015.
- [28] Atiq Waliullah Siddiqui and Mohamed Ben-Daya. Reliability centered maintenance. *Handbook of Maintenance Management and Engineering*, 60:397–415, 2009.
- [29] C. J. Bamber, J. M. Sharp, and M. T. Hides. Factors affecting successful implementation of total productive maintenance: A UK manufacturing case study perspective. *Journal of Quality in Maintenance Engineering*, 5(3):162–181, 1999.
- [30] J. Van den Bergh, P. De Bruecker, J. Belien, and J. Peeters. Aircraft maintenance operations: State of the art. *FEB@Brussel*, 2013.
- [31] W. Clarke, L. Johnson, Nemhauser L., and Z. Zhu. The aircraft rotation problem. *Annals of Operations Research*, 1997.

- [32] Jie Cai, Jiawei Luo, Shulin Wang, and Sheng Yang. Feature selection in machine learning: A new perspective. *Neurocomputing*, 300:70–79, 2018.
- [33] Jianyu Miao and Lingfeng Niu. A Survey on Feature Selection. *Procedia Computer Science*, 91(Itqm):919–926, 2016.
- [34] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, March 2003.
- [35] Girish Chandrashekhar and Ferat Sahin. A survey on feature selection methods. *Computers & Electrical Engineering*, 40(1):16 – 28, 2014. 40th-year commemorative issue.
- [36] Jacek Biesiada and Włodzisław Duch. Feature selection for high-dimensional data — a pearson redundancy based filter. In Marek Kurzynski, Edward Puchala, Michał Woźniak, and Andrzej Zolnierek, editors, *Computer Recognition Systems 2*, pages 242–249, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.
- [37] Liam Paninski. Estimation of entropy and mutual information. *Neural Comput.*, 15(6):1191–1253, June 2003.
- [38] Kenji Kira and Larry A. Rendell. A practical approach to feature selection. In Derek Sleeman and Peter Edwards, editors, *Machine Learning Proceedings 1992*, pages 249 – 256. Morgan Kaufmann, San Francisco (CA), 1992.
- [39] Ryan Urbanowicz, Melissa Meeker, William LaCava, Randal Olson, and Jason Moore. Relief-based feature selection: Introduction and review. *Journal of Biomedical Informatics*, 85, 11 2017.
- [40] Sheldon M. Ross. Chapter 11 - analysis of variance. In Sheldon M. Ross, editor, *Introductory Statistics (Third Edition)*, pages 503 – 536. Academic Press, Boston, third edition edition, 2010.
- [41] Kruskal Wallis Test. <http://www.biostathandbook.com/kruskalwallis.html>. Accessed: 2019-07-15.
- [42] Ron Kohavi and George H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273 – 324, 1997. Relevance.
- [43] Naoual El Aboudi and Laila Benhlima. Review on wrapper feature selection approaches. In *2016 International Conference on Engineering MIS (ICEMIS)*, pages 1–5, 09 2016.
- [44] Darrell Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4(2):65–85, Jun 1994.
- [45] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, Nov 1995.
- [46] P. Pudil, J. Novovicova, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, 15(11):1119 – 1125, 1994.
- [47] Avrim L. Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1):245 – 271, 1997. Relevance.
- [48] T.N. Lal, Olivier Chapelle, J Weston, A Elisseeff, I Guyon, S Gunn, Masoud Nikravesh, and L Zadeh. Embedded methods. *Feature Extraction, Foundations and Applications*, 137-165 (2006), 04 2019.

- [49] A. Jovic, K. Brkic, and N. Bogunovic. A review of feature selection methods with applications. In *2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, pages 1200–1205, May 2015.
- [50] Leo Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, October 2001.
- [51] Robert Tibshirani. Regression shrinkage and selection via the lasso: a retrospective. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 73(3):273–282, 2011.
- [52] Valeria Fonti and Eduard N. Belitser. Paper in business analytics feature selection using lasso, 2017.
- [53] Wieslaw Staszewski. Intelligent signal processing for damage detection in composite materials. *Composites Science and Technology*, 62:941–950, 06 2002.
- [54] Imola Fodor. A survey of dimension reduction techniques, 2002.
- [55] Laurens van der Maaten, Eric Postma, and H Herik. Dimensionality reduction: A comparative review. *Journal of Machine Learning Research - JMLR*, 10, 01 2007.
- [56] Herve Abdi and Lynne J. Williams. Principal component analysis. *WIREs Comput. Stat.*, 2(4):433–459, July 2010.
- [57] Basics and Examples of Principal Component Analysis (PCA). https://www.projectrhea.org/rhea/index.php/PCA_Theory_Examples. Accessed: 2019-04-26.
- [58] Jieping Ye. Least squares linear discriminant analysis. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 1087–1093, New York, NY, USA, 2007. ACM.
- [59] Discriminant Analysis. http://www.di.fc.ul.pt/~jpn/r/discriminant_analysis/discriminant_analysis.html#lda-projections-iris-eg. Accessed: 2019-04-26.
- [60] T. Kohonen. The self-organizing map. *Proceedings of the IEEE*, 78(9):1464–1480, Sep. 1990.
- [61] Self-Organizing Maps. <http://www.pitt.edu/~is2470pb/Spring05/FinalProjects/Group1a/tutorial/som.html>. Accessed: 2019-04-26.
- [62] Self Organizing Maps: Fundamentals. <http://www.cs.bham.ac.uk/~jxb/NN/116.pdf>. Accessed: 2019-04-26.
- [63] Juan Burguillo and Bernabe Dorronsoro. Using complex network topologies and self-organizing maps for time series prediction. *Advances in Intelligent Systems and Computing*, 210, 01 2013.
- [64] Liang Guo, Yaguo Lei, Naipeng Li, Tao Yan, and Ningbo Li. Machinery health indicator construction based on convolutional neural networks considering trend burr. *Neurocomputing*, 292:142–150, 2018.
- [65] Dong Wang, Kwok Leung Tsui, and Qiang Miao. Prognostics and Health Management: A Review of Vibration Based Bearing and Gear Health Indicators. *IEEE Access*, 6:665–676, 2017.

- [66] Raffaella Di Sante. Fibre optic sensors for structural health monitoring of aircraft composite structures: Recent advances and applications. *Sensors (Switzerland)*, 15(8):18666–18713, 2015.
- [67] Jiuping Xu, Yusheng Wang, and Lei Xu. PHM-oriented integrated fusion prognostics for aircraft engines based on sensor data. *IEEE Sensors Journal*, 14(4):1124–1132, 2014.
- [68] Pingfeng Wang, Byeng D. Youn, and Chao Hu. A generic probabilistic framework for structural health prognostics and uncertainty management. *Mechanical Systems and Signal Processing*, 28:622–637, 2012.
- [69] Arnaz Malhi, Ruqiang Yan, and Robert X. Gao. Prognosis of defect propagation based on recurrent neural networks. *IEEE Transactions on Instrumentation and Measurement*, 60(3):703–711, 2011.
- [70] Zheng Xin Zhang, Xiao Sheng Si, and Chang Hua Hu. An Age- and State-Dependent Nonlinear Prognostic Model for Degrading Systems. *IEEE Transactions on Reliability*, 64(4):1214–1228, 2015.
- [71] Cody Spiegel Junda Zhu, Thomas Nostrand and B. Morton. Survey of Condition Indicators for Condition Monitoring Systems. *PHM 2014 - Proceedings of the Annual Conference of the Prognostics and Health Management Society 2014*, 2014.
- [72] H. S. Kumar, Srinivasa P. Pai, N. S. Sriram, and G. S. Vijay. Rolling element bearing fault diagnostics: Development of health index. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 231(21):3923–3939, 2017.
- [73] Jing Lin Hai Qiu, Jay Lee and Gang Yu. Robust performance degradation assessment methods for enhanced rolling element bearing prognostics. *Advanced Engineering Informatics*, 17(3-4):127–140, 2003.
- [74] J. Mina and C. Verde. Fault detection using dynamic principal component analysis by average estimation. In *2005 2nd International Conference on Electrical and Electronics Engineering*, pages 374–377, Sep. 2005.
- [75] Achmad Widodo and Bo Suk Yang. Application of relevance vector machine and survival probability to machine degradation assessment. *Expert Systems with Applications*, 38(3):2592–2599, 2011.
- [76] Kamran Javed, Rafael Gouriveau, Noureddine Zerhouni, and Patrick Nectoux. Enabling health monitoring approach based on vibration data for accurate prognostics. *IEEE Transactions on Industrial Electronics*, 62(1):647–656, 2015.
- [77] Weizhong Yan, Hai Qiu, and Naresh Iyer. Feature extraction for bearing prognostics and health management (phm) - a survey (preprint), 01 2008.
- [78] B. R. Nayana and P. Geethanjali. Analysis of statistical time-domain features effectiveness in identification of bearing faults from vibration signal. *IEEE Sensors Journal*, 17(17):5618–5625, Sep. 2017.
- [79] Jianzhong SUN, Chaoyi LI, Cui LIU, Ziwei GONG, and Ronghui WANG. A data-driven health indicator extraction method for aircraft air conditioning system health monitoring. *Chinese Journal of Aeronautics*, 32(2):409 – 416, 2019.

- [80] Salah M. Ali Al-Obaidi, M. Salman Leong, R. I. Raja Hamzah, Ahmed M. Abdellatif, and Mahmoud Danaee. Acoustic emission parameters evaluation in machinery condition monitoring by using the concept of multivariate analysis. *ARPN Journal of Engineering and Applied Sciences*, 11(12):7507–7514, 2016.
- [81] B Abou El Anouar, M Elamrani, B Elkhalil, F Delaunois, and Laboratoire Electronique. A comparative experimental study of ultrasound technique and vibration analysis in detection of bearing defect. *13th Congress of Mechanics*, pages 13–15, 2017.
- [82] Zhenyou Zhang, Yi Wang, and Kesheng Wang. Fault diagnosis and prognosis using wavelet packet decomposition, Fourier transform and artificial neural network. *Journal of Intelligent Manufacturing*, 24(6):1213–1227, 2013.
- [83] Linxia Liao. Discovering prognostic features using genetic programming in remaining useful life prediction. *IEEE Transactions on Industrial Electronics*, 61(5):2464–2472, 2014.
- [84] Bo Wu, Wei Li, and Ming-quan Qiu. Remaining Useful Life Prediction of Bearing with Vibration Signals Based on a Novel Indicator. *Shock and Vibration*, 2017:1–10, 2017.
- [85] Bernd Geropp. Envelope Analysis - A Signal Analysis Technique for Early Detection and Isolation of Machine Faults. *IFAC Proceedings Volumes*, 30(18):977–981, 1997.
- [86] Pavle Boškoski, Matej Gašperin, and Dejan Petelin. Bearing fault prognostics based on signal complexity and Gaussian process models. *PHM 2012 - 2012 IEEE Int. Conf. on Prognostics and Health Management: Enhancing Safety, Efficiency, Availability, and Effectiveness of Systems Through PHM Technology and Application, Conference Program*, 2012.
- [87] Matej Gaperin, Dani Juriić, Pavle Bokoski, and Joef Viintin. Model-based prognostics of gear health using stochastic dynamical models. *Mechanical Systems and Signal Processing*, 25(2):537–548, 2011.
- [88] E. Bechhoefer. Signal processing techniques to improve an acoustic emissions sensor. *Annual Conference of the Prognostics and Health Management Society 2013*, oct 2013.
- [89] R. Polikar, F.C. Morabito, G. Simone, S. Udupa, P. Ramuhalli, and L. Udupa. Feature extraction techniques for ultrasonic signal classification. *International Journal of Applied Electromagnetics and Mechanics*, 15(1-4):291–294, 2018.
- [90] R.X. Gao and R. Yan. Non-stationary signal processing for bearing health monitoring. *International Journal of Manufacturing Research*, 1(1):18, 2006.
- [91] J. Allen. Short term spectral analysis, synthesis, and modification by discrete fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(3):235–238, June 1977.
- [92] Yan Zhang, Baoping Tang, Yan Han, and Lei Deng. Bearing performance degradation assessment based on time-frequency code features and SOM network. *Measurement Science and Technology*, 28(4), 2017.
- [93] Parasuram P. Hariharan and Alexander G. Parlos. Using motors to detect degradation. *World Pumps*, 2011(11):24–29, 2011.

- [94] A. Graps. An Introduction to Wavelets. *IEEE Computational Science & Engineering*, 2:1–18, 1995.
- [95] Zijun Que, Yu Wang, Tommy W. S. Chow, Yi Sun, and Xiaohang Jin. Anomaly Detection and Fault Prognosis for Bearings. *IEEE Transactions on Instrumentation and Measurement*, 65(9):2046–2054, 2016.
- [96] Wei He, Qiang Miao, Michael Azarian, and Michael Pecht. Health monitoring of cooling fan bearings based on wavelet filter. *Mechanical Systems and Signal Processing*, 64-65:149–161, 2015.
- [97] Christian Grosse and H.W. Reinhardt. Signal conditioning in acoustic emission analysis using wavelets. *NDT.net*, 7:1–9, 01 2002.
- [98] Lokenath Debnath. *The Wigner-Ville Distribution and Time-Frequency Signal Analysis BT - Wavelet Transforms and Their Applications*, pages 307–360. Birkhäuser Boston", Boston, MA, 2002.
- [99] Abdenour Soualhi, Kamal Medjaher, and Noureddine Zerhouni. Bearing Health Monitoring based on Hilbert-Huang Transform, Support Vector Machine, and Regression. *IEEE Transactions on Instrumentation and Measurement*, 64(1):52–62, 2015.
- [100] S.J. Loutridis. Damage detection in gear systems using empirical mode decomposition. *Engineering Structures*, 26(12):1833 – 1841, 2004.
- [101] Yu Wang, Yizhen Peng, Yanyang Zi, Xiaohang Jin, and Kwok Leung Tsui. A Two-Stage Data-Driven-Based Prognostic Approach for Bearing Degradation Problem. *IEEE Transactions on Industrial Informatics*, 12(3):924–932, 2016.
- [102] A. Saxena and K. Goebel. PHM08 Challenge DataSet. *NASA Ames Prognostics Data Repository (<http://ti.arc.nasa.gov/project/prognostic-data-repository>)*, NASA Ames Research Center, Moffett Field, CA, 2008.
- [103] Liang Guo, Yaguo Lei, Naipeng Li, Tao Yan, and Ningbo Li. Machinery health indicator construction based on convolutional neural networks considering trend burr. *Neurocomputing*, 292:142–150, 2018.
- [104] NASA Prognostics Data Repository. <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>. Accessed: 2019-2-21.
- [105] Chen Xiongzi, Yu Jinsong, Tang Diyin, and Wang Yingxun. Remaining Useful Life Prognostic Estimation for Aircraft Subsystems or Components : A Review. *IEEE 2011 10th International Conference on Electronic Measurement & Instruments*, 2:94–98, 2011.
- [106] Tom Brotherton. Prognosis of Faults in Gas Turbine Engines. In *2000 IEEE Aerospace Conference. Proceedings (Cat. No.00TH8484)*, pages 163–171, 2000.
- [107] Feng Xue, Piero Bonissone, Anil Varma, Weizhong Yan, Neil Eklund, and Kai Goebel. An instance-based method for remaining useful life estimation for aircraft engines. *Journal of Failure Analysis and Prevention*, 8(2):199–206, Apr 2008.
- [108] Y G Li and P Nilkitsaranont. Gas turbine performance prognostic for condition-based maintenance. *Applied Energy*, 86(10):2152–2161, 2009.
- [109] A Vatani, K Khorasani, and Nader Meskin. Health monitoring and degradation prognostics in gas turbine engines using dynamic neural networks. *ASME Turbo Expo 2015: Turbine Technical Conference and Exposition*, 06 2015.

- [110] Idriss El-thalji and Erkki Jantunen. A summary of fault modelling and predictive health monitoring of rolling element bearings. *Mechanical Systems and Signal Processing*, 60-61:252–272, 2015.
- [111] S. Marble and B. P. Morton. Predicting the remaining life of propulsion system bearings. In *2006 IEEE Aerospace Conference*, pages 8 pp.–, March 2006.
- [112] Nathan Bolander, Hai Qiu, Neil Eklund, Ed Hindle, and Taylor Rosenfeld. Physics-based Remaining Useful Life Prediction for Aircraft Engine Bearing Prognosis. *Annual Conference of the Prognostics and Health Management Society*, pages 1–12, 2009.
- [113] M. Tariq, A. I. Maswood, C. J. Gajanayake, and A. K. Gupta. Aircraft batteries: current trend towards more electric aircraft. *IET Electrical Systems in Transportation*, 7(2):93–103, 2017.
- [114] Todd D Batzel and David C Swanson. Prognostic Health Management of Aircraft Power Generators. *IEEE Transactions on Aerospace and Electronic Systems*, 45(2):473–482, 2009.
- [115] Jie Liu, Abhinav Saxena, Kai Goebel, Bhaskar Saha, and Wilson Wang. An adaptive recurrent neural network for remaining useful life prediction of lithium-ion batteries. *Annual Conference of the Prognostics and Health Management Society, PHM 2010*, 01 2010.
- [116] Datong Liu, Yue Luo, Yu Peng, Xiyuan Peng, and Michael G. Pecht. Lithium-ion battery remaining useful life estimation based on nonlinear ar model combined with degradation feature. *Annual Conference of Prognostics and Health Management Society 2012*, 2012.
- [117] Susana Ferreiro and Aitor Arnaiz. Prognostics applied to aircraft line maintenance: Brake wear prediction based on bayesian networks. *IFAC Proceedings Volumes*, 43(3):146 – 151, 2010. 1st IFAC Workshop on Advanced Maintenance Engineering, Services and Technology.
- [118] Susana Ferreiro, Aitor Arnaiz, Basilio Sierra, and Itziar Irigoien. Application of bayesian networks in prognostics for a new integrated vehicle health management concept. *Expert Systems with Applications*, 39(7):6402 – 6418, 2012.
- [119] Alexandra Coppe, Raphael Haftka, and Nam Kim. Least squares-filtered bayesian updating for remaining useful life estimation. *Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 01 2010.
- [120] Yiwei Wang, Christian Gogu, Nicolas Binaud, and Christian Bes. A model-based prognostics method for fatigue crack growth in fuselage panels. *Chinese Journal of Aeronautics*, 2019.
- [121] Mark Schwabacher and Kai Goebel. A Survey of Artificial Intelligence for Prognostics. The Intelligence Report. *Association for the Advancement of Artificial Intelligence AAAI Fall Symposium 2007*, pages 107–114, 2007.
- [122] Farzaneh Ahmadzadeh and Jan Lundberg. Remaining useful life estimation: review. *International Journal of Systems Assurance Engineering and Management*, 5(4):461–474, 2013.

- [123] Ying Peng, Ming Dong, and Ming Jian Zuo. Current status of machine prognostics in condition-based maintenance: A review. *International Journal of Advanced Manufacturing Technology*, 50(1-4):297–313, 2010.
- [124] Y. Li, S. Billington, C. Zhang, T. Kurfess, S. Danyluk, and S. Liang. Adaptive prognostics for rolling element bearing condition. *Mechanical Systems and Signal Processing*, 13(1):103–113, 1999.
- [125] P Paris, Assistant Director, and F Erdogan. A Critical Analysis of Crack Propagation Laws. *Journal of Basic Engineering*, 85(4):528–533, 1963.
- [126] Jaydeep M. Karandikar, Nam Ho Kim, and Tony L. Schmitz. Prediction of remaining useful life for fatigue-damaged structures using Bayesian inference. *Engineering Fracture Mechanics*, 96:588–605, 2012.
- [127] T. Wang. *Trajectory Similarity Based Prediction for Remaining Useful Life Estimation*. PhD thesis, University of Cincinnati, 2010.
- [128] Wen Ben Jone, Carla Purdy, and Hal Carter. Evaluation of Health Assessment Techniques for Rotating Machinery. *Computer Engineering*, 2009.
- [129] Kwok L. Tsui, Nan Chen, Qiang Zhou, Yizhen Hai, and Wenbin Wang. Prognostics and health management: A review on data driven approaches. *Mathematical Problems in Engineering*, 2015, 2015.
- [130] Pin Lim, Chi Keong Goh, Kay Chen Tan, and Partha Dutta. Estimation of Remaining Useful Life Based on Switching Kalman Filter Neural Network Ensemble. *PHM Society*, pages 1–8, 2014.
- [131] C. S. Byington, M. Watson, and D. Edwards. Data-driven neural network methodology to remaining life predictions for aircraft actuator components. In *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)*, volume 6, pages 3581–3589 Vol.6, March 2004.
- [132] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter, 1995.
- [133] J. Z. Sikorska, M. Hodkiewicz, and L. Ma. Prognostic modelling options for remaining useful life estimation by industry. *Mechanical Systems and Signal Processing*, 25(5):1803–1836, 2011.
- [134] Man Shan Kan, Andy C.C. Tan, and Joseph Mathew. A review on prognostic techniques for non-stationary and non-linear rotating systems. *Mechanical Systems and Signal Processing*, 62:1–20, 2015.
- [135] Yimin Fan and C. James Li. Diagnostic rule extraction from trained feedforward neural networks. *Mechanical Systems and Signal Processing*, 16(6):1073–1081, 2002.
- [136] Zhang Jian and Xuewu Wang. The application of feed-forward neural network for the x-ray image fusion. *Journal of Physics: Conference Series*, 312, 09 2011.
- [137] J. A. Leonard and M. A. Kramer. Radial basis function networks for classifying process faults. *IEEE Control Systems Magazine*, 11(3):31–38, April 1991.
- [138] A. Graves, A. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, May 2013.

- [139] Ramon Quiza and J Davim. Computational modeling of machining systems, 01 2009.
- [140] David McSherry. Strategic Induction of Decision Trees BT - Research and Development in Expert Systems XV. *Research and Development in Expert Systems XV*, 1999.
- [141] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, Mar 1986.
- [142] Lior Rokach and Oded Maimon. *Data Mining With Decision Trees: Theory and Applications*. World Scientific Publishing Co., Inc., River Edge, NJ, USA, 2nd edition, 2014.
- [143] Peter Harrington. *Machine Learning in Action*. Manning Publications Co., Greenwich, CT, USA, 2012.
- [144] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, Sep 1995.
- [145] Santhoshkumar Rajaram and M Kalaiselvi Geetha. Recognition of emotions from human activity using step feature. *International Journal of Engineering Science Invention (IJESI)*, NCIOT:88–97, 09 2018.
- [146] Stephen Marsland. *Machine Learning: An Algorithmic Perspective, Second Edition*. Chapman & Hall/CRC, 2nd edition, 2014.
- [147] Bernhard E. Boser, Isabelle M. Guyon, and Vladimir N. Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 144–152, New York, NY, USA, 1992. ACM.
- [148] Kurt Hornik, David Meyer, and Alexandros Karatzoglou. Support vector machines in r. *Journal of Statistical Software*, 15, 04 2006.
- [149] Avinanta Tarigan, Dewi Agushinta R., Adang Suhendra, and Fikri Budiman. Determination of svm-rbf kernel space parameter to optimize accuracy value of indonesian batik images classification. *Journal of Computer Science*, 13:590–599, 11 2017.
- [150] Erdal Kayacan and Mojtaba Ahmadieh Khanesar. Chapter 2 - fundamentals of type-1 fuzzy logic theory. In Erdal Kayacan and Mojtaba Ahmadieh Khanesar, editors, *Fuzzy Neural Networks for Real Time Control Applications*, pages 13 – 24. Butterworth-Heinemann, 2016.
- [151] Piyush Singhala, Dhrumil Shah, and Bhavikkumar Patel. Temperature control using fuzzy logic. *International Journal of Instrumentation and Control Systems*, 4, 02 2014.
- [152] Artificial Intelligence - Fuzzy Logic Systems. https://www.tutorialspoint.com/artificial_intelligence/artificial_intelligence_fuzzy_logic_systems.htm. Accessed: 2019-04-08.
- [153] A. Talon and C. Curt. Selection of appropriate defuzzification methods: Application to the assessment of dam performance. *Expert Systems with Applications*, 70:160 – 174, 2017.
- [154] Werner Van Leekwijck and Etienne E. Kerre. Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, 108(2):159 – 178, 1999.

- [155] E. Van Broekhoven and B. De Baets. A comparison of three methods for computing the center of gravity defuzzification. In *2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542)*, volume 3, pages 1537–1542 vol.3, July 2004.
- [156] Aarthi Chandramohan, M. V. C. Rao, and M. Senthil. Arumugam. Two new and useful defuzzification methods based on root mean square value. *Soft Computing*, 10(11):1047–1059, Sep 2006.
- [157] Farinaz Behrooz, N Mariun, Mohammad Marhaban, Mohd Amran Mohd Radzi, and Abdul Ramli. Review of control techniques for hvac systems - nonlinearity approaches based on fuzzy cognitive maps. *Energies*, 11:495, 02 2018.
- [158] Z. Ghahramani. An introduction to hidden markov models and bayesian networks. *International Journal of Pattern Recognition and Artificial Intelligence*, pages 9–42, 2001.
- [159] Charles M. Grinstead and J. Laurie Snell. *Introduction to Probability*. AMS, 2003.
- [160] L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE ASSP Magazine*, 1986.
- [161] Degirmenci A. An introduction to Hidden Markov Models. *IEEE ASSP Magazine*, pages 1–5, 2013.
- [162] Mark Stamp. A revealing introduction to hidden markov models. *Science*, pages 1–20, 01 2004.
- [163] Emmanuel Ramasso and Abhinav Saxena. Performance Benchmarking and Analysis of Prognostic Methods for CMAPSS Datasets, 2014.
- [164] Sajid Khan, Ayyaz Hussain, Abdul Basit, and Sheeraz Akram. Kruskal-wallis-based computationally efficient feature selection for face recognition. *TheScientificWorld-Journal*, 2014:672630, 05 2014.
- [165] Abhinav Saxena, Member Ieee, Kai Goebel, Don Simon, and Neil Eklund. Damage Propagation Modeling for Aircraft Engine Prognostics. *Proceedings of IEEE International Conference on Prognostics and Health Management*, pages 1–9, 2008.
- [166] Yuan Liu, Dean Frederick, Jonathan DeCastro, Jonathan Litt, and William Chan. User’s Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS): Version 2, 2012.
- [167] Mach Number. <https://www.grc.nasa.gov/www/k-12/airplane/mach.html>. Accessed: 2018-12-5.
- [168] Throttle Resolver Angle. https://en.wikipedia.org/wiki/Thrust_lever. Accessed: 2018-12-5.
- [169] Aircraft Bleed Air Systems. https://www.skybrary.aero/index.php/Aircraft_Bleed_Air_Systems#. Accessed: 2019-06-28.
- [170] Project Jupyter. Jupyter notebook. <https://jupyter.org/>.
- [171] Application for monitoring and prediction of energy and water consumption in domestic cyber-physical systems. In *REV2019 - 16th International Conference on Remote Engineering and Virtual Instrumentation*, 2019.

- [172] A. M. Riad, Hamdy K. Elminir, and Hatem M. Elattar. Evaluation of neural networks in the subject of prognostics as compared to linear regression model. *International Journal of Engineering & Technology*, 10(06):52–58, 2010.