Kevin Martin Wesendrup

A Decision Model for the Selection of Fleet-Based Prognostic Methods

Master Thesis

at the Chair for Information Systems and Supply Chain Management
(Westfälische Wilhelms-Universität, Münster)

Supervisor:	Prof. Dr.-Ing. Bernd Hellingrath
Tutor:	Carolin Wagner, M.Sc.
Presented by:	Kevin Wesendrup Franz-Mülder-Straße 10 48282 Emsdetten +49 157 89254 808 kevin.wesendrup@uni-muenster.de
Date of Submission:	2019-06-28

Content

Figures

Tables

Abbreviations

A	General Accuracy
AI	Artificial Intelligence
ANN	Artificial Neural Network
ARMA	Autoregressive Moving Average
BHM	Bayesian Hierarchical Model
Cat	Categorical
CBM	Condition-based Maintenance
CMAPSS	Commercial Modular Aero-Propulsion System Simulation
CV	Cross Validation
D	Randomly Distributed Estimate
DM	Decision Maker
DT	Decision Tree
E	Explainability
ESN	Echo State Network
FFT	Fleet Feature Type
FSB	Fuzzy Similarity-Based
FT	Fleet Type
GPR	Gaussian Process Regression
HDD	Hard Disk Drive
HDTFSSMM	Homogeneous Discrete-Time Finite-State Semi-Markov Model
He	Heterogeneous
Ho	Homogeneous
I	Interval-estimate
Id	Identical
IS	Information System
kNN	k-Nearest-Neighbor
LED	Light-Emitting Diode
M	Missing Data
MAD	Median Absolute Deviation
MADM	Median Absolute Deviation to Median
MAE	Mean Absolute Error
MC	Monte Carlo
ML	Machine Learning
MLE	Maximum Likelihood Estimation
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
N	Noise
NFS	Neuro-Fuzzy System
NIS	Negative Ideal Solution
NIST	National Institute of Standards and Technology
NN	Neural Network
No	None
Num	Numerical
O	Output Type
OOB	Out-of-Bag
P	Parameter Handling
PIS	Positive Ideal Solution
Pt	Point-estimate
PDF	Probability Density Function

PHM	Prognostics and Health Management
PrHM	Proportional Hazard Model
R	Robustness
RAPP	Risk Analysis and Prognosis of Complex Products
RBF	Radial Basis Function
RF	Random Forest
RMS	Root Mean Square
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RSF	Random Survival Forest
RTF	Run-To-Failure
RVM	Relevance Vector Machine
S	Space Complexity
SBI	Similarity-Based Interpolation
Sem	Semantic
SLR	Structured Literature Review
SMART	Self-Monitoring, Analysis and Reporting Technology
SOM	Self-Organizing Maps
SS	Low Sample Size
SVM	Support Vector Machine
SVR	Support Vector Regression
T	Time Complexity

Symbols

$\lambda_0(t)$	Baseline Function of Cox Regression
γ_x	Coefficient of covariate x (Cox regression)
$Z_1(t)$	Value of covariate x at time t (Cox regression)
x_t	(Predicted) value at time t
a	Autoregressive coefficient (ARMA)
b	Moving average coefficient (ARMA)
ε_t	Gaussian noise term (ARMA)
p	Number of previous signals (ARMA)
q	Number of previous periods (ARMA)
$B(t)$	Brownian motion
σ	Diffusion coefficient (Wiener process) or standard deviation
λ	Drift term (Wiener process) or failure rate (MCDM) or eigenvalue (AHP/ANP)
θ	Basic laws
A	Alternative (MCDM) or pairwise alternative matrix (AHP) or alternative-criteria matrix (TOPSIS)
$\underline{A}$	Set of alternatives
ρ_{ij}	Outcome of ith basic law and jth alternative
$U(\rho)$	Utility of outcome ρ
w	Weight
C	Criterion
U	Eigenvector
$C.R$	Consistency ratio (AHP)
μ	Consistency index (AHP) or mean
r	Random index (AHP)
X	Principal eigenvector matrix (AHP)
D	Priority vector (AHP)
r_{ij}	Normalized value of ith alternative and jth criterion
v_{ij}	Weighted normalized value of ith alternative and jth criterion
$\underline{v_p}$	Positive ideal solution (TOPSIS)
$\underline{v_n}$	Negative ideal solution (TOPSIS)
d	Euclidean distance
s_{ib}	Similarity of ith alternative to PIS
σ^2	Variance
W_{out}	Set of output weights of ESN
s	Similarity of FSB
$MADM$	Median absolute deviation to median ratio
$\bar{X}$	Mean
min	Minimum
max	Maximum
ptp	Peak-to-peak
$skew$	Skewedness
$kurt$	Kurtosis
rms	Root mean square
SN	Sensitivity

1 Introduction

1.1 Motivation

Since the beginning of the industrial revolution until today, the relevance of systems and machines increased manifold. Currently we are in midst of the so-called fourth industrial revolution. Today's era is characterized by automation, mass customization and the internet of things. Thus, a rapid increase in systems complexity can be seen that simultaneously increases the effort to maintain a continuous system functionality, especially in increasingly dynamic environments. A continuous flawless operation of engineered systems is important to keep competitiveness as a digitalized company (Guillén et al. 2016, p. 991).

An advanced and predictive maintenance strategy can increase system reliability and reduce costs (Elattar et al. 2016, p. 127). It encompasses health-oriented design, development and testing of systems and can ensure the flawlessness of a running system through prognostics and health management (PHM). PHM encompasses detection, diagnosis and prognostics. Prognostics or PHM are one of the most challenging engineering disciplines (Elattar et al. 2016, p. 126).

In general, it can be distinguished between different prognostic approaches such as physics-based or data-driven approaches (Elattar et al. 2016, p. 133). Physics-based approaches require a deep understanding of the system and can be costly or even impossible to model for complex systems. In this case, data-driven approaches become the only choice (Wang 2010, p. 2). Provided much data is supplied, technologies such as multivariate statistics, artificial intelligence (AI) or machine learning (ML) have the advantage, that little to no knowledge about the system is required and that hidden patterns can be identified (LeCun et al. 2015, p. 436).

Because data-driven approaches require many data to work, which are often not available for single or new systems, fleet-based approaches can be advantageous, because they can generate a considerable volume of historical health and status data. Using fleets however also comes with drawbacks. Because systems within a fleet are often not identical to the system under investigation, prognostic approaches must account for differences such as different parts, working conditions, usage or environments (Zaidan 2014 p. 2).

Fleets can be distinguished as three general types: identical fleets, homogeneous and heterogeneous fleets (Al-Dahidi et al. 2016, p. 110). However, there are also many further dimensions of fleets (Wagner and Hellingrath 2017). Fleet-based approaches have a high potential to improve data-driven prognostics, but as of today, fleets are insufficiently

researched. While there have been approaches making use of information of other systems of the same fleet (Zaidan 2014), it is not obvious for which type of fleet and context they are suitable in general. Because there exists no work that classifies prominent prognostic algorithms that make use of fleets, there is no decision support on how to choose the best approach for a given fleet and business context. This makes it especially hard for practitioners, which want to implement PHM using fleets.

1.2 Research Objectives

To tackle above-mentioned research gaps, this thesis addresses the following main goal: Rest of useful life (RUL) fleet prognostic approaches should be synthesized and a tested, tangible and actionable decision model for the selection of prognostic methods be constructed. The main goal is achieved through three research objectives.

The first objective is the exploration of the current state-the-art in fleet PHM by carrying out a structured literature review (SLR). Pivotal fleet-data-based approaches for fault prognostics should be identified, the general class of approaches (such as Bayesian, Neural Networks, etc.) and their aptness to specific fleet types be analyzed. Also, it should be emphasized, how fleet-based approaches work differently than their non-fleet-based counterparts, by identifying their generalized operation steps.

Secondly, a decision support model for the selection of the best-fit method for a given prognostics scenario is constructed. As a prerequisite it is examined, what criteria are relevant for the selection of prognostic approaches. Subsequently, these approaches and the decision criteria, which have been identified, should be structured into a selection support model. To choose the optimal approach only the decision criteria must be known, and a scholar or practitioner should be able to derive the appropriate approach for his/her dataset and the prognostics context. This objective also facilitates a synthesis and structuring of the current literature on fleet-based prognostics.

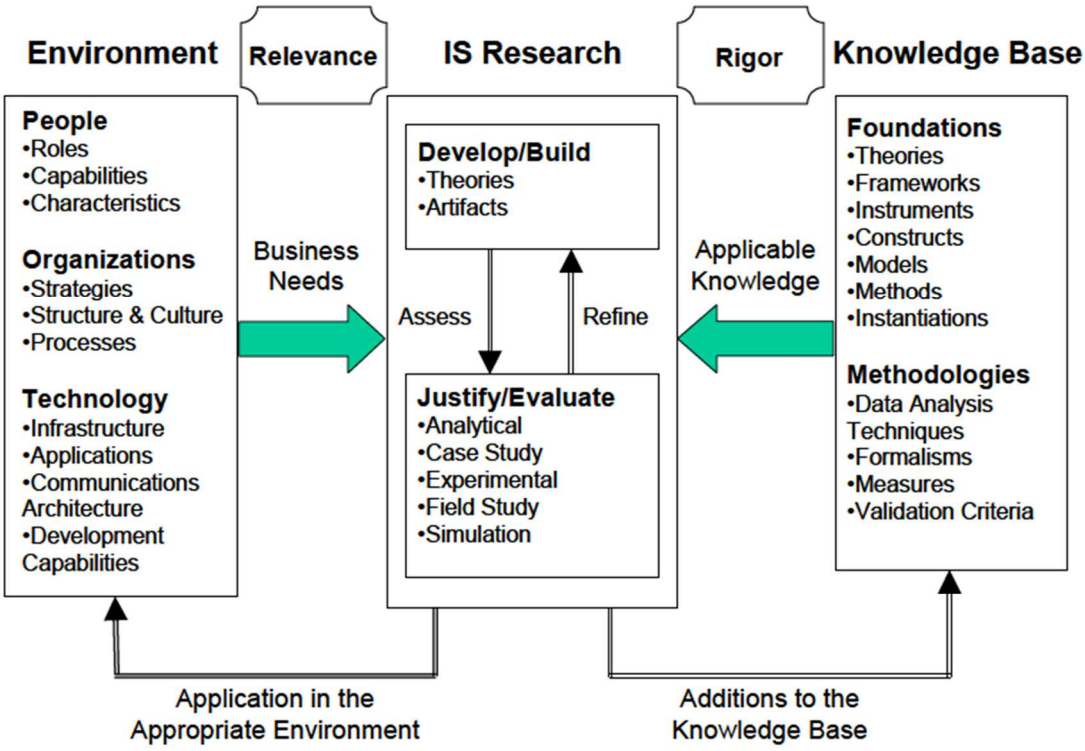
The third research objective encompasses the testing of the proposed model. A case study research is carried out by presenting and using an exemplary dataset and business scenario. The dataset and business environments are closely examined and introduced to the reader. From these context variables, a preference towards what criteria an optimal method should fulfill is given by the decision-maker (DM) and a suggestion on the best-fit algorithm is returned. For the data set, the best- and the worst-fit algorithm will be implemented. The results of both prognoses are then compared through appropriate measures.

1.3 Research Design

The thesis' research design follows the design science information systems (IS) research framework proposed by HEVNER ET AL. (2004) which can be seen in Fig. 1.

The environment encompasses the problem context that is composed of people, organizations and technology for information systems (Hevner et al. 2004, p. 79). The problem context of this research, which has been described in Section 1.1, stems from increasing systems complexity, even though algorithms exist to handle the complexity and estimate their reliability (technology). Several business stakeholders want to employ PHM (organizations), but practitioners lack the knowledge to select the right algorithms (people).

To be able to answer the business needs, one must also consider the existing knowledge base "through which IS research is accomplished" (Hevner et al. 2004, p. 80). This knowledge base are existing theoretical foundations and research methodologies. In this thesis, rigor is maintained by a SLR based on the framework of VOM BROCKE ET AL. (2009) which can be seen in Appendix A.a (Fig. 20). First, the general review scope was defined in phase I, by using the taxonomy for literature reviews presented by COOPER (1988, pp. 109–112). COOPER distinguishes between six characteristics, which are introduced in Tab. 1. In this table the chosen review scope is highlighted in grey. A detailed explanation can be found in Appendix A.b.



(Hevner et al. 2004, p. 80)

Fig. 1 Information Systems Research Framework

Characteristic	Categories			
Focus	Research Outcomes	Research Methods	Theories	Applications
Goal	Integration	Criticism	Identification of Central Issues	
Perspective	Neutral Representation		Espousal of Position	
Coverage	Exhaustive	Exhaustive with Selective Citation	Representative	Central or Pivotal
Organization	Historical	Conceptual	Methodological	
Audience	Specialized Scholars	General Scholars	Practitioners or Policy Makers	General Public

Based on: (Cooper 1988, pp. 109–112)

Tab. 1 Taxonomy for Literature Reviews

From the knowledge base, the business need can be finally addressed by synthesizing existing theory (i.e. fleet-based prognostic methods, research objective 1), developing an artifact (i.e. the decision model, research objective 2) and evaluating it (i.e. through a case study, research objective 3).

The whole research follows the seven guidelines of HEVNER ET AL. (2004, p. 83) which are summarized in Tab. 2. The first guideline is accomplished by constructing a decision model which is the artifact. The thesis relates to the problem of selecting an appropriate algorithm for a prognostic scenario that is faced by practitioners as well as scholars (guideline 2). Through a case study the design should be adequately evaluated by benchmarking the performance of a suggested and non-suggested algorithm (guideline 3). The work contributes to research by offering a classification of previous literature and a decision support model (guideline 4). The construction of the decision model is accomplished

Guideline	Description
Guideline 1: Design as an Artifact	Design-science research must produce a viable artifact in the form of a construct, a model, a method, or an instantiation.
Guideline 2: Problem Relevance	The objective of design-science research is to develop technology-based solutions to important and relevant business problems.
Guideline 3: Design Evaluation	The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods.
Guideline 4: Research Contributions	Effective design-science research must provide clear and verifiable contributions in the areas of the design artifact, design foundations, and/or design methodologies.
Guideline 5: Research Rigor	Design-science research relies upon the application of rigorous methods in both the construction and evaluation of the design artifact.
Guideline 6: Design as a Search Process	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
Guideline 7: Communication of Research	Design-science research must be presented effectively both to technology-oriented as well as management-oriented audiences.

(Hevner et al. 2004, p. 83)

Tab. 2 Design-Science Research Guidelines

through a rigorous SLR and the evaluation is done with scientifically proven prognostics measures (guideline 5). The design is conducted as an iterative search and the diverse algorithms of the underlying literature are abstracted and generalized (guideline 6). The ready-for-use decision model is suited for management-oriented audiences, while the presented prognostic methods are easy to understand for technology-oriented users (guideline 7).

1.4 Thesis Structure

In the next Chapter, the conceptualization of the topic is presented. Key words are introduced and the fundamental vocabulary that is required to understand the key concepts of this work are shown to the reader (i.e. pivotal concepts such as PHM, condition-based maintenance, fleets etc.).

In Chapter three, the first research objective is addressed by collecting literature on different fleet-based prognostic methods and analyzing these by considering their properties and modes of operation.

In Chapter four, the decision-relevant criteria of prognostic approaches are identified and assigned to the previously researched methods. Through this, a decision model is constructed (research objective two).

In Chapter five, the model is tested with a hard drive disk (HDD) dataset under a realistic business scenario and suggested algorithms are returned. The decisions of the DM are deduced by qualitative and quantitative reasoning. Afterwards, a best-fit approach that has been returned and a worst-fit approach are compared and evaluated for one test data set. Th evaluation is done by implementing both algorithms and comparing them with relevant (qualitative and quantitative) performance indicators.

The results of the thesis are discussed within Chapter six.

Chapter seven presents a conclusion and potentials for future research are suggested (research agenda).

2 Theoretical Concepts

In this Chapter, phase II of the framework by VOM BROCKE ET AL. (vom Brocke et al. 2009) is presented. Phase II introduces definitions of key terms and presents "what is known about the topic and potential areas where knowledge may be needed" (Torraco 2005, p. 359). The key terms were analyzed by looking at summary journal articles and seminal handbooks and by concept mapping.

2.1 Prognostics

Since the existence of complex and expensive systems, the desire for accurate prognostics is around (Vachtsevanos et al. 2006, p. xv). To be able to successfully employ prognostics, multiple disciplines need to be combined.

2.1.1 Prognostics and Health Management

Condition-based maintenance (CBM) is one of the oldest disciplines of prognostics and was originally used for purely diagnostics purposes (Tinga and Loendersloot 2014, p. 163). CBM in its most general sense is the monitoring of a system's reliability by looking at some key health indicators. Prominent CBM techniques are "vibration monitoring, oil analysis, acoustic emission and thermography" (Tinga and Loendersloot 2014, p. 163). Especially data driven techniques benefit from the collection of condition data through CBM and today they can accurately diagnose whether a machine's health is declining. More challenging however is the combination of condition data with prognostic methods to accurately predict the RUL of a system.

CBM shares many commonalities and is also an enabling factor for prognostics and health management (PHM). PHM is a very topical subject in modern research. In its most general sense, PHM comprises methods to evaluate the reliability of a system with special regard to its working environment (Trapani et al. 2015, p. 996). While the name indicates otherwise, PHM is not only occupied with prognostics, but also with detection and diagnosis of failures (Guillén et al. 2016, p. 992). The discipline encompasses several steps from data acquisition to decision support. As a reference for the practical application of PHM, some steps are recorded in ISO 13374:2012. The ISO guideline comprises the following six levels (steps) that make up PHM: 1) Data acquisition, 2) data manipulation, 3) state detection, 4) health assessment, 5) prognosis assessment and 6) advisory generation. In level 5, typically the RUL is predicted, to be able generate advisory. The RUL "refers to the time left before observing a failure given the current machine age and condition, and the past operation profile" (Jardine et al. 2006) and it is normally represented "as a point-estimated value, an interval estimated value, and a randomly distributed value based

on a normal, log-normal, Weibull, or inverse Gaussian distribution" (Yang et al. 2018, p. 407). In this thesis, prognostics algorithms are evaluated which predict the RUL of a given system from available fleet data.

2.1.2 Categories of Prognostic Approaches

To carry out a prognosis, many different prognostic algorithms can be used. ELATTAR ET AL. distinguished between four types of prognostic approaches (Elattar et al. 2016, pp. 133–137):

1) Reliability-based approaches. These approaches do not assess individual components in real time and do not consider different operating conditions. Instead, they rely on historic data and average run-to-failure (RTF) rates. The approaches are very simple, but also inaccurate and can cause too early or too late replacements (Elattar et al. 2016, p. 134).

2) Physics-based approaches. These approaches are the most complex, because a physical model of a machine is developed. These approaches require a lot of effort and domain knowledge and are rarely used in practice, due to their complexity (Elattar et al. 2016, p. 134).

3) Data-driven approaches. Although physics-based approaches are more accurate, data-driven approaches are more predominant in practice. Data-driven approaches use CBM data (i.e. pressure, speed, temperature, vibration, current, etc.) and correlate it with degradation to build prognostic models. The approaches do not require domain knowledge and are fairly simple to setup (Elattar et al. 2016, p. 135).

4) Hybrid approaches. These approaches combine data-driven and physics-based approaches to account for a lack of data or the lack of knowledge of a system's physics (Elattar et al. 2016, p. 137).

In this thesis, data-driven approaches are the focal point of the decision model, as they offer a high level of generalizability and can be easily adapted and implemented to different contexts. Reliability-based approaches are not considered, because the fleet-principle is based upon the foundation that no sufficient historic data for the machine exists. Physics-based approaches are not examined, because they are very specific to the context in which they have been implemented and thus they are hard to adapt to different systems (Leone et al. 2017, p. 163). Hybrid approaches are not considered as they too comprise a physics-based, context-specific component (cf. for instance Blancke et al. 2018).

2.1.3 Data-driven Approaches

SIKORSKA ET AL. state, that "there is little consensus among reviewers of the prognostic field as to what classifications are most appropriate for grouping remaining useful life prediction models" (Sikorska et al. 2011, p. 1809). Multiple (meta-)reviews by JARDINE ET AL. (2006), HENG ET AL. (2009), SI ET AL. (2011), SIKORSKA ET AL. (2011), LEE ET AL. (2014), KAN ET AL. (2015), SUTHARSSAN ET AL. (2015), LEI ET AL. (2018) and SCHWABACHER AND GOEBEL (2007) were used to find types of RUL prediction algorithms. The relevant data-driven algorithms are classified as statistical approaches or AI approaches and are presented in the following.

Statistical Approaches

Bayesian Statistics. Three different types of Bayesian techniques are commonly used for PHM: Bayesian networks, Kalman filters and particle filters (Lee et al. 2014, p. 323; Sikorska et al. 2011, p. 1812). Bayesian networks are multilevel models whose coefficients vary at multiple levels. They are represented in a directed acyclic graph where the conditional relations are modeled (Lee et al. 2014, p. 323). They can be used to draw upon knowledge from multiple prior distributions, such as a single system prior and a fleet prior (Zaidan, Harrison, et al. 2015, p. 542). While Bayesian networks can be solely used for prognostics, "both Kalman and Particle filters, are not different types of models per se, but rather different approaches to implementing generic dynamic Bayesian networks" (Sikorska et al. 2011, p. 1822). Particle filters start with a set of parameters for a degradation model that are drawn randomly from a prior distribution. Then some samples (particles) are drawn from (signal) observations and the likelihood to the degradation models is measured. All particles are then resampled, while particles with a higher likelihood are given more weight (Raghavan and Frey 2016, p. 2). A Kalman filter is another Bayesian method. It minimized the estimation covariance of a state estimation by incorporating measurements related to the state (Lee et al. 2014, p. 323).

Reliability Functions. A first hazard function was proposed by COX (1972); the Cox regression. The model which can be seen in equation (1) comprises a baseline hazard function that takes the age of the system into account and a covariate term that accounts for the effect of m covariates on the hazard function.

$$h(t) = \lambda_0(t) e^{\gamma_1 Z_1(t) + \gamma_2 Z_2(t) + \cdots + \gamma_m Z_m(t)} \quad (1)$$

In contrast to simple regression methods (e.g. linear), the covariates have a multiplicative instead of an additive effect on the function (Sikorska et al. 2011, p. 1825). A often used baseline function is the Weibull distribution (Jardine et al. 2006, p. 1490).

Additionally, the Weibull and other statistical distributions, such as exponential, normal, lognormal or Gaussian functions are often used for modeling failure behavior outside of the Cox regression and have been widely accepted for over 30 years (Sikorska et al. 2011, p. 1818). Normally, the parameters of the distributions are calculated by maximum likelihood estimation, maximum-a-posteriori estimation or expectation maximization (Sutharssan et al. 2015, pp. 3–4). The gamma process model is another frequently used reliability function model, that works on the gamma distribution (Lei et al. 2018, pp. 816–817).

Markov Model. Markov models assume that systems and their corresponding signals go through a finite discrete or infinite continuous number of states (Kan et al. 2015, p. 5). A system can only be in one state at a time and degradation is estimated by transition probabilities to a final failure state. The core assumption of Markov models, the so-called Markov property, is that the transition probabilities are only dependent on the current state (Sikorska et al. 2011, p. 1821). Regular Markov models in prognostics work on observable (health) states of the system. These are often not observable and thus hidden Markov models can be used (Lei et al. 2018, p. 817). In Markov, as well as hidden Markov models, it is also assumed that state transition probabilities are static. Because this is not always realistic, semi-Markov models offer increased flexibility by adapting the probability based on how long the system already remained in the current state (Si et al. 2011, p. 7).

Regression analysis. Regression-based models are often used for RUL estimation in practice and research due to their simplicity (Si et al. 2011, p. 3). The simplest form, a linear regression, is based on correlating a single independent variable (i.e. a single signal or compound value) to the dependent variable (i.e. the RUL) of the system (Sikorska et al. 2011, p. 1823). Regression analysis can also be adapted to multiple independent variables. The regression function adapts a polynomial shape in this case.

Additionally, logistic regression and the autoregressive moving average (ARMA) method is also used for prognostics to calculate the probability of failure (Jardine et al. 2006, p. 1496). Logistic regression is suited for predicting dichotomous dependent variables. In a prognostic context the existence (or absence) of failure is an exemplary dichotomous variable. ARMA models assume that the future of a system can be modeled as a linear function of historic values as can be seen in equation (2)

$$x_t = a_1 x_{t-1} + \cdots + a_p x_{t-p} + \varepsilon_t - b_1 \varepsilon_{t-1} - \cdots - b_q \varepsilon_{t-q} \tag{2}$$

where x_t is the predicted variable for time t which is derived by the historic signals x, a Gaussian noise term $\sim \mathcal{N}(0, \sigma^2)$, multiple autoregressive components, comprising a para-

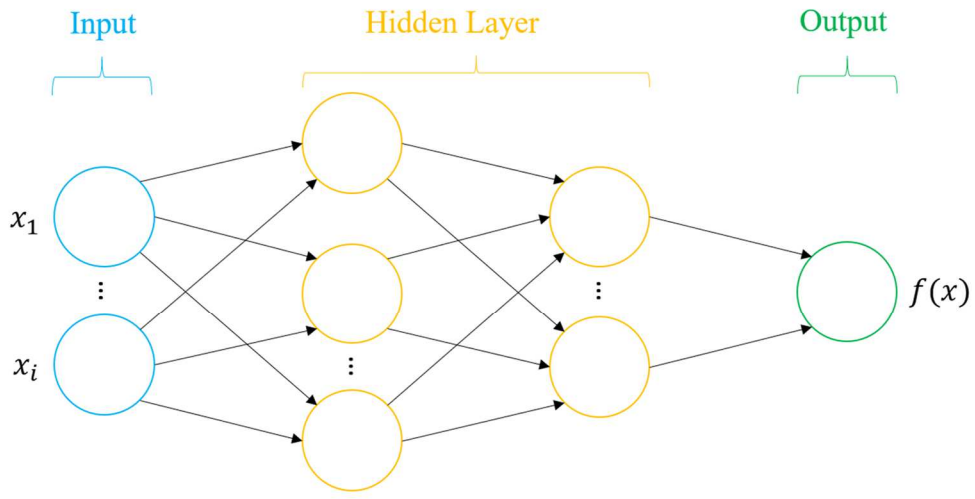
meter a and an order p and multiple moving average terms consisting of a parameter b and an order q (Jardine et al. 2006, p. 1487). A match matrix is an enhanced ARMA model, because it utilizes historical data from various operations (Kan et al. 2015, p. 3).

"Strictly speaking Wiener processes are types of regression models" (Si et al. 2011, p. 4), but it comprises some special properties. A Wiener process comprises three components: a drift term $\boldsymbol{\lambda}$, a diffusion coefficient $\boldsymbol{\sigma}$ and Brownian motion $\boldsymbol{B(t)}$. Wiener processes are arguably one of the most commonly used stochastic process models (Lei et al. 2018, p. 816).

Artificial Intelligence Approaches

Gaussian Process Regression. Gaussian processes regression (GPR) can be used to estimate the RUL function through a finite number of joint variables which all follow a Gaussian distribution. Thus, it is suited for non-linear regression tasks (Sutharssan et al. 2015, p. 5). Gaussian process regression is one of the most important Bayesian ML algorithms (Kan et al. 2015, p. 6). It is used to fit models and retrieve an underlying degradation process by placing a prior distribution that constrains the posterior function (Lee et al. 2014, p. 322). Similar to most AI algorithms, GPR can be divided into a training (offline) phase, where the output is predicted by weighting targets, and a prediction phase (online).

Artificial Neural Networks. The artificial neural network (ANN) is the most used data-driven technique in general prognostics literature (Heng et al. 2009, p. 729). ANNs try to simulate the structures of biological neural networks. They can learn complex relationships between inputs and outputs and find patterns (Lee et al. 2014, p. 323). NNs can be depicted as graphs, as can be seen in Fig. 2, and they comprise an input layer, one or



Own depiction

Fig. 2 Exemplary Neural Network

multiple hidden layers and one output layer (Kan et al. 2015, p. 8). Weights are assigned to the edges, while the graph's nodes, the so-called neurons, contain an activation function that is only activated if the weighted input value exceeds a certain threshold. NNs can assume any complex mathematical function, e.g. one that represent the degradation of a system, by adjusting the interconnecting weights. This is done in a training phase were inputs are used for making predictions and the weights are then adjusted to minimize the error (Sutharssan et al. 2015, p. 5). There exist different types of NN for both supervised (outcome known) and unsupervised learning (outcome not known). Frequently used types that are used for prognostics include feed-forward networks, recurrent NNs, self-organizing maps (SOM) and radial basis function networks (Kan et al. 2015, pp. 8–9).

Fuzzy Logic and Neuro-Fuzzy Systems. Fuzzy logic is recognized as a powerful tool to solve real-world problems under high levels of uncertainty (Mahdaoui and Mouss 2012, p. 477). Also, while NNs are not very explainable, fuzzy logic overcomes this drawback due to its "higher level of transparency and openness" (Kan et al. 2015, p. 9). Fuzzy logic changes the bivalent assignment of objects to classes into gradual membership. Instead of an object belonging to one class, it can gradually belong to multiple classes; this is determined by a membership function. Typically, fuzzy logic is used in conjunction with another AI method for PHM (Schwabacher and Goebel 2007, p. 3). Neuro-fuzzy systems (NFS) combine fuzzy logic with NNs, where the inference structure is defined by expert knowledge and the membership functions are optimized through NNs (Heng et al. 2009, p. 729). Fuzzy NNs typically have six layers: 1) input, 2) assignment of input to fuzzy set through membership functions, 3) inference with fuzzy expert rules, 4) normalization, 5) function parameter estimation, 6) output (Kan et al. 2015, p. 10).

Random Forests. Decision trees (DT) can be used for regression by mapping features to outcomes (Voronov et al. 2018, p. 626). For a DT, the feature space is split into different distinct regions and paired with the best fitting outcome. The number of splits and thus the tree structure can be typically determined with three parameters: the minimum number of outcomes that are assigned to each terminal node of the tree, the number of split variables and the number of split values (Frisk et al. 2014, p. 4). The final structure is then dynamically determined by partitioning while minimizing a cost function (Voronov et al. 2018, p. 626).

Regressive random forests (RF) are ML regression techniques that make use of an ensemble of DTs. Through the ensemble, random forests can generalize well without overfitting. This is done by bootstrap aggregating, so-called bagging (Wu et al. 2017, p. 4). Bagging draws $i = 1, \dots, m$ training samples X_i with size n from the total set of data X and constructs m decision trees. Afterwards, each regressive decision tree is split in a

recursive binary fashion as stated above. The final assemble of trees is then output and aggregated, e.g. by averaging.

A random survival forest (RSF) is a type of random forest that is adapted to survival analysis. It works by using the log-rank test as a cost function, a statistical test that compares the distributions of different observations (Voronov et al. 2018, p. 627). Sample sets that vary as much as possible in their distributions are put into two child nodes and the algorithm is recursively called again for each child node.

Vector Machines. A support vector machine (SVM) was proposed by VAPNIK (2009) and tries to project input data into high dimensional space so that a hyperplane can be setup to optimally split the data based on their labels (Sutharssan et al. 2015, p. 5). The key process for SVMs is the kernel function (Lee et al. 2014, p. 324) that is defining one base function per training sample. Originally, the algorithm assumed two classes for a classification problem (Kan et al. 2015, p. 7), but different SVMs have since been used for prognostics, such as least-square SVMs, one-class SVMs or multi-class SVMs. A typical form for RUL prognostics is the support vector regression (SVR) that can predict point estimates through regressions (Lei et al. 2018, p. 818). Instead of separating two classes, the hyperplane is fitted through each observation with the goal of minimizing the error between the observations (Baptista et al. 2016, pp. 3–4).

TIPPING (2001) proposed the relevance vector machine (RVM) that tackles the issue of returning only point estimates. This is done through applying Bayesian statistics that constrain the weights of the vector model through a prior distribution (Tipping 2001, p. 213). Beyond returning a probabilistic estimate, RVMs also utilize less kernel functions.

Ensembles

Ensembles have been introduced in the context of RFs, but the principle can be extended to any prognostic algorithm. Ensembles combine the predictions of multiple member algorithms which need not necessarily to be of the same type, such as RVMs, SVMs, exponential functions and neural networks (Hu et al. 2012). Their predictions can be combined with a weighted-sum, where predictions of algorithms which perform better in the training phase are weighted stronger.

All in all, data-driven methods are promising, however all approaches have one requisite: huge amounts of multivariate historic data about behavior of one specific system must be available; a feature which is often missing for unique complex systems (Michau et al. 2018). The required data must cover all phases of a systems lifecycle, i.e. normal as well as faulty operation data, but unfortunately, moderately big systems are seldom identical

to other systems from which RTF data could be extracted and used for RUL determination (Al-Dahidi et al. 2016, p. 110). Fleet-based approaches try to overcome this issue and ultimately increase the training data size.

2.1.4 Fleets

Instead of looking at a singular system, data from similar machines of a fleet could potentially increase the training data size, making data-driven approaches much more reliable. When combining data from machines with similar working conditions, it is important to know what the differences are and how these affect the behavior of the system. Normally this is a computationally intensive task, but today many algorithms, such as ML approaches, can be used to effectively measure similarities between data sets (e.g. Michau et al. 2018). To know which approaches are applicable, it is critical to know which type of fleet is available. In current research it is distinguished between three general types of fleets (Al-Dahidi et al. 2016, p. 110; Medina-Oliva et al. 2012, p. 2f.).

Identical fleets. The features, usage and working condition of the machines are identical. Example: a fleet of identical diesel engines located in one ship (Al-Dahidi et al. 2016, p. 110).

Homogeneous fleets. The features are similar, however the usage or working conditions are different. Example: a fleet of trains driving on the same route (Umiliacchi et al. 2011).

Heterogeneous fleets. The fleets have different technical features and also differ in usage and working conditions. Example: highly standardized steam turbines of nuclear plants (Al-Dahidi et al. 2016, p. 110).

While these three types are very general, there are multiple further dimensions and types of fleets (Wagner and Hellingrath 2017); for this research however, the introduced terminology suffices. Because there are so many different fleet types and other relevant factors, there are also proportional many different data-driven algorithms, which are tailored to different situations. Unfortunately, the in Section 2.1.3 presented (meta-)reviews of data-driven prognostic approaches do not take the notion of fleets into account. Consequently, it is hard to find the optimal prognostic approach and there exists no framework in current research to support the decision of selecting adequate algorithms based on fleet dimensions and other variables. In this thesis the introduced classifications are extended to fleet-based approaches (research objective 1). This classification is necessary for constructing a decision model and will be introduced in Chapter 3. Before this is done, a further important conceptual domain – decision theory – is introduced which is also fundamental for the decision model.

2.2 Decision Theory

Decision theory is a subset of game theory, more specifically a game with two individuals, where one player is the decision-maker and nature is adopting the role of the second player. The only difference to game theory is, that instead of both players trying to maximize their winnings, nature "chooses" a state without having a particular goal (de Almeida and Bohoris 1995, p. 39).

2.2.1 Multi-Criteria Decision Making

DE ALMEIDA AND BOHORIS identified some basic ingredients of a decision process (1995, pp. 39–43).

The basic laws. The basic laws or the "state of nature" θ define the circumstances of the decision problem. Depending on which constellation of criteria apply, n different states can influence the decision. The different states can be denoted as $\theta_1, \theta_2, \dots, \theta_n$. The state of nature is not changeable by the decider. A simple example of basic laws in a maintenance context would be a situation where the failure rate λ of machines is under or above a certain threshold λ_0. $\theta_1(\lambda > \lambda_0)$ signifies unreliable hardware, $\theta_2(\lambda < \lambda_0)$ signifies reliable hardware (de Almeida and Bohoris 1995, p. 40). Another example would be r which denotes the similarity of a fleet of machines. $\theta_1(r \to 0)$ is a scenario where the observed fleets are heterogeneous, $\theta_2(r \to 1)$ denotes homogeneous and $\theta_3(r = 1)$ identical machines. The number of states correlate to the number of possible values of a criterion (correspondingly: two for the first and three for the second example) in a single-criteria decision problem or the Cartesian product of all criteria values when there are multiple criteria.

The alternatives. The DM can choose between a set of alternatives $\underline{A}$. $\underline{A}$ can be denoted as $\{A_1, A_2, \dots, A_m\}$ and the DM can choose any alternative A_j from this set. Each alternative is one prognostic algorithm or even a group of similar algorithms.

The consequences. Each combination of basic law θ_i and alternative A_j have an outcome ρ_{ij}. For n basic laws and m alternatives, the number of outcomes can be depicted in a $n \times m$ matrix. The consequences are normally assumed by typical characteristics of a prognostics scenario, previous knowledge or the results of experiments and preceding research (de Almeida and Bohoris 1995, p. 40).

The utility function. The utility function $U(\rho) = U(\theta, A)$ quantifies the outcome and represents the utility the DM gains from choosing alternative A under θ. The goal of the DM is to optimize $U(\rho)$. It is rarely the case that there exists a dominant solution A_j, which is

a solution that has a bigger $U(\rho_j)$ than any other alternative, regardless of the state of the nature θ.

*Multi-attribute utility theory (MAUT). MAUT a*llows the aggregation of multiple consequences $\rho = \{\rho', \rho'', \dots\}$, if θ comprises multiple attributes (criteria) that influence the overall outcome. In a two-criteria case, the utility $U(\rho', \rho'')$ can be calculated by adding the single-attribute functions and weighing them by two weights w_1, w_2, where $w_1 + w_2 = 1$.

$$U(\rho', \rho'') = w_1 U(\rho') + w_2 U(\rho'') \tag{3}$$

If θ_i is consisting of multiple decision relevant criteria, the choice of A_j by optimization of $U(\rho_{ij})$ can be made through multi-criteria decision making (MCDM) or multi-criteria decision analysis (MCDA). It can be hard to model each combination of states, actions and consequences and it might also take a long time to optimize the utility function. Thus, faster and more efficient decision algorithms are required.

Additionally, the choice of the optimal prognostics algorithm cannot be done by purely objective probability measures (Chen and Hwang 1992, p. 2). There are many dimensions that cannot be expressed by values or and mathematical functions (e.g. explainability, suitability to fleet) and a good MCDM method takes this into account. The most basic MCDM method is the weighted sum model (WSM).

WSM. The WSM is one of the simplest methods for decision making. For each a in A, the utility U is calculated by multiplying the n-criteria outcome ρ by a weighting factor w and summing up the products (Triantaphyllou et al. 1997, p. 18). For simplification, the state θ is known and treated as a constant.

The utility function is calculated for each alternative and the alternative with the highest utility score is chosen as the best alternative (cf. equation (4)). The weighted sum model is tailored for problems where the majority of criteria is fuzzy (Chen and Hwang 1992, p. 12).

$$U(a) = \sum_{j=1}^{n} w_j U_j(a) \tag{4}$$

While the method is based on “additive utility” (Triantaphyllou et al. 1997) and therefore supports multiple criteria, it has the drawback that all attributes must be numerical and their scales identical, because the function is not scale invariant (Bevilacqua and Braglia 2000, p. 75).

CHEN AND HWANG identified multiple MCDM methods (1992, p. 12) and categorized them by problem size. MARDANI ET AL. systematically reviewed 1081 MCDM techniques in more than 150 peer-reviewed journals (2015). The three most suited methods, the Analytic Hierarchy Process (AHP), the Technique for Order Performance by Similarity to Ideal Solution (TOPSIS) and the Analytic Network Process (ANP) are presented in the following. The methods were chosen, because they are most commonly used for prognostics (Mardani et al. 2015) and each fulfills a specific role for different MCDM problems.

2.2.2 Analytic Hierarchy Process

AHP is a decision tool with ratio scales that was introduced by SAATY (Saaty 1980, 1987, 1990). It tries to tackle the scale variance of the WSM by asking the decision-maker to supply ratios for pairwise comparisons between alternatives $A_1, A_2, \dots, A_n$ for each of their criteria $C_1, C_2, \dots, C_m$. Additionally, the attributes can be ordered hierarchically, and the weights of each child-node must amount to the total of its parent node.

First, by pairwise comparisons, the DM calculates the preference of each criterion with a scale from one for equal importance to nine for extreme importance. When comparing the criteria, a reciprocal matrix C is set up for each attribute (equation (5)). Let c_{ij} be the preference of the criterion C_i to C_j, where w_i is the weight of preference of C_i and $c_{ij} = w_i/w_j$. The reciprocal of c_{ij} is $c_{ji} = w_j/w_i$.

$$C = \begin{array}{c} \\ C_1 \\ C_2 \\ \vdots \\ C_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_m \end{array} \\ \begin{bmatrix} 1 & c_{12} & \cdots & c_{1m} \\ c_{21} & 1 & \cdots & c_{2m} \\ \vdots & \vdots & & \vdots \\ c_{m1} & c_{m2} & \cdots & 1 \end{bmatrix} \end{array} \tag{5}$$

For the matrix, the eigenvalues λ and -vectors U are calculated. The largest eigenvalue λ_{max} and its corresponding eigenvector, called the priority vector U_{max} are used for further calculation. U_{max} represents the weights of the judgements. SAATY suggests an approximation by normalizing the elements in each column of C and then averaging over the rows (1987, p. 170) to calculate U_{max}.

Often, pairwise judgements are not consistent. For instance, the DM could rate $c_{12} > c_{23} > c_{31}$. Inconsistency might be valid to some degree, because human judgements cannot be so accurate that there is no inconsistency (Chen and Hwang 1992, p. 333). On the contrary, it is even important, because with it, new knowledge that changes the order of

preference can be admitted (Saaty 1987, p. 172). Inconsistency can be measured with the consistency ratio which can be calculated with equation (6).

$$C.R. = \mu / r \tag{6}$$

The consistency ratio comprises the consistency index μ and the random index r. μ can be calculated with equation (7).

$$\mu = \frac{\lambda_{max} - n}{n - 1} \tag{7}$$

n denotes the number of dimensions of the preference matrix (i.e. the number of criteria or alternatives). r is a random index that is computed by randomly assigning the 17 values $(1/9, 1/8, \dots, 1, 2, \dots, 8, 9)$ to the entries above the main diagonal of a $m \times m$ matrix, filling the lower triangle with the reciprocals and filling the main diagonal with ones (Saaty and Tran 2007, p. 966). Then, the consistency index μ is calculated for the random matrix and the process is repeated k times (k should be adequately high, e.g. 50.000). The index r can then be calculated with Equation (8).

$$r = \frac{1}{k} \sum_{i=1}^{k} \mu_k \tag{8}$$

A consistency ratio $C.R. \leq 0.1$ is acceptable. If it is over 0.1, the judgements should be revised until the consistency is adequate (Saaty 1987, p. 171).

After the preferences of the attributes have been analyzed, the same is done for the alternatives and their fit to the attributes. Therefore, a set $\bar{A}$ of m (number of criteria) further matrices with the size $n \times n$ ($n =$ number of alternatives) are constructed. The generalized matrix for n alternatives is shown in equation (9). Again, the set $\bar{A}$ is created by using the rating scale from 1 to 9. Instead of rating the preference of criterion C_i over C_j, the fits of the alternatives A_i over A_j regarding $C_k, \forall i, j, k$ is calculated. For each resulting matrix, the eigenvalues Λ_{max} and -vectors V_{max} are again constructed and the consistency ratio $C.R.$ is calculated.

$$A_k = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_n \end{array} \begin{array}{c} \begin{array}{cccc} A_1 & A_2 & \cdots & A_n \end{array} \\ \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ a_{21} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & 1 \end{bmatrix} \end{array} \tag{9}$$

The results are n principal eigenvectors V_j with a magnitude of m for each alternative. If i is the index of the alternative and j the index of the criterion, then v_{ij} is the ith component of the eigenvector corresponding to the largest eigenvalue of the matrix A that

corresponds to the jth criterion. Column wise and transposed, the results can be represented in a $m \times n$ matrix X that can be seen in equation (10). To decide upon which alternative is superior, the matrix must be multiplied by the principal eigenvector U_{max}^{T} of the criterion preference matrix C. u_j represents the component of the principal eigenvector for the jth criterion. The multiplication computes the priority vector D, with the priority values $d_1, d_2, \dots, d_m$ for each alternative (cf. equation (11)).

$$X = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mn} \end{bmatrix} \tag{10}$$

$$D = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & & \vdots \\ v_{m1} & v_{m2} & \cdots & v_{mn} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_m \end{bmatrix} \tag{11}$$

The highest d_i corresponds to the best solution for the decision problem and the DM should choose this alternative. Numerical examples can be found in various publications (Chen and Hwang 1992; Saaty 1980, 1987, 1990; Saaty and Tran 2007) and for cases in a maintenance context (Bevilacqua and Braglia 2000; Triantaphyllou et al. 1997).

To calculate above mentioned steps, there also exist multiple software solutions for the AHP model (Ossadnik and Lange 1999), as well as packages for R (e.g. CRAN ahp) and Python (e.g. pyAHP). The AHP is especially suited for complex problems where fuzzy, as well as crisp (i.e. non-fuzzy, clearly assignable to a specific set) aspects must be considered (Bevilacqua and Braglia 2000, p. 75). It is also scale invariant, because it deals with relative values and thus equation (12) holds (Triantaphyllou et al. 1997, p. 18).

$$\sum_{i=1}^{m} v_{ij} = 1 \,, \forall j \wedge \sum_{i=1}^{n} u_i = 1 \tag{12}$$

AHP is by far the most applied MCDM method in the past decades (Mardani et al. 2015, p. 4130).

2.2.3 Technique for Order Performance by Similarity to Ideal Solution

TOPSIS is a method that was introduced by HWANG AND YOON (1981) and extended to the fuzzy domain by CHEN (2000). According to this technique, the best alternative has the least geometric distance to the positive ideal solution (PIS) and is the farthest away from the negative ideal solution (NIS) (Assari et al. 2012, p. 2290). In their original work, HWANG AND YOON identified six steps to derive the optimal decision (Hwang and Yoon 1981, pp. 130–140).

Step 1. Instead of constructing pairwise preference matrices for the m criteria and for the n alternatives for each criterion as in AHP, only one matrix and one vector are constructed initially for TOPSIS. First, a vector $\underline{w}$ for the absolute preference of each criterion is constructed where the DM rates the weights of the n criteria while the sum of the weights must equal to one (equation (13)).

$$w = (w_1, w_2, \dots, w_m), \text{ where } \sum_{j=1}^{m} w_j = 1 \tag{13}$$

Then, the DM assigns absolute preference values to each x_{ij}, which is the preference value of the ith alternative for the jth criterion. The results are stored in a $n \times m$ matrix A which can be seen in equation (14).

$$A = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_n \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \cdots & C_m \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \end{array} \tag{14}$$

Step 2. The matrix A is then normalized by dividing each x_{ij} by the Euclidean length of the column vector of each criterion, thus obtaining $r_{ij} = x_{ij} / \sqrt{\sum_{i=1}^{n} x_{ij}^2}$.

Step 3. The new matrix $A_{norm}$is then transformed into the weighted normalized matrix V, by multiplying each value with the w_j of its corresponding criterion, thus obtaining $v_{ij} = w_j * r_{ij}$.

Step 4. Now the PIS and NIS vectors $\underline{v_p}$ and $\underline{v_n}$ are constructed. The vectors contain the best (e.g. highest) values v_{bj} and the worst values v_{wj} for each criterion respectively. The values can be obtained from any alternative; thus, the ideal solutions need not necessarily be real alternatives. The solutions can be expressed as $\underline{v_p} = (v_{b1}, v_{b2}, \dots, v_{bm})$ and $\underline{v_n} = (v_{w1}, v_{w2}, \dots, v_{wm})$.

Step 5. For each normalized weighted alternative i (i.e. each row of matrix V), the Euclidean distances d_{ib} to the best, and d_{iw} to the worst solution are calculated. The formula of the Euclidean distances can be found in equations (15) and (16).

$$d_{ib} = \sqrt{\sum_{j=1}^{m} \left(v_{ij} - v_{bj}\right)^2} \tag{15}$$

Step 6. Lastly, the similarity of the alternatives to the best solution is calculated with $s_{ib} = d_{iw}/(d_{iw} + d_{ib})$, $s_{ib} \in [0, 1]$. s_{ib} is one, iff the examined solution is also the best solution and zero, iff it is the worst solution.

$$d_{iw} = \sqrt{\sum_{j=1}^{m} \left(v_{ij} - v_{wj}\right)^2} \tag{16}$$

Multiple numerical examples can be found in the literature (Assari et al. 2012; Hwang and Yoon 1981, pp. 133–140). There exist packages in R (CRAN topsis), as well as Python (Papathanasiou and Ploskas 2018). TOPSIS is easy to use and calculate. A significant difference to the WSM is that the best possible solution has the shortest distance to the PIS and longest distance to the NIS, whereas the WSM's optimal solution is the farthest away from the origin. THOR ET AL. compared AHP, ELECTRE, WSM and TOPSIS and concluded that the latter "exhibited the highest potential in maintenance decision analysis", because it is suitable for large scale data and easy to implement (2013, p. 33). However, TOPSIS is criticized for having an unreliable weight assessment of the criteria (Hwang and Yoon 1981, p. 137) and is therefore often combined with other methods, such as AHP (Ilangkumaran and Kumanan 2009).

2.2.4 Analytic Network Process

ANP is the third most widely researched MCDM method (Mardani et al. 2015, p. 4130). It is the extension of the AHP and was developed in 1996 (Saaty 1996). The ANP uses a network instead of a hierarchy to depict criteria that have inner and outer dependencies (Saaty 1999, p. 2). It is moved away from the top-down structure "Goal – Criteria – Subcriteria – Alternative" to a node-based representation in which goals, criteria and alternatives are treated equally (Saaty and Vargas 2006, p. 8). The mathematical foundations are shared with the AHP and thus, the reader is advised to recall them from Section 2.2.2.

ANP comprises four steps (Shahin et al. 2012, pp. 470–473).

1. *Model construction.* Four so-called control hierarchies are set up for benefits, costs, opportunities and risks. The hierarchies can be decomposed into sub-components (Saaty 1999, p. 1). The control hierarchies are connected to each other in a network structure.

2. *Comparison matrices and priority eigenvectors.* Like the AHP, all criteria are compared pair-wisely. Peculiar for the AHP is, that control hierarchies are compared with each other and that the effect of each element on each other element is rated too with SAATY'S 1-9 scale (cf. Section 2.2.2). The normalized eigenvectors of λ_{max} of all matrices are the local priority vectors and the consistency of the comparisons is checked with the consistency ratio $C.R$.

3. *Super-matrix formation.* To obtain the global priorities a super-matrix is constructed. The super matrix contains the clustered nodes (e.g. goals, criteria or alternatives) on both axes and the connections and weights are stored within the matrix. The weights are stored as column vectors that represent the priority vectors from step two. The super-matrix must then be normalized.

4. Because the super-matrix covers the whole network, the priority weights for the alternatives can be found in the intersection of the alternatives on the vertical axis and the goal column on the horizontal axis. The alternative with the highest priority is the best selection.

Like for the other methods, there exists some numerical examples in the literature (Saaty 1996, 2005; Saaty and Vargas 2006), also for a maintenance context (Chemweno et al. 2015; Sadeghi and Manesh 2012; Shahin et al. 2012). There also exists a Python package for ANP (pyanp).

The ANP is preferable to the AHP, when there are criteria that cannot be structured in a hierarchy, for instance when higher-level components interact with or depend on lower-level components (Saaty and Vargas 2006). For instance, this is the case when the criteria weight has not only an influence on ranking of the alternative, but also when available alternatives have influence on the criteria ranking.

To conclude which MCDM method is the best one for the selection of prognostic algorithms, the types of goal, criteria and alternatives are crucial. It must be determined how many dimensions the decision problem includes and if the elements have reciprocal effects on each other.

3 Fleet-Based Prognostic Methods

After the conceptualization of the topic (phase II of the review framework of VOM BROCKE ET AL.) that has been presented in the last Chapter, the first research objective is solved through a SLR that synthesizes pivotal fleet-based prognostic methods. In the first Section the methodology of a SLR is presented, the second Section presents the different fleet types that are used in the acquired theory and a three-step process of fleet-based prognostic methods is introduced in Sections 3.3, 3.4 and 3.5.

3.1 Methodology

In phase III the actual literature search was carried out through journal, database, keyword, backward and forward search and continuous evaluation of all gathered sources (vom Brocke et al. 2009, p. 2212). Phase III is conducted in multiple steps. First, appropriate journals and conference proceedings must be identified. Journals are preferred, because they are peer-reviewed before publication. Nevertheless, articles from renowned conferences can also be included, but authors should focus on high quality articles; these can be identified by looking a rankings (e.g. WALSTROM AND HARDGRAVE offer rankings for IS conferences (2001, p. 121)). For PHM, it is hard to narrow down appropriate journals, because PHM is multidisciplinary and "draws from electrical, electronics, mechanical, civil, and chemical engineering, computer and materials science, reliability, test and measurement, artificial intelligence, physics, and economics" (Saxena 2014, p. 1). While there exist some journals specifically targeted to the area (e.g. International Journal of Prognostics and Health Management), it would be fatal to exclude any journal, because many works are still published in media of the specialist fields. Thus, for this literature review, the literature search process of VOM BROCKE ET AL. (2009, p. 2212) was slightly altered by moving the journal and proceedings search to step three.

1. *Database search.* Instead of identifying relevant journals first, a database search is carried out. The Scopus database was chosen, because it is the largest abstract and citation database of peer-reviewed literature. Numerous publications from relevant fields mentioned by SAXENA (2014, p. 1) are indexed at Scopus. Additionally, because they are not available in Scopus, the conference proceedings of the PHM society were also included by accessing them from the PHM society website (https://www.phmsociety.org/).

2. *Keywords.* To query the databases, adequate key words were phrased and connected with logical operators. The key words stem from three important domains: fleet, prognostics and methods. To accommodate for an exhaustive literature review, many keywords and synonyms were used. For instance, the domain fleet was hard to query,

because fleet is not an established keyword in science (yet). Synonyms like “similar unit” or “heterogeneous system” were used as well. Additionally, only the title, abstract and article keywords were searched, to narrow down the review. A visualization of the key words and the actual search string can be found in the Appendix B.a and B.b. The keyword search led to 318 article and 485 conference proceedings hits.

3. *Journal and conference proceedings search.* As mentioned earlier, the first initial hits are only now filtered by their source. Journals, that do not fit into any of the fields mentioned in SAXENA (2014, p. 1) were filtered (e.g. medicine, arts, psychology). Additionally, because the quality of conference proceedings is considered to be lower (Levy and Ellis 2006, p. 187), only conferences that are specifically targeted at prognostics or closely related sciences were included. A list of all included conferences can be found in Appendix B.c. The journal and conference in-/exclusion reduced the hits to 175 journal articles and 84 proceedings.

4. *Title/Abstract/Full text evaluation.* Ultimately, the literature was closely examined by first looking at the title. If it was obvious that the article did not contain a fleet-based prognostic method, then it was omitted. If it was not clear, the same procedure was applied to the abstract and then the full text. The final literature that was reviewed comprised 37 works (21 journal articles and 16 conference proceedings).

5. *Backwards search.* When methods were referenced that were not included in the initial search, they were additionally included via a backwards search. The backward search was kept very narrow and strict, so that only methods were included that served as a direct foundation for the referencing papers. For instance, LIM ET AL. mentioned that their method “was recreated to the best of knowledge based on the details given in Peel” (2014, p. 6), so the publication by PEEL (2008) was included via a backward search. The search led to twelve further inclusions, resulting in a total of 49 articles.

The results of the literature review are summarized in Tab. 3. The column ‘Hits’ contains the results by just querying the keywords (step two), ‘Hits After Source Exclusion’ the hits after filtering for specific journals and conferences (step three), ‘Reviewed’ the sources after title, abstract and full text evaluation (step four) and “Backward Search” the results from the identically named step (five).

Document Type	Hits	Hits After Source Exclusion	Reviewed	Backward Search	Total
Journals	318	175	21	6	27
Conference Proceedings	448	84	16	6	22
Total	766	259	37	12	49

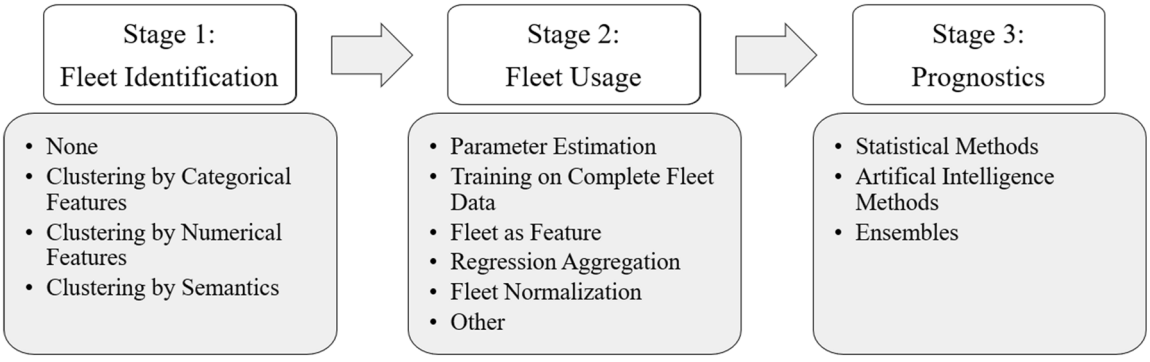
Tab. 3 Results of the Literature Review

3.2 Fleet Types

Depending on the type of fleet, all identified methods apply different data preprocessing steps, before the fleet data is used for prediction. The author could identify a general three-stage approach for fleet-prognostic methods that is depicted in Fig. 3. The first two stages, fleet identification and usage, are steps that only occur in fleet-based approaches, while the prediction is a stage that is also found in non-fleet-based data-driven approaches.

From the total of 49 articles, 33 dealt with homogeneous fleets, 9 with heterogeneous fleets and 7 with identical fleets. All the articles could be classified by the three definitions; either they were named by the authors directly or the type of features and working conditions was described and thus the established definitions could be easily applied.

As defined in Section 2.1.4, homogeneous fleets are characterized by similar features, however the usage or working conditions are different. Homogeneous fleets are by far the most used type of system groups. DUONG ET AL. researched two fleets of homogeneous light-emitting diodes (LEDs) that have undergone an accelerated degradation within two different working conditions. One fleet ran with a current of 450 mA and a temperature of 550°C, while the other ran with 200 mA and 900°C (2018, p. 82). It is notable that the different working conditions led to very different degradation trajectories (2018, p. 82). Multiple algorithms worked on the popular NASA Commercial Modular Aero-Propulsion System Simulation (CMAPSS) dataset that comprises data of homogeneous turbofan engines that were structurally identical but operated in different working conditions (Saxena, Goebel, et al. 2008). All of the authors could identify six different conditions by clustering three numerical features that expressed the altitude, Mach number and throttle resolver angle (Al-Dahidi et al. 2016; Babu et al. 2016; Hu et al. 2012; Lim et al. 2014; Peng et al. 2012; Riad et al. 2010; Rigamonti et al. 2016). TRILLA ET AL. examined



Own depiction

Fig. 3 Stages of Fleet-based Prognostic Methods

breakdowns of a fleet of structurally identical brakes, where the homogeneity stems from different positions in the train (Trilla et al. 2018, p. 3).

Heterogeneous fleets, in contrast to homogenous fleets, are characterized by different technical features and working conditions. To adapt algorithms to these data is often a difficult task, because there must be some data preprocessing to extract similar sub-fleets from the data. LUKENS AND MARKHAM used data from heterogeneous rotating assets that varied in component type, manufacturer and model (2018, p. 6). LEONE ET AL. worked with data from medium voltage and high voltage circuit breakers that are included in power transmission and distribution systems (2017, p. 164). A paper co-authored by AL-DAHIDI used aluminum electrolytic capacitors that are used in electric vehicle powertrains (Al-Dahidi et al. 2017a). Another paper examined one eight-station flexible manufacturing system with hundreds of different components (Kammoun and Rezg 2018, p. 1098). Typical for flexible productions systems, not all stations are used for every manufacturing process and thus a high level of heterogeneity is given.

In the best case, fleets are identical and thus a complex clustering of sub-fleets is not required. Identical fleets are characterized by identical features and working conditions. In practice identical fleets are rare, because outside of the laboratory, identical working conditions are hard to maintain and even small differences in the environment can have a huge impact on the degradation pattern of the systems. It is not surprising that research of identical fleets often happens in a laboratory environment, such as the work of HUBBARD ET AL. that examined eight Li-ion prismatic cells that are used in medical devices (2016). The devices were hermetically sealed and tested under constant temperatures (37°C), charge and discharge rates and times (Hubbard et al. 2016, p. 5). GEBRAEEL AND LAWLEY obtained data from an identical fleet of bearings that were tested in testing setup under a constant load of 200lbs and a rotational speed of 2,200rpm (Gebraeel and Lawley 2008, p. 156). A similar system was also used earlier (Gebraeel et al. 2004, p. 695). HUANG ET AL. independently setup another bearing test rig including an automatic oil circulation system that regulates lubrication and temperature. Additionally a constant load of 1205kg is applied (2007, p. 202).

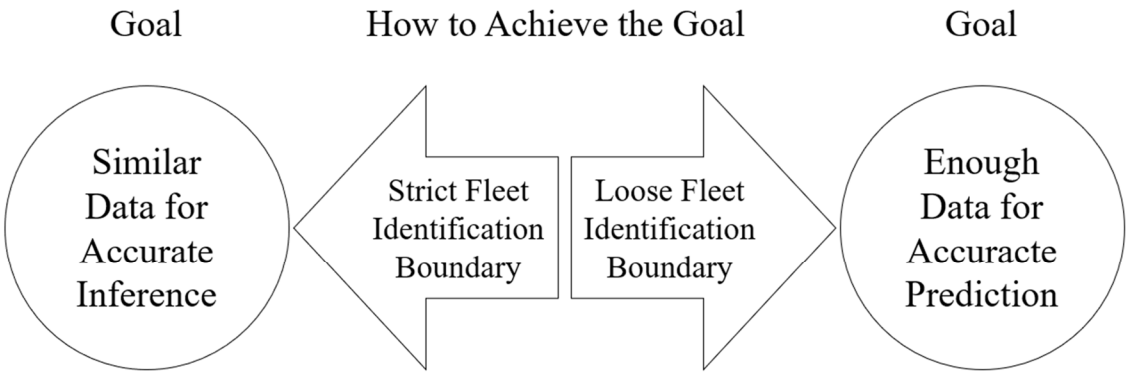
3.3 Stage 1: Fleet Identification

Depending on the fleet type, different algorithms were applied to find similarities between systems of the overall fleet. If systems in a fleet are alike other systems, they can be added to a common sub-fleet. For instance, after identifying the similarities, different models can be trained for each sub-fleet group. The challenge is to find similar systems so that two contrary goals are fulfilled: 1) The grouping of systems must be chosen strictly enough, so that the similarities between their degradation data is of a high level. This

makes it possible to predict future behavior of a system by looking at past data from systems of the same sub-fleet. 2) The grouping of systems must be chosen loosely enough, so that there is sufficient data for the algorithms. The author calls this the “Fleet Paradox” (cf. Fig. 4). In most of the gathered literature, many different strategies are employed to find a compromise between the two goals. Methods either identify sub-fleets by identifying categorical, numerical or semantic similarities or they presume that all input data already stems from an identical or suitably similar fleet (cf. Fig. 3).

The latter case is mostly used in literature. 19 of the 49 articles presumed that the degradation data that was used, belonged to fleets whose systems follow a similar distribution and thus no identification was necessary. This assumption is typical for identical fleets (7 out of 21), but also for many homogeneous fleets it was assumed that all systems of the examined fleet can be used to infer behavior of other systems of the fleet. RAZAVI-FAR ET AL. assumed that a predictor for four homogeneous lithium-ion batteries can be trained by the leave-one-out principle, where three batteries are used to build the predictor for the left-out battery (2018, p. 6). It was assumed that the degradation pattern of the one battery could be inferred by the three other trajectories, even though the batteries do not have identical working conditions (Razavi-Far et al. 2018, p. 3). Another work examined homogeneous lithium-ion batteries that were employed under different states of charge, temperatures and currents (Nuhic et al. 2018, pp. 41–42).

When the assumption of similarity between fleet degradation does not hold, more sophisticated algorithms for fleet identification must be employed before a prediction model can be build. This is required for heterogeneous fleets, but also many homogeneous fleets comprise structurally identical components, that nevertheless exhibit a completely different degradation behavior due to their varying working conditions. The common techniques for identification of the sub-fleet, when similarity cannot be presumed, are: 1)



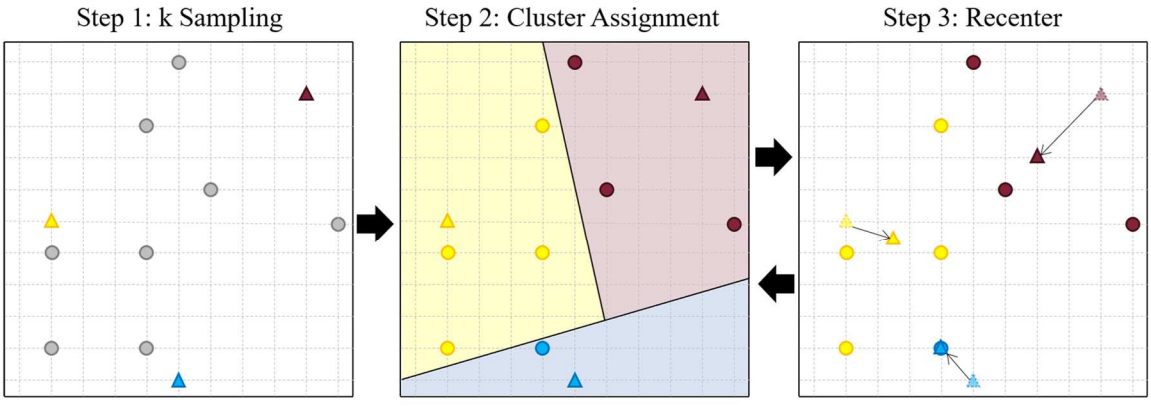
Own depiction

Fig. 4 The Fleet Paradox

Identification by categorical features (e.g. geographical location, manufacturer). 2) Identification by numerical features. 3) Identification by semantics.

Following are some examples of identification by categorical features which were used in eleven publications. VORONOV ET AL. clustered 56,163 trucks into five classes based on the geographical locations (Voronov et al. 2018, p. 625). FRISK ET AL. used the same principle on 33,603 vehicles (2014, p. 1). Another method derived the sub-fleet of oils by categorizing vehicles by harsh and typical environments, which was also derived by the geographical location (Wolak 2018, p. 1). JORDAN ET AL. identified groups of photo-voltaic converters based on three technologies: heterojunction, interdigitated backcontact and all other x-Si (2018, p. 530). WANG ET AL. classified bearings of rotating machinery by their production batch (2018, p. 213). POOT-GEERTMAN ET AL. constructed a degradation prognostic model for two types of trains: four and three coach trains (Poot-Geertman et al. 2015, p. 1869). DUONG ET AL. (2018) and DUONG AND RAGHAVAN (2019) separated two classes of homogeneous LEDs by two different sets of working conditions, defined by temperature and electric current (Duong et al. 2018, p. 82). It is notable that the mentioned categorical methods (n=12) were all applied to homogeneous fleets. Only one of the 13 publications used categorical identification on heterogeneous fleets; this was done by examining multiple different categories, such as component type, manufacturer, model and unit type (Lukens and Markham 2018, p. 6).

The second grouping method, identification by numerical features, is different. Here, 13 publications were identified, where ten dealt with homogeneous and seven with heterogeneous fleets. Multiple methods were applied to the PHM 2008 dataset (Saxena and Goebel 2008), that contains three operating condition signals (altitude, Mach number and throttle resolver angle). RIGAMONTI ET AL. applied a fuzzy c-means algorithm to the three signals (2016, p. 5). The c-means algorithm was introduced by BEZDEK ET AL. (1984) and is an extension of the crisp k-means algorithm. K-means is depicted in Fig. 5 and



Own depiction

Fig. 5 Steps of k-Means

comprises three steps. 1) A suitable number of clusters (k) are defined, and the same number of centroids (triangles) are sampled. 2) Each observation (sphere) is assigned to its nearest cluster. 3) The centroids are shifted to the average of their assigned observations and step two and three are iterated until an Equilibrium is attained. C-means' only difference is, that instead of being assigned to only one cluster, the observations are gradually assigned to multiple clusters, depending on the distance to each centroid. The blue observation in step two of Fig. 5, might then have a gradual assignment of 0.7, 0.2, 0.1 to the blue, yellow and red centroid, respectively. The c-means algorithm was able to identify six distinct operating conditions (Rigamonti et al. 2016, p. 5). The same operating conditions were also identified by PEEL through a Neuroscale mapping that is a multidimensional scaling technique based on Sammon mapping through a radial basis function neural network and through 3D visualization (2008, p. 2). The technique of PEEL was also used by LIM ET AL. (2014, p. 3). HU ET AL. identified sub-fleets of power transformers by clustering nine geometries and material properties, such as wall thickness, angular width of support joints and density of joints (2012, p. 129).

For heterogeneous fleets, fleet identification is a definite requirement, because degradation patterns vary a lot. The k-means algorithm was used this time for an eight-station flexible manufacturing system that comprises thousands of different parts. Components were characterized by multiple failures and substitutions and clustered by their mean-time-between failure and number of corrective and preventive maintenance actions (Kammoun and Rezg 2018, p. 1100). LEONE ET AL. used the Kolmogorov-Smirnov test to test if two degradation trajectories come from identical distributions and thus are grouped into the same fleet (2017, p. 166).

One single approach identified sub-fleet through semantics. The ontology based approach groups diesel engines with semantically similar features, such as engine models "Wärtsilä RT-flex 50" and "Wärtsilä 12V38" (Voisin et al. 2013, p. 7).

It is notable that fleet identification algorithms can be used as a standalone part of a method most of the time. In some cases, it might be easy to detach the identification part of the algorithm (e.g. k-means) and attach it to another prognostic method. An exhaustive list of sources and their fleet identification method can be found in Appendix C.a.

3.4 Stage 2: Fleet Usage

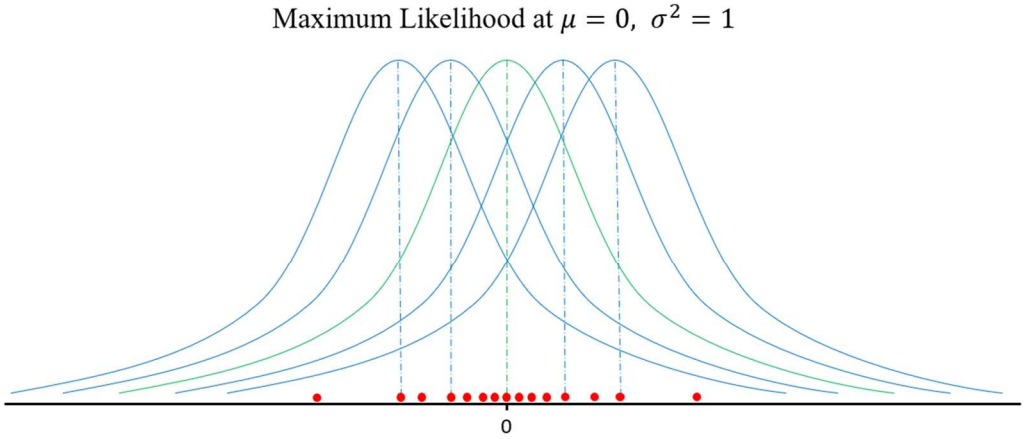
After the fleets have been identified, they are utilized to construct prognostic models in stage two. The identified components can be classified into six categories that can also be seen in Fig. 3: 1) Parameter estimation 2) Training on complete fleet data 3) Fleet as

feature 4) Regression aggregation 5) Fleet normalization 6) Other. An exhaustive list can be found in the Appendix C.b.

Parameter estimation. The most used method (n=17) is suitable for regression algorithms that work with parametrized functions or distributions. These include Bayesian statistics, reliability functions, regression analyses, as well as Markov models.

A simple to understand method to estimate the parameters is maximum-likelihood estimation (MLE). It is a technique to estimate the underlying distribution of a randomly distributed sample (Sutharssan et al. 2015, p. 3). A simple example can be described with a sample of observations as seen in Fig. 6 (red dots). Here it is assumed that the sample may follow a Gaussian distribution with parameters mean μ and variance σ^2. Note that the Gaussian distribution is typically not used as a degradation function, but sometimes used as an intermediate function (Zaidan, Harrison, et al. 2015; Zaidan, Relan, et al. 2015; Zaidan et al. 2016), however it serves as a good example. The Gaussian distribution is laid over the observations with a varying μ and σ^2 and the probability that the observations occur under this distribution are calculated. In the case of Fig. 6, a Gaussian distribution $\sim\mathcal{N}(\mu = 0, \sigma^2 = 1)$ maximizes the likelihood (colored in green). Maximum-likelihood estimation is not the only parameter estimation method. SUTHARSSAN ET AL. describes maximum-a-posteriori estimation and expectation maximization as parameter estimation and Gaussian mixture modeling and Parzen-Window as density estimation methods (2015, p. 4). In fleet prognostics, all observations of the fleet are used for estimation.

ZAIDAN ET AL. constructed a Bayesian network that uses one unit prior distribution and three fleet priors (2016, pp. 126–127). The single unit prior uses a Gamma distribution, the fleet priors are Gaussian, Wishart and Gamma distributions. For specification of the parameters, MLE is used (Zaidan et al. 2016, p. 129). WANG ET AL. constructed a Weibull reliability distribution by log-likelihood estimation. LUKENS and MARKHAM used the



Own depiction

Fig. 6 Maximum Likelihood Estimation

Kaplan-Meier estimate, which is a specialized estimator to construct the reliability function from lifetime data (2018, p. 7). Parameter estimation is also used for regression analyses. WOLAK calculates a simple linear regression function with parameters slope a and intercept b (2018, p. 7). WANG established a Wiener process and also estimated its parameters by MLE (2018, p. 219). HUBBARD ET AL. used bilinear kernel regression, a non-parametric kernel estimator, that was trained via 8-fold cross validation (CV) on all identified fleet data (2016, p. 6).

Training on complete fleet data. A straightforward fleet usage approach is to train a model on the complete sub-fleet data that was identified in stage one. ML approaches can learn patterns from the input data. NNs for instance, in the strict sense, do nothing else than a parameter estimation of a highly complex, non-linear function.

Multiple approaches used a relevance vector machine that was trained on all data of the previously identified fleet (Baptista et al. 2016; Nicchiotti and Rüegg 2014; Nuhic et al. 2018). When training on complete fleet data, it is important to minimize computational effort as the model must be retrained on all observations again. NUHIC ET AL. (2018, p. 12) used an incremental approach that only retrains when the prediction error of the last prediction was higher than a certain threshold (i.e. the trained model is deemed to be no longer explanatory for newly incoming data). An extreme learning machine was also trained on three batteries to predict the RUL of a fourth battery (Razavi-Far et al. 2018, p. 8). Random forests are also able to train on complete fleet data as can be seen in VORONOV ET AL. (2018) and FRISK ET AL. (2014).

Fleet as feature. Eight approaches used the fleet as an additional feature for training input. The fleet can only be added to the model, if it is able to learn how to incorporate the fleet information and how it affects the degradation behavior. It is not surprising that the eight approaches were exclusively NNs. PENG ET AL. (2012) and HEIMES (2008) identified six different working conditions and added the time the engine spent in each working condition as a feature. TRILLA ET AL. added a flag for the position of brakes in trains of the British Rail Class and fed it into the NN (2018, p. 4).

Regression aggregation. While previous approaches made use of the fleet by constructing a common fleet predictor, eight other approaches construct single predictors for each historic degradation profile and aggregate the regression afterwards. VOISIN ET AL. trained RVMs for each trajectory (2013). Each model of past trajectories predicts the RUL for one focal system and the predictions are aggregated as a weighted average, where RVMs of systems which have a similar health index to the focal system at prediction time are weighted more (Voisin et al. 2013, p. 6). WAYNE and ARRES constructed Gaussian

process regressions for each trajectory which were used as predictors for the focal system (2013) and the predictions are simply averaged (Wayne and Arres 2013, p. 709).

Fleet normalization. Fleet normalization is also called multi-regime normalization if the fleet's distinction criteria is the operating condition (Wang 2010, p. 68) and was used by six of the reviewed articles. Data normalization per operating condition was exclusively used in conjunction with appending the fleet as a feature for neural networks (Babu et al. 2016; Hu et al. 2012; Lim et al. 2014; Peel 2008; Riad et al. 2010; Rigamonti et al. 2016). This makes sense, because the different working conditions had a huge impact on the degradation signals.

Other. Lastly, there were three other fleet usage approaches that did not fit into any category. For instance, one approach determined the degree of degradation by k-means clustering, where the centroid distance to a health component was calculated; the farther away the component from the healthy centroid, the higher the degradation ratio (Kammoun and Rezg 2018, p. 1098). The interested reader is also referred to (Bracke and Sochacki 2015; Palau et al. 2019) for further approaches.

Finally, it must be noted that fleet usage methods are heavily dependent on the employed prognostic methods that will be described in the next Section.

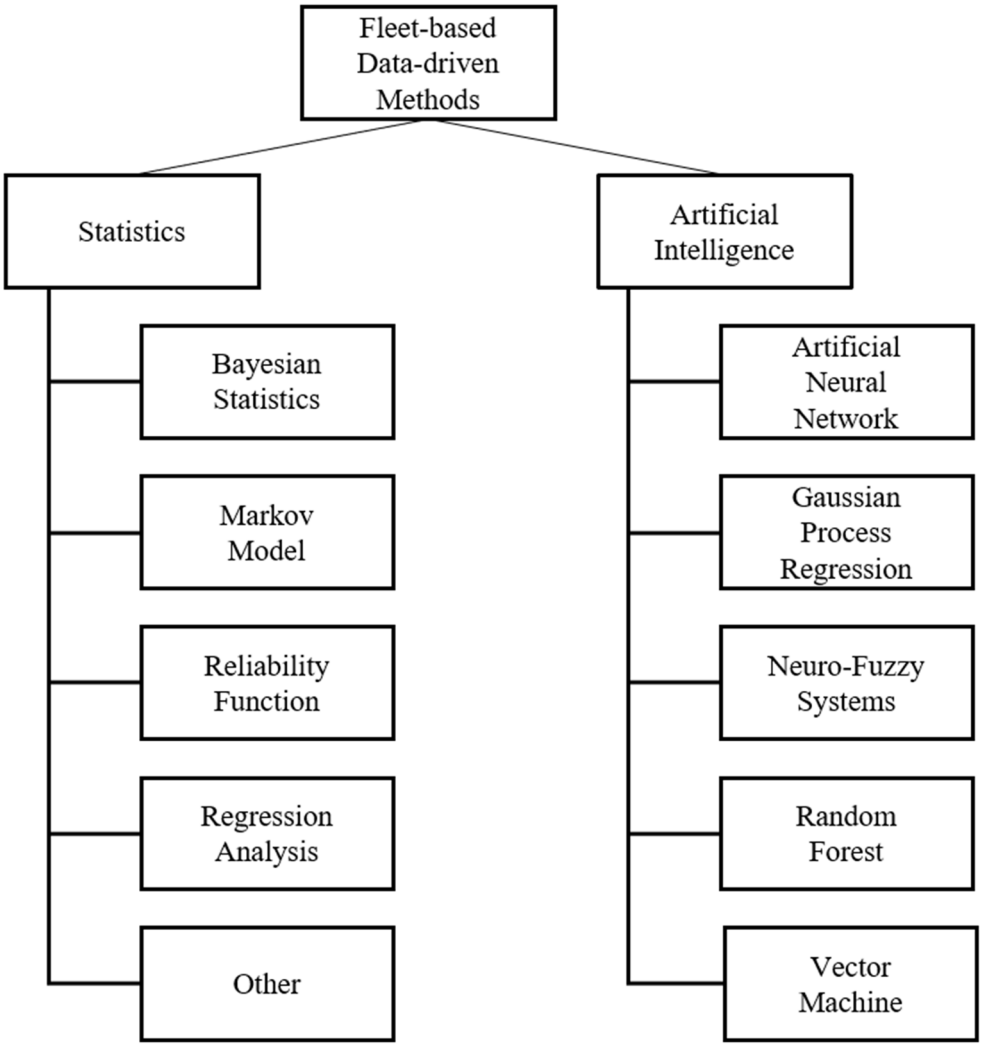
3.5 Stage 3: Prognostics

Data-driven prognostic methods have already been introduced in Section 2.1.3 by looking at review articles from the domain of PHM. All the previously described prediction methods, except for NFS, could also be identified within the literature review, i.e. most methods were adapted to make use of fleet data by at least one publication.

All algorithms can again be separated into statistical (a total of 22 publications) and artificial intelligence (a total of 21 publications). Additionally, there were six ensemble methods (two comprising only AI methods and four comprising statistical as well as AI methods). All types of prognostic techniques that were considered for the decision model can be seen in Fig. 7 and an exhaustive list of all 49 publications and their general class (i.e. Markov model, Vector machine etc.) can be found in Appendix C.c, C.d and C.e. In the following Section it is described how the fleet-based prognostic methods work.

Statistical Methods

Bayesian Statistics. ZAIDAN, HARRISON, ET AL. introduced a Bayesian hierarchical model (BHM) that works with three fleet prior distributions and a single unit prior distribution (2015). The fleet prior is used for parameter estimation as explained in Section 3.4 and



Own depiction

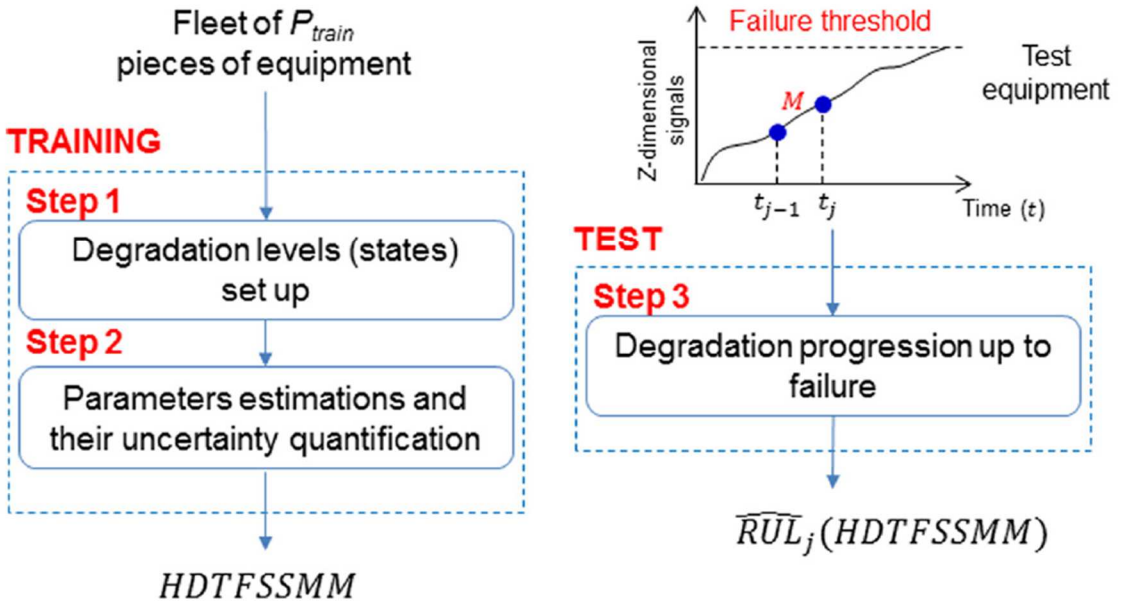
Fig. 7 Regression Methods of the Decision Model

the parameters and single unit prior are then used via Bayes' rule to derive the posterior distribution that can be used for RUL calculation (Zaidan, Harrison, et al. 2015, p. 544). The posterior distributions are stochastically computed using a Gibbs sampler and a Monte Carlo (MC) simulation (Zaidan, Harrison, et al. 2015, p. 545). ZAIDAN ET AL. improved the algorithm in another publication, by adding a change-point detection (2015). The direct-density-ratio-based change point detection is used to apply two different degradation priors. One is used for normal degradation and one is used for accelerated behavior after a specific event. The algorithm can also cope with maintenance events that renew the system by resetting the Bayesian network (Zaidan, Relan, et al. 2015, p. 8475). While the first two publications used the Gibbs sampler and MC simulations that are computationally expensive, ZAIDAN ET AL. improved the algorithm by using variational Bayes to approximate the degradation curves (2016). The new network also uses an additional fleet prior (Zaidan et al. 2016, pp. 126–127). Ling and Mahadevan used another

Bayesian network that works with MC simulations to address “physical variability, data uncertainty and model uncertainty” (Ling and Mahadevan 2011, 2012, p. 103).

Reliability Functions. Reliability functions were also adapted to make use of fleet data. The Weibull distribution was used in a case study for brake pads of trains and was estimated through inspection observations each 60,000km (Poot-Geertman et al. 2015, p. 1866). WANG ET AL. also used the Weibull distribution to estimate the RUL and additionally added a normally distributed confidence interval to the estimations (2012, p. 90). LUKENS and MARKHAM used the Kaplan-Meier estimate to derive a survival function for each fleet that was grouped by site (2018, p. 8). The advantage of the Kaplan-Meier estimate is, that it is non-parametric and can work with right-censored data (Lukens and Markham 2018, p. 7). Another publication made use of the Cox regression where a baseline function is multiplied by a covariate term with two covariates derived from measurements of two chemical values to estimate wheel motor failure (Jardine et al. 2001).

Markov Model. A Markov model, called homogeneous discrete-time finite-state semi-Markov model (HDTFSSMM) was used for heterogeneous fleets by AL-DAHIDI ET AL. (2016). The same model was also used in an ensemble of two methods within two separate publications (Al-Dahidi et al. 2017a, 2017b) and consists of three steps that can be seen in Fig. 8 (Al-Dahidi et al. 2017b, p. 4). First a sub-fleet data set, that was grouped in the fleet identification stage, is fed into the model. Then the number of Markovian states is determined by an unsupervised ensemble clustering technique that is based on k-means. The goodness of the solution is measured by the Silhouette index (Al-Dahidi et al. 2016,



(Al-Dahidi et al. 2017b, p. 4)

Fig. 8 Flowchart of the HDTFSSMM

p. 112). Through the fleet data, the transition probabilities are calculated with a discrete Weibull distribution and a log-likelihood estimation of the parameters. The probabilities are used to forecast the degradation after multiple steps through a significant number of Monte Carlo simulations (Al-Dahidi et al. 2017b, p. 4). In another study the number of degradation states was assumed to be fixed and the degradation pattern assumed to follow an exponential distribution. A particle filter was used to reduce noise (Raghavan and Frey 2016, p. 2).

Regression analysis. One of the simplest regression analysis methods is the linear regression. Linear regression was applied in one case for five different oil fleets to estimate the total base number, which is a health indicator of oils (Wolak 2018, p. 8). LEONE ET AL. extract health indicators of a fleet at each times step and derive a concatenated regression function through stochastic four-step Monte Carlo simulations (2017, pp. 4–5). Another study used bilinear kernel regression, which is a non-parametric method using kernels. Kernels are set of weighted identical functions that are adjusted in a training phase, so that a best-fit regression line constructed (Hubbard et al. 2016, p. 2).

LIU ET AL. introduced the notion of a match matrix (2007). A match matrix is an enhanced ARMA model that utilizes historical data from various operations (Kan et al. 2015, p. 3). A match matrix examines the pairwise similarities of two degradation trajectories. Each trajectory is compared via the Mahalanobis distance and each pair is stored in one matrix (Liu et al. 2007, p. 559). The means of the best matches are then input into an ARMA model (Liu et al. 2007, p. 561).

Another study uses a generalized nonlinear-drift-driven Wiener process that uses parameter estimation for the drift term λ, the diffusion coefficient σ and the Brownian motion B(t) (Wang et al. 2018). The RUL is expressed as probability density function (PDF), estimated by a Bayesian rule and is update at each new entry (Wang et al. 2018, pp. 218–219).

Other. Other algorithms could not be classified into traditional statistical techniques. A study used a multivariate k-means algorithm to cluster components and derive their health coefficient by measuring its distance to a new and old component (Kammoun and Rezg 2018, p. 1100), i.e. a component with a distance of zero to the “best-component” is 100% healthy. The RUL is then calculated by multiplying the degradation coefficient with the estimated useful life of the component which is provided by the manufacturer (Kammoun and Rezg 2018, p. 1104).

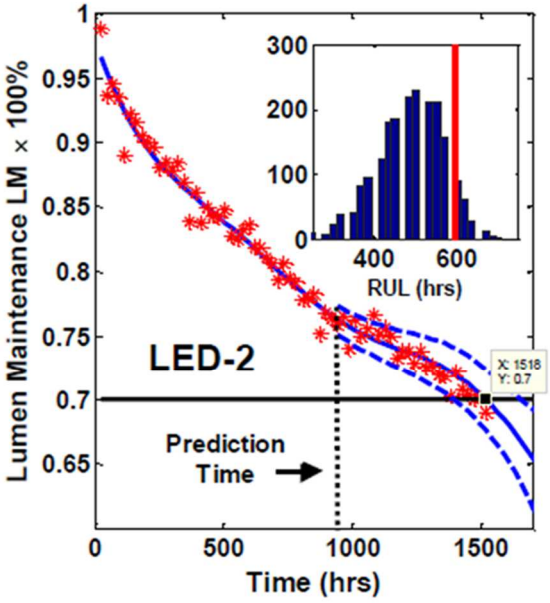
BRACKE and SOCHACKI use the “Risk Analysis and Prognosis of Complex Products (RAPP)”, a portfolio of multiple statistical methods, such as correlation coefficients by

Spearman and Kendall, Weibull distributions, regression analysis, coefficient determination and distribution models (2014, p. 1143). Further approaches use year on year degradation rate calculations (Jordan et al. 2018) and social asset networks that exchange real-time information (Palau et al. 2019).

Artificial Intelligence Methods

Gaussian Process Regression. Three publications make use of GPR, which is a Bayesian ML technique. Two studies extended a single-output GPR to incorporate fleet data from multiple LEDs by assuming that different LEDs with similar conditions follow similar patterns (Duong et al. 2018). This technique was called multi-output (Duong et al. 2018, p. 80) or multi-task (Duong and Raghavan 2019, p. 1182) GPR and makes use of fleet system time series by treating multiple LEDs as having a single time series and a common covariance function that represents the correlation between multiple LEDs (Duong and Raghavan 2019, p. 1183). An exemplary process can be seen in Fig. 9 where a prediction is made for one LED, by training the GPR on a full time series of LED-1 and partial data from LED-2 up to 920h. The histogram inset represents the predicted RUL at each time step (blue) and the true RUL (red) and shows promising results in regards to the small fleet of two LEDs that was used (Duong and Raghavan 2019, p. 1184). Wayne and Modarres also tested another GPR that was trained on data from a fleet of 1392 vehicles (Wayne and Arres 2013, p. 707).

Artificial Neural Networks. With 13 publications, ANNs are the most used class for fleet-based prognostics, presumably because ANNs can be used for a multitude of prognostic



(Duong and Raghavan 2019, p. 1184)

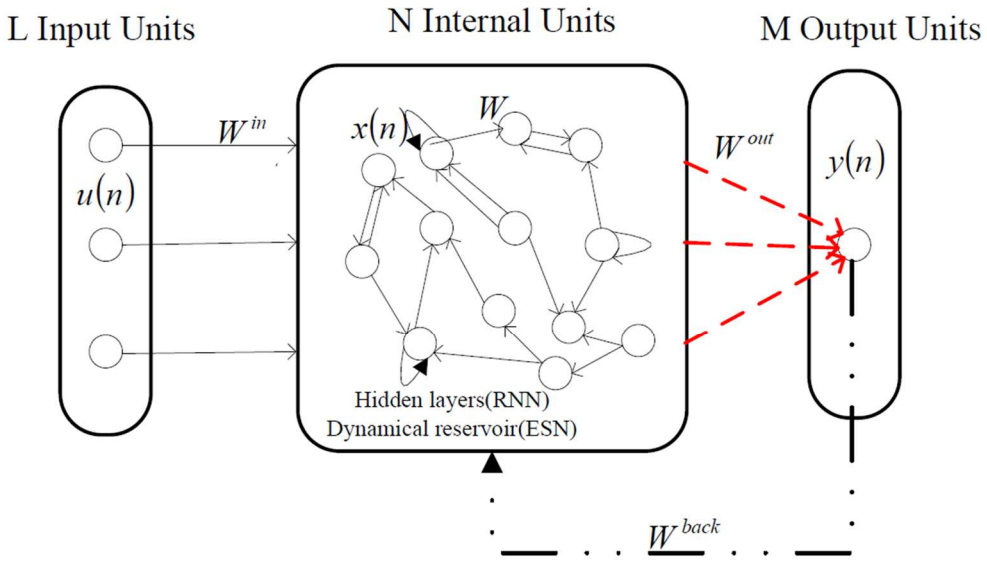
Fig. 9 Exemplary GPR Prediction

tasks, such as time series prediction, exponential projection or data interpolation (Heng et al. 2009, p. 727). ANNs for RUL prediction rose to popularity, due to the PHM 2008 challenge that worked on a fleet of homogeneous unknown systems with six different working conditions. Six publications found in the literature review constructed ANNs to solve the challenge (Babu et al. 2016; Heimes 2008; Hu et al. 2012; Lim et al. 2014; Peel 2008; Riad et al. 2010). The winning algorithm used an ensemble of a radial basis function (RBF) and multi-layer perceptron (MLP) networks (Peel 2008, p. 5) that were filtered by a Kalman filter. HEIMES placed second in the challenge and utilized a single recurrent network (RNN) with three layers of feed-forward and recurrent connections that significantly increased in accuracy compared to a single MLP network (Heimes 2008, p. 5).

Four further ANNs were applied to identical fleets of bearings (Gebraeel et al. 2004; Gebraeel and Lawley 2008; Huang et al. 2007; Shao and Nezu 2000). In comparison to the PHM challenge ANNs, GEBRAEEL and LAWLEY constructed a single feed-forward backpropagation network for each fleet unit that is aggregated via weights, depending on the average squared error of each network within the training (2008, p. 158). The work is an extension of (Gebraeel et al. 2004).

A special type of RNNs are echo state networks (ESN) that were used in two publications (Peng et al. 2012; Rigamonti et al. 2016). Instead of a hidden layer it uses a dynamic reservoir as can be seen in Fig. 10. Instead of optimizing weights from each hidden node to each output node, a reduced set of weights (W_{out}) must be calculated (Peng et al. 2012, p. 2), that increases training speed.

A further approach used extreme learning machines that randomly generate hidden nodes and thus offer a fast and low complex method for RUL prediction (Razavi-Far et al. 2018,



(Peng et al. 2012, p. 2)

Fig. 10 ESN Architecture

p. 3). RIGAMONTI also used a type of ANN, called self-organizing maps (Rigamonti et al. 2018).

Fuzzy Logic and Neuro-Fuzzy Systems. Fuzzy logic can be quite powerful when working with real-world problems of high uncertainty. AL-DAHIDI ET AL. constructed an ensemble that makes use of a fuzzy logic approach called fuzzy similarity-based (FSB) model (2017a). Through complete RTF histories, the FSB model can predict the RUL of a test trajectory through four steps (Al-Dahidi et al. 2017a, p. 4): 1) The Euclidean distances (dissimilarities) between signals of the test trajectory and each complete reference trajectory are calculated. 2) The fuzzy similarities between the test and reference trajectories are calculated with equation (17), where d is the dissimilarity of step 1. The fuzziness is achieved by the similarity function, that gradually determines the similarity between two trajectories. Through the two parameters α and β, the level of fuzziness can be defined.

$$s = e^{-\left(\frac{-\ln(\alpha)}{\beta^2}d^2\right)} \tag{17}$$

3) Weights are assigned to the reference trajectories. The higher the similarity, the higher the weight. 4) The weighted RUL of all the reference trajectories are summed and used as a RUL prediction for the test trajectory.

Surprisingly, to the author's knowledge, there exists no work on a fleet-based NFS. The interested reader is referred to (Fagang et al. 2009; Mahdaoui and Mouss 2012; Wang et al. 2004), where non-fleet based NFSs are used for prognostics.

Random Forests. Two studies examined the use of random forests in RUL prediction (Frisk et al. 2014; Voronov et al. 2018). FRISK ET AL. trained a RSF with 200 trees on a big problem with over 30,000 vehicles and 30 variables (2014, p. 5). Originally, the problem comprised 291 variables. The variables were increased using histograms that reduce dimensionality by putting distinct continuous values into discrete histogram bins and counting the frequencies. Each bin was used as one variable, leaving the model with 1031 variables (Frisk et al. 2014, pp. 5–6). Because models with these variables could not be computed in reasonable time, feature importance was used to reduce the data to 30 variables. Still, the RSF was able to train its model in 15 minutes on a very large computer (Frisk et al. 2014, p. 5). VORONOV ET AL. extended the work by cross-validating optimal RSF models on 50,000 trucks (2018). It was concluded that a 1000-tree RSF is most appropriate (Voronov et al. 2018, p. 636).

Vector Machines. Four further publications made use of support (Baptista et al. 2016; Hu et al. 2015; Nicchiotti and Rüegg 2014) and relevance vector machines (Voisin et al. 2013). BAPTISTA ET AL. compared a regressive SVM with a reliability function and a

regression analysis method and concluded that the SVM could provide significantly better estimates than the other two methods (2016, p. 8). The trained SVM was very simple and used only a linear hyperplane for regression (Baptista et al. 2016, p. 5). Two further studies used more complex SVMs that were implemented with Gaussian kernels and can handle multivariate non-linear data (Nicchiotti and Rüegg 2014, p. 4; Nuhic et al. 2018, p. 13). VOISIN ET AL. used an RVM that is a generalized linear model of the SVM in a Bayesian form (2013, p. 5). The RVM is trained for each completed RTF time series and retrained whenever a new time series is completed. In contrast to the SVM, the RVM outputs the RUL estimate as a randomly distributed value.

Ensemble Methods

Ensemble methods are a special type of algorithm. They do not fit into the categories of statistical nor AI approaches, because they can comprise multiple methods at once. Six publications employed ensembles to increase robustness of predictions (Al-Dahidi et al. 2016, 2017a, 2017b; Hu et al. 2012; Lim et al. 2014; Peel 2008). Parts of these ensembles were also explained in the previous Sections, such as the ensemble of AL-DAHIDI ET AL. that comprises a Markov model, called HDTFSSMM, and a fuzzy method, called FSB model (Al-Dahidi et al. 2016, 2017a, 2017b). The combination of the different predictions of the models work in a weighted sum model, where weights increase inversely proportional to the logarithm of the normalized validation mean absolute error (MAE) and are corrected by a bias (Al-Dahidi et al. 2017a, pp. 5–6).

A futher study used five different techniques: an RVM, an SVM, an exponential fitting method, an extrapolation-based approach and an RNN (Hu et al. 2012, pp. 122–124). Three different weighting schemes were applied in this study. The schemes increase the weights of ensemble member if they are more accurate, more diverse (i.e. predicting diverse RULs) and more optimized (i.e. predicting with high accuracy and robustness). For that, different optimization functions are minimized (Hu et al. 2012, pp. 125–126).

The previously described winning algorithm of the PHM 2008 challenge employed three NNs aggregated by a switching Kalman filter (Peel 2008). The method has been recreated in 2014 and benchmarked against single MLPs, a Kalman filter ensemble and a Gibbs filter, where its superiority could be validated (Lim et al. 2014).

The introduced algorithms are diverse in nature and as a practitioner, it is hard to evaluate which algorithm yields the optimal results for the available data and the business requirements. In the next Chapter, the introduced methods are generalized and assessed by decision-relevant criteria and a decision model is constructed that can mathematically select the optimal algorithm.

4 The Decision Model

Referring to the used design science approach by HEVNER ET AL. (2004), the next step is the creation of an artifact. After the methods have been identified and synthesized in the previous Chapter, they can now be used to attain the second research objective and build the artifact: the decision model for the selection of fleet-based prognostic methods. In the first Section the methodology is described, while the next Section introduces the reader to the criteria that represent the second essential component of MCDM besides the methods / alternatives that have been identified in the previous Chapter. Section 4.3 presents the decision model that is the principal artifact of this research.

4.1 Methodology

After the methodology of VOM BROCKE ET AL. has been used to identify existing approaches, these approaches must be now synthesized in a decision model

The results of the literature review explained in the previous Chapter were used for the second objective: Dimensions of prognostic algorithms that are decision-relevant for the practitioner are identified from literature that has been gathered in the previous objective and by the help of diverse literature on classification of different algorithms done by AMURADOV ET AL. (2017), AL-DAHIDI ET AL. (2016), YANG ET AL. (2018), KOTSIANTIS (2012), SINGH ET AL. (2016), SAXENA ET AL. (2008) AND VENKATASUBRAMANIAN (2005). The classification articles have been collected in the conceptualization phase (stage two of the literature review reference model of VOM BROCKE ET AL.) by searching for overviews in the domain of prognostics and supervised learning. For that, “sources most likely to contain a summary or overview of the key issues relevant to a subject” (Baker 2000, p. 222) were consulted and the criteria of the acquired literature were implemented in a concept map (2009, p. 2215).

With these results, a decision support model for the selection of appropriate algorithms for a given prognostics scenario is constructed. This is done by assigning the decision relevant dimensions to the previously analyzed algorithms and evaluating how well an algorithm is suited for the different criteria. How well it is suited, is reasoned by drawing upon all previously collected theory and scanning it for qualitative (how did the authors describe the performance of the methods, what were described caveats, etc.) and quantitative specifications by the authors (what was the error, runtime, space requirement of the methods, how did it perform in benchmarks, etc.). The results are constructed in an alternative-criteria matrix where alternatives are placed on the row-axis and the decision-relevant dimensions are depicted on the column-axis. The matrix is then used to construct a actionable, tangible model to evaluate the fitness of an algorithm for a certain PHM

context by combining it with one of the MCDM methods presented in Section 2.2. Decision-making practitioners can easily choose an optimal algorithm for their own business case that is shaped by the available system monitoring data and the business-specific requirements towards an algorithm's characteristics.

4.2 Criteria

There are different types of criteria. One must distinguish between how criteria must be rated for the different MCDM methods and how the criteria can be measured in terms of how well each algorithm performs regarding them.

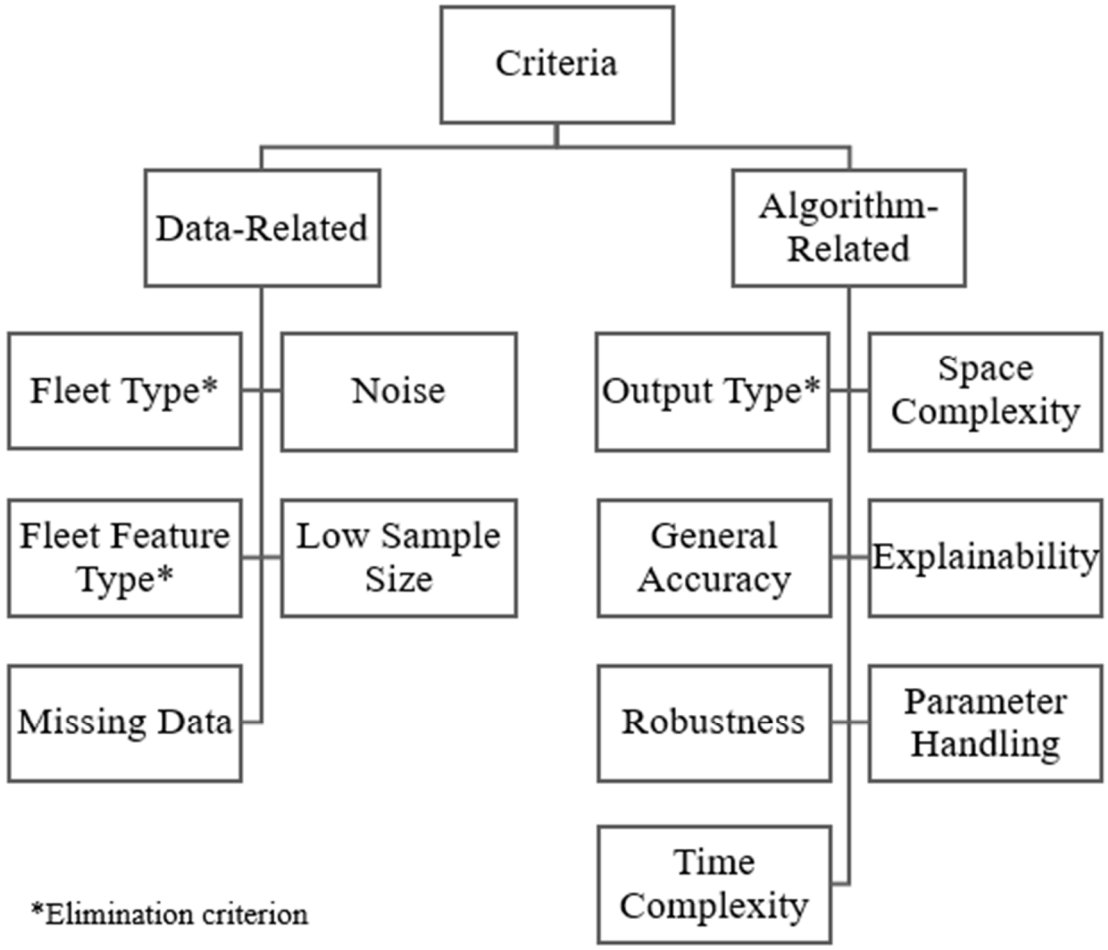
MCDM methods can generally only work on numerical preferences, because e.g. AHP and ANP use eigenvector and weight calculations and TOPSIS uses Euclidean distances. However categorical data can also be integrated by using it for the selection of a subset of the alternatives. For instance, the type of output of the algorithms is a categorical criterion, i.e. point-, interval- or randomly distributed estimate. The user can use this criterion to eliminate alternatives which do not fulfill it and proceed with the regular (numeric) calculation on the selected subset.

In terms of measuring the performance of algorithms, criteria can be classified as qualitative or quantitative. Qualitative criteria, e.g. explainability, are often ordinal and while it can be generally said that one algorithm outclasses another one, it cannot be stated by how much. Quantitative criteria, e.g. general accuracy or robustness, can be exactly measured (Saxena, Celaya, et al. 2008, p. 8).

All in all, three elimination criteria (fleet type, fleet feature type, output type) and nine preference criteria (missing values, noise, few data, general accuracy, robustness, speed, memory efficiency, explainability, parameter handling) were identified. The different criteria are rated by whether the methods presented in Chapter 3 are suited for the data's characteristics, and how well they perform in regard to the algorithm-related criteria. The identified criteria are depicted in Fig. 11.

4.2.1 Data-Related Criteria

ATAMURADOV ET AL. classify methods by two categories: basic requirements and tool efficiency (2017, p. 24). Basic requirements are predefined by the existing business context. For instance, they encompass the nature of the available data (e.g. non-linear, Gaussian). They are often "hard" criteria, because they are more or less dictated by the characteristics of the data. In this research the criteria are called data-related criteria.



Own depiction.

Fig. 11 Criteria Hierarchy

4.2.1.1 Fleet Type (FT)

The fleet type describes the level of hetero- or homogeneity of the underlying data. The methods are rated by looking at what fleet was used for the experiments and by looking at the fleet identification and usage stages (stage 1 and 2). E.g. algorithms that use clustering algorithms to define a sub-fleet out of the whole fleet, perform generally worse for identical fleets, because they try to find differences between identical systems and reduce the relevant data size.

As explained earlier, 21 out of 49 publications used no fleet identification method to find sub-fleets. They thus assume that the measured degradation data of the systems show a high level of homogeneity, either because the observed fleet is identical or highly homogeneous.

The higher the level of variance, the more crucial it is to find similarities between data. Many approaches used features of the fleet as an input for their models. PEEL appended six operating conditions of homogeneous aircraft engines to the data and fed it into a

neural network ensemble (2008, p. 3). This allows the method to use information from all systems, but also account for a differentiation between fleets. PENG ET AL. also used this method in combination with an ESN (2012, p. 3).

For highly heterogeneous fleets and degradation data, stricter cluster methods must be employed. The flexible manufacturing system introduced by KAMMOUN and REZG that comprises thousands of different components had to be clustered more restrictively (2018). Therefore, the k-means clustering technique was used to partition the data by their mean-time-to-failure, and the number of preventive and corrective maintenance (Kammoun and Rezg 2018, p. 1100). Instead that all data was fed into the algorithm, only the data of the same cluster was used further.

Scores: identical (Id), homogeneous (Ho), heterogeneous (He).

4.2.1.2 Fleet Feature Type (FFT)

It is also important to know in which form the fleet characteristics are stored in the data. The fleet feature type is important for the fleet identification (stage 1) explained in Section 3.3.

For instance, studies that employed clustering techniques, such as the k-means or c-means cluster methods (Gebraeel et al. 2004; Kammoun and Rezg 2018; Rigamonti et al. 2016), require the fleet features to be numeric, as clusters are formed based on their Euclidean distances to a centroid. Examples of numeric fleet features are the working conditions of the PHM 2008 challenge data (Saxena and Goebel 2008).

Categorical fleet features are attributes of the system that directly or indirectly explain different degradation patterns. These can characterize the structure of the system as well as its working conditions. Examples for features that characterize the system are component type, manufacturer, model or unit type (Lukens and Markham 2018, p. 6). Examples for working conditions are current and temperature of LEDs, that are stored as discrete mA and °C values (Duong et al. 2018). Additionally working conditions need not only to be numerical, but can also be expressed categorically, such as the job type or the location (Wayne and Arres 2013, p. 707).

When categorical features have too many unique values it is hard to cluster them, because fleets would only contain few systems. In this case, the data can be considered of semantic nature. VOISIN ET AL. (2013, p. 7) tried to extract fleet data through identifying fleets by parts of their semantic data, such as an “engine ref” that contained a lot of different values (e.g. Wärtsilä 12V38, Wärtsilä RT-flex50, Wärtsilä RT-FLEX82T).

In case of the 21 publications where no fleet identification was used, fleet features are generally not existing and all data is used for prognostics.

Scores: Numeric (Num), categoric (Cat), semantic (Sem), none (No)

4.2.1.3 Missing Data (M)

In practical scenarios data is often missing (Lee et al. 2014, p. 329), due to intermittent sensor malfunction or failure (Duong and Raghavan 2019, p. 1182). Whether, and how well an algorithm adapts to missing data is decisive for its success.

Bayesian statistics, such naïve Bayes or hierarchical models can handle missing data successfully. "Naive Bayes is naturally robust to missing values" (Kotsiantis 2007, p. 262), while Bayesian networks "can readily manage incomplete datasets" (Sikorska et al. 2011, p. 1812). This is also facilitated due to smoothing (e.g. Laplace), which adds a probability of an event into the posterior distribution, when the event is not seen in the training data but known to exist.

Further, VORONOV ET AL. implemented a RSF for a dataset with a missing data rate of 40% (2018, p. 625). RFs generally handle missing values (Frisk et al. 2014, p. 4), because the DTs can consider null to be a possible value with its own branch (Sharma 2008, p. 74). Also, regression analysis, as well as reliability functions are well suited for missing values, as their parameter estimation can simply ignore these values. WANG ET AL. even used linear regression in the preprocessing phase to impute missing values (2012, p. 87). A similar principle is applied by GPR and thus this technique can also be used when data are missing (Duong and Raghavan 2019, p. 1186). Fairly good results can also be achieved with Markov models that have a "quick management of incomplete data sets" (Kan et al. 2015, p. 13) and "can readily manage incomplete datasets" (Sikorska et al. 2011, p. 1812).

Proportional hazard models (PrHM) perform moderately and missing data cannot be handled directly (Frisk et al. 2014, p. 4). On the lower end are vector machines, that are only rated with two out of four stars by KOTSIANTIS (2007, p. 263) and NFSs that “need lots of high-quality training data” (Lei et al. 2018, p. 818).

NNs perform worst on missing data, because these are not manageable by the activation functions. NNs generally require complete records to do their work (Kotsiantis 2007, p. 262). The same is valid for the k-nearest-neighbor (kNN) algorithm (Kotsiantis 2007, p. 262) that is used by PALAU ET AL. (2019).

Scores: high (5), moderately high (4) medium (3), moderately low (2), low (1)

4.2.1.4 Noise (N)

In real-life cases the measurement data is often contaminated with noise and random unexplainable variance (Liu et al. 2007, p. 561).

Especially well-suited for high levels of noise are Bayesian techniques. ZAIDAN ET AL. developed a BHM that is suited for poor signal-to-noise ratio (2016, p. 122). Another Bayesian technique, the Kalman filter, can handle Gaussian process or measurement noise (Sikorska et al. 2011, p. 1812) and is for instance used within an MLP ensemble (Lim et al. 2014). Particle filter even eliminate the constraint that noise must be linear or Gaussian (Sikorska et al. 2011, p. 1813).

Decision trees, which are used by random forests, are fairly resistant to noise and a similar level of intolerance is also seen in the kNN algorithm (Kotsiantis 2007, p. 262). HUBBARD ET AL. used a bilinear kernel regression that copes with input data noise (Hubbard et al. 2016, p. 2). GPR can also be adapted to noise through choosing the right kernel function (Wayne and Arres 2013, p. 703).

The noise resistance of vector machines and NNs are only rated with two out of four stars by KOTSIANTIS (2007, p. 263) and signals should possess a low signal-to-noise ratio to guarantee a good performance (Gebraeel et al. 2004, p. 695). RIGAMONTI ET AL. developed a SOM that is susceptible to noise (2018, p. 1310). NFS as a type of NNs also "could not predict well" under noise (Heng et al. 2009, p. 729).

On the lower performing end are regression analyses and reliability functions that are very "sensitive to noise" (Sikorska et al. 2011, p. 1813). For example, a match matrix does not cope well with noise and it must be eliminated before using a PrHM (Jardine et al. 2001, p. 295).

Scores: high (5), moderately high (4) medium (3), moderately low (2), low (1)

4.2.1.5 Low Sample Size (SS)

Instead that degradation models are constructed through knowledge of the physical mechanisms, data-driven methods estimate degradation patterns through historical data. Almost all data-driven prognostic methods need vast amounts of data to work; the reason why fleet-data-driven approaches are facilitated in this thesis.

One of the best methods for "small size datasets" is GPR (Lei et al. 2018, p. 819). GPR generally "performs well with small training datasets" (Kan et al. 2015, p. 7) and DUONG ET AL. were able to successfully employ a GPR that works on only one RTF history and

forecasts the RUL for partial data of a second LED (Duong et al. 2018; Duong and Raghavan 2019).

Furthermore, Bayesian statistics work well with few data. While naïve Bayes is known to require only few data (Kotsiantis 2007, p. 262), Bayesian networks, such as a BHM, even become too big and computationally not feasible with large amounts of data (Kotsiantis 2007, p. 259). A Bayesian network which used only 14 aircraft engines was successfully implemented by ZAIDAN, HARRISON, ET AL. (2015, p. 550).

Moderate performance on few data can be achieved with regression analyses and distributions. While they require a "large sample size to achieve stable results" (Singh et al. 2016, p. 1313), WOLAK was able to use linear regression of natural logarithms with only 96 samples (2018, p. 2) and WANG ET AL. used a Weibull distribution with 54 diesel engines (2012, p. 88).

A moderately large amount of data is required for random forests and also match matrices are “not appropriate unless sufficient historical data from different operation cycles is available" (Lee et al. 2014, p. 324). The same applies for SVMs (Kotsiantis 2007, p. 262), which are still superior to ANNs (Lei et al. 2018, p. 819).

ANNs perform worst on low sample sizes (Kotsiantis 2007, p. 262; Sikorska et al. 2011, p. 1813) and the same is true for its fuzzy counterpart, the NFS (Heng et al. 2009, p. 729; Kan et al. 2015, p. 10). Moreover Markov models also need large volumes of data for training (Lei et al. 2018, p. 817; Sikorska et al. 2011, p. 1812).

Scores: low (5), moderately low (4), medium (3), moderately high (2), high (1)

4.2.2 Algorithm-Related Criteria

Tool efficiency is something that depends on the business requirements (Atamuradov et al. 2017, p. 24). As an example, an often-occurring compromise that must be made is between run time and accuracy. Which criterion is favored depends solely on the requirements the business has towards a prognostic method. They are thus “soft” criteria whose importance can be adjusted to the preference of the company. In this research the criteria are called algorithm-related criteria.

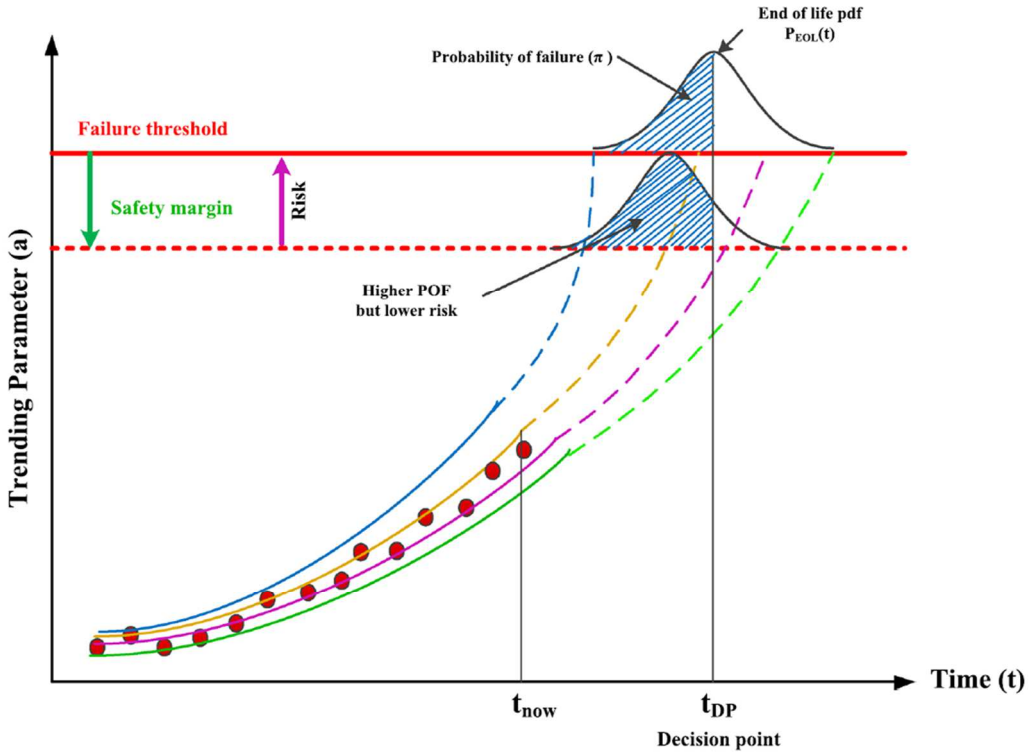
4.2.2.1 Output Type (O)

The output type is another elimination criterion. “The RUL is usually mathematically expressed in three forms: as a point-estimated value, an interval estimated value, and a randomly distributed value based on a normal, log-normal, Weibull, or inverse Gaussian

distribution" (Yang et al. 2018, p. 407). An interval- or distribution-estimate, for instance, allows the practitioner to output a prediction with a confidence level.

GPR outputs a Gaussian probability distribution of the RUL (Kan et al. 2015, p. 6). Also, an advantage of an RVM in contrast to an SVM is, that it can provide probability distributions for the predicted values (Nuhic et al. 2018, p. 17), due to its Bayesian nature. While regression analysis can only output point-estimates, because one function is estimated, some approaches have been adapted to be stochastic, by implementing a random component. LEONE ET AL. derives a RUL PDF randomly sampled Monte Carlo simulations (2017, p. 4). Bayesian networks (and Bayesian approaches in general) are also able to estimate posterior RUL distributions (Zaidan et al. 2016, p. 122). A visual example of how RUL is expressed by distributions can be seen in Fig. 12.

Markov models work by calculating probabilities of transitioning to a final failure state and thus can "provide confidence limits as part of their RUL prediction" (Kan et al. 2015, p. 14; Sikorska et al. 2011, p. 1812). Also the ARMA component that is used in the match matrix fleet approach by LIU ET AL. allows an expression for the variance of prediction errors and can thus yield confidence intervals for the estimate of the mean best match indices (Liu et al. 2007, p. 561).



(Elattar et al. 2016, p. 145)

Fig. 12 RUL Distribution-Estimate

Half of the identified methods only express the RUL prediction as a point-estimate. These comprise SVMs (Kan et al. 2015, p. 13) or Wiener processes (Si et al. 2011, p. 5). Also most NNs cannot provide confidence limits for the output (Sikorska et al. 2011, p. 1813) and the same applies for NFS that output a weighted average of the output layer signals (Fagang et al. 2009, p. 1083). Estimating confidence intervals is also an active research area for RFs (Frisk et al. 2014, p. 9).

Scores: point-estimate (Pt), interval-estimate (I), distribution (D)

4.2.2.2 General Accuracy (A)

While accuracy heavily depends on the use of adequate prognostics model for the characteristics of the data (e.g. missing values, sample size etc.), some approaches generally perform better than others. Accuracy can be measured by e.g. the error or bias (Saxena, Celaya, et al. 2008, pp. 8–10).

The most used and best performing methods are NNs. NNs have a high accuracy, a long prediction horizon (Atamuradov et al. 2017, p. 24) and fare in the top ranks of multiple benchmarks (Caruana and Niculescu-Mizil 2006, p. 165; Kotsiantis 2007, p. 263). Their superiority in accuracy could be confirmed versus a fuzzy similarity based approach (Rigamonti et al. 2016, p. 11), SVMs, RVMs (Babu et al. 2016, p. 225) and ARMA methods (Shao and Nezu 2000, p. 226). Additionally, NNs are often used in ensembles to guarantee even higher accuracy (Hu et al. 2012; Lim et al. 2014; Peel 2008). VMs are also top performers and KOTSIANTIS rates the general accuracy with four out of four stars (2007, p. 263). Also, a study was identified that provided significantly better results with an SVM versus an ARMA model and a Weibull distribution (Baptista et al. 2016, p. 8).

Further highly accurate models are random forests that ranked as the top performing methods in an empirical benchmark case study by CARUANA and NICULESCU-MIZIL (2006, p. 165), while they were only rated with two out of four stars by another study (Kotsiantis 2007, p. 263). Additionally, NFS have "high accuracy" (Kan et al. 2015, p. 10).

Bayesian techniques deliver accurate results, but long term predictions are less reliable (Sikorska et al. 2011, p. 1813). Nevertheless, they can be used when relatively accurate and precise RUL estimates are required (Sikorska et al. 2011, p. 1815). GPR has been proven to perform better than regression models and NNs in some cases (Kan et al. 2015, p. 7), but they should be only used for medium prognostic horizons (Atamuradov et al. 2017, p. 24).

On the lower performing end are regression and reliability functions that could only perform with an average of 63% accuracy in the case study of Caruana and Niculescu-Mizil (2006, p. 165). If high accuracy is required, they should only be used for short-term predictions (Atamuradov et al. 2017, p. 24).

Scores: high (5), moderately high (4) medium (3), moderately low (2), low (1)

4.2.2.3 Robustness (R)

Robustness "signifies insensitivity to small deviations from the assumptions" (Huber 1981, p. 1), i.e. robust prognostics algorithms are less affected by deviations from the norm (e.g. peaks, outliers, etc.). A simple example would be a statistic for central tendency, such as the mean or the median. The former is not a robust measure, because outliers will skew it, while the latter is robust. Robustness can be measured by e.g. the Brier score or the sensitivity (Saxena, Celaya, et al. 2008, pp. 8–10).

The most robust algorithms are ensembles that comprise multiple methods. Ensembles are often employed in the ML community to increase robustness (Hu et al. 2012, p. 121). Lim et al. used an ensemble of MLPs with a different number of hidden neurons (Lim et al. 2014, p. 7), and Hu et al. used two similarity-based interpolation (SBI) models, an RVM, SVM, a Bayesian linear regression and an RNN within an ensemble to improve robustness (Hu et al. 2012, p. 121).

ATAMURADOV ET AL. also classify GPR and NNs as highly robust methods (2017, p. 24), while a high robustness was benchmarked within the study of CARUANA AND NICULESCU-MIZIL for NNs and RFs (2006, p. 5) through measurement of the receiving operating characteristic, which is a common performance indicator for robustness (Saxena, Celaya, et al. 2008, p. 10). NFS are also very robust, "due to the combination of two complementary tools: fuzzy systems and neural networks" (Kan et al. 2015, p. 10). Also, while SOMs have only medium robustness (Atamuradov et al. 2017, p. 24), RIGAMONTI ET AL. could achieve high robustness through multiple classifications (2018, p. 1308).

Besides that, VMs perform moderately well in terms of robustness (Caruana and Niculescu-Mizil 2006, p. 5). VOISIN ET AL. used an RVM that gained robustness through uncertainty management (2013, p. 6). LEE ET AL. and ATAMURADOV ET AL. also rated the match matrix as a method with moderate robust predictions (2017, p. 24; 2014, p. 324).

Regression analyses and hazard functions are performing worst regarding robust predictions (Atamuradov et al. 2017, p. 24; Caruana and Niculescu-Mizil 2006, p. 5).

Scores: high (5), moderately high (4) medium (3), moderately low (2), low (1)

4.2.2.4 Time Complexity (T)

Depending on the prognostics use case, speed can be critical. Speed determines how fast a model can be fitted or trained and how fast the algorithm returns a prediction (less important as predictions are generally in real-time). Of course, speed highly depends on the size of the dataset in terms of observations and dimensions and on the available hardware. Nevertheless, some algorithms work generally faster than others.

PALAU ET AL. developed an social asset prognostic algorithm that calculates its health prediction by exchanging data with its fleet neighbors in real time (Palau et al. 2019). Regression and reliability functions also have a moderately low learning and running time (Atamuradov et al. 2017, p. 24), because only fitting through MLE is necessary.

Because "decision trees are […] quite fast" (Kotsiantis 2007, p. 262), RFs, which are ensembles of decision trees, are the fastest AI method. FRISK ET AL. conducted a 200-tree RF survival analysis on a dataset of 33603 vehicles with 30 variables that takes about 15 minutes on a fairly large setup (2014, p. 5). Also, match matrices have a generally low learning and run time (Atamuradov et al. 2017, p. 24), but can be "very inefficient and time-consuming for large datasets" (Kan et al. 2015, p. 4).

GPR has a high time complexity (Kan et al. 2015, p. 10; Lei et al. 2018, p. 819), mainly due to its long training phase (Atamuradov et al. 2017, p. 24). The time complexity of $O(n^3)$ (Duong and Raghavan 2019, p. 1187) is due to a matrix inverse calculation by MLE and CV (Wayne and Arres 2013, p. 704). Also, while Bayesian networks take less time for training (Singh et al. 2016, p. 1313), they are often combined with MC methods to stochastically compute the posterior distribution, such as by ZAIDAN, HARRISON, ET AL. (Zaidan, Harrison, et al. 2015, p. 545). MC is computationally expensive (Zaidan et al. 2016, p. 123) and often used by approaches to derive RUL probability distributions and make predictions more robust (e.g. Leone et al. 2017; Ling and Mahadevan 2011; Poot-Geertman et al. 2015).

NN and VM perform worst and have a medium to high training time (Atamuradov et al. 2017, p. 24). KOTSIANTIS rated each with one out of four stars (Kotsiantis 2007, p. 263). The complexity can only be surpassed by ensembles that comprise multiple methods, although the training and prediction can potentially be parallelized.

Scores: low (5), moderately low (4) medium (3), moderately high (2), low (1)

4.2.2.5 Space Complexity (S)

The space complexity can also be a crucial factor when choosing the optimal prognostics method. Data-driven approaches naturally work with huge amounts of data and thus already require fundamental amounts of volatile and non-volatile memory. So, it is desired to have a prognostic method that does not increase the required space further. Space complexity has a high positive correlation to time complexity, thus only some remarkable examples are discussed.

Reliability functions and regression analyses neither require a lot of non- nor volatile memory, as only one MLE score per iteration must be stored and the result is stored as a mathematical function.

Match matrices can be very inefficient for large datasets (Kan et al. 2015, p. 4), as the similarity between runs is computed for each pair of runs (Liu et al. 2007, p. 560).

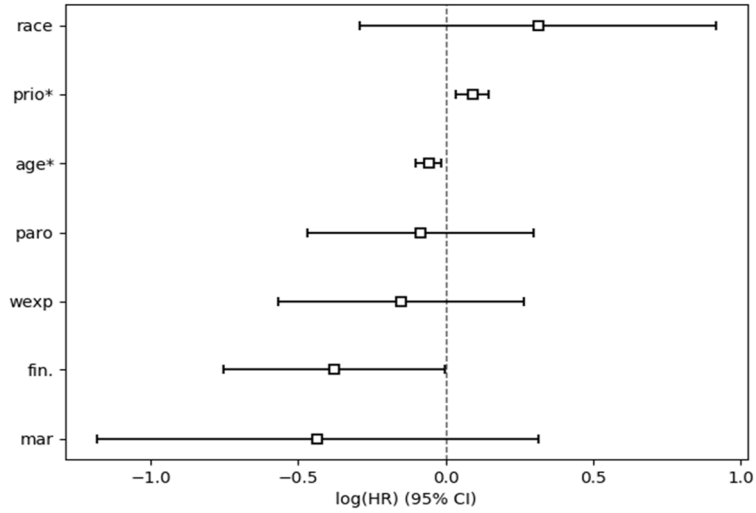
Again, the highest complexity can be found in AI methods. NNs "requires sufficient computational resources" (Lee et al. 2014, p. 323) and Markov models are not suggested, when "suitable hardware for computation is not available" (Sikorska et al. 2011, p. 1814). Also, while the model of FRISK ET AL. was finished in 15 minutes, the computation was executed on a reasonable amount of RAM (Frisk et al. 2014, p. 5) and significant memory resources are generally required for RFs (Voronov et al. 2018, p. 632)

Scores: low (5), moderately low (4) medium (3), moderately high (2), low (1)

4.2.2.6 Explainability (E)

Explainability can be crucial, if maintenance must be justified or reasons for failure should be made transparent. In these cases, a standalone RUL prediction is not sufficient and should be accompanied by a clear explanation for the estimate.

Reliability functions and regression analyses generally have a good explainability, because the shape of the regression or reliability function is fitted to the observations. For instance, JARDINE ET AL. used a Cox regression that is fitted to multi-dimensional data (2001). The general function can be seen in equation (1) and was already explained in Chapter 2. From the function, it is directly visible how the dimensions affect degradation through the coefficients which are exemplarily plotted in Fig. 13. Here, the coefficients are the dots and their lower and upper 95% confidence interval is plotted as a “whisker”. In the exampled. it can be seen, that “race” has the biggest positive, while “mar” has the biggest negative coefficient (i.e. both variables have high positive, respectively negative importance). The life expectancy PDF can also be easily plotted in a 2D graph.



(Davidson-Pilon 2019b)

Fig. 13 Plotted Coefficients of Cox Proportional Hazard Model

While Cox or linear regression is easy to interpret there are some more complicated methods, such as Wiener processes or bilinear kernel regression, which are, nevertheless, moderately well explained. Random forests also show a high level of explainability as “rules can be built from a single decision tree" (Voronov et al. 2018, pp. 626–627).

The midfield comprises GPR that allows to identify physics of the system through its model structure (Kan et al. 2015, p. 6). Also Markov models are “closer to engineering applications” (Si et al. 2011, p. 11) and are characterized by their relative easy model interpretation (Kan et al. 2015, p. 14).

While NNs are not transparent at all, a NFS as a type of NN has a “higher level of transparency and openness" through its rule-based design (Kan et al. 2015, p. 10). Besides NNs, VMs are also the tail light in terms of explainability. They are characterized by optimizing a loss function or margin in a black box manner and thus have notoriously poor interpretability (Kotsiantis 2007, p. 263).

Scores: good (5), moderately good (4) medium (3), moderately bad (2), bad (1)

4.2.2.7 Parameter Handling (P)

Parameter tuning can be very complex. It can be ordinally defined how well the parameters of a method can be handled (i.e. by the number of parameters or the possible values the parameters can adopt). In the best case, the method is nonparametric, in the worst case a lot of parameters must be adjusted, and their values can adopt any real number.

The easiest-to-handle methods are most regression and reliability functions and Bayesian networks where few or no free parameters must be set (Singh et al. 2016, p. 1313). Further there exist some more complex methods, such as Markov models, which are still “simple to develop and implement” (Sikorska et al. 2011, p. 1814), but require identification of the states. Also bilinear kernel regression requires only 2 optimization parameters, the kernel bandwidth and the number of iterations to be set (Hubbard et al. 2016, p. 5).

Moderate effort must be expended for the kernel function, which must be tailored to the data (e.g. noise), and various hyperparameters of GPR (Wayne and Arres 2013, pp. 703–704).

On the low end are NNs that have multiple parameters (Kotsiantis 2007, p. 263) and constructing an optimal model can be time consuming (Sikorska et al. 2011, p. 1814). To decrease parameter complexity (for the price of increased time complexity), RIGAMONTI ET AL. used an evolutionary algorithm for their ESN, where only the search space must be maintained (Rigamonti et al. 2016, p. 10). Lastly, the parameters of VMs are hard to handle and comprise a penalty, kernel coefficient and epsilon-tube that are crucial for the outcome (Nuhic et al. 2018, p. 51).

Scores: easy (5), moderately easy (4) moderate (3), moderately hard (2), hard (1)

4.3 The Decision Model

The 49 publications were classified in a concept matrix. The final decision matrix can be seen in Tab. 4. On the row axis are the different sources and the corresponding class (i.e. the alternatives), on the column axis are the different decision-relevant criteria. The body contains the scores of the alternatives and criteria that have been introduced in the prior Sections. Alternatives and sources that were identical in their criteria were grouped. All in all, the 49 publications could be grouped to 33 distinct alternatives and twelve criteria (nine numeric and three ordinal elimination). It must be noted, that NFSs which have not been found in the fleet-focused literature review, but which have been found in the overview papers of popular prognostic methods, were added at the bottom.

This matrix can now be used as an input for one of the MCDM techniques described in Section 2.2, namely AHP, TOPSIS and ANP. The three elimination criteria are specified for each alternative and can be used to filter subsets. For AHP and ANP, the alternatives and numeric criteria must be rated pairwise. The worst-case amount of comparisons can be expressed with equation (18).

$$\sum_{i=1}^{n-1} i * m + \sum_{j=1}^{m-1} j \tag{18}$$

$\sum_{i=1}^{n-1} i$ is the sum of pairwise comparisons for one criterion, where n is the number of alternatives. m is the number of criteria and $\sum_{j=1}^{m-1} j$ the sum of pairwise comparisons of the criteria weighting. Inserting the numbers into the equation results in $496 * 9 + 36 = 4464$ comparisons. The effort of pairwise rating and computation of eigenvalues for the consistency index is deemed intractable for a practitioner and in fact, the AHP method is only suited for problems with a small number of criteria and alternatives (Chen and Hwang 1992, p. 12). Additionally, the criteria are neither ordered in a more complex hierarchy that would justify AHP, nor do they have considerable reciprocal effects on each other, which would justify using ANP.

On the other hand, TOPSIS is deemed as a good fit for selecting the best fleet-based prognostic method, because it can handle bigger decision problems. As a reminder, the six steps of TOPSIS from Section 2.2.3 are summarized again. A Python program for the decision model can also be found in Appendix D.

Step 1. In this chapter, the preference scores of alternatives in regards to the criteria have been partially presented (297 in total). Now, only the relevance of the criteria must be rated by the DM (twelve in total). The DM must then filter the alternative-criteria matrix (Tab. 4) by its three elimination criteria to get matrix A (cf. equation (14)). Afterwards he/she must construct a weighting vector w (cf. equation (13)) for the remaining nine numeric criteria.

Step 2. The subsetted alternative-criteria matrix A is normalized.

Step 3. The normalized alternative-criteria matrix A is weighted with vector w.

Step 4. The PIS and NIS is created from the weighted A.

Step 5. The distances of each alternative to the PIS and NIS are calculated.

Step 6. The similarity of each alternative to the PIS is calculated. The most similar alternative is the most optimal choice.

Of course, before being able to construct w, the DM must analyze the nature of the data to rate the data-related criteria and must identify the business requirements towards the algorithm-related criteria. This is shown in the next Chapter, where the proposed decision model is evaluated within a case study.

Source	Class	Data-Related Criteria					Algorithm-Related Criteria						
		FT	FFT	M	N	SS	O	A	R	T	S	E	P
(Ling and Mahadevan 2011, 2012)	Bayesian Statistics	Ho	No	5	5	4	D	3	3	2	3	3	4
(Zaidan et al. 2015, 2016)	Bayesian Statistics	Ho	Cat	5	5	4	D	3	3	3	2	3	4
(Zaidan, Harrison, et al. 2015)	Bayesian Statistics	Ho	Cat	5	5	4	D	3	3	2	2	3	4
(Hubbard et al. 2016)	Regression Analysis	Id	No	3	3	3	D	2	3	4	4	4	4
(Voronov et al. 2018; Frisk et al. 2014)	Random Forest	Ho	Cat	4	4	2	Pt	4	4	3	1	4	3
(Duong et al. 2018; Duong and Raghavan 2019)	Gaussian Process Regression	Ho	Cat	4	2	5	I	3	4	2	3	3	3
(Wayne and Arres 2013)	Gaussian Process Regression	Ho	No	4	2	5	I	3	4	2	3	3	3
(Lukens and Markham 2018)	Reliability Function	He	Cat	3	1	3	Pt	1	1	4	5	5	5
(Wolak 2018)	Regression Analysis	Ho	Cat	3	1	3	Pt	1	1	4	5	5	5
(Wang et al. 2012; Jardine et al. 2001)	Reliability Function	Ho	Cat	3	1	3	D	1	1	4	4	5	5
(Bracke and Sochacki 2015)	Other	Ho	No	3	1	2	Pt	1	1	4	5	5	5
(Kammoun and Rezg 2018)	Other	He	Num	4	1	3	Pt	2	3	4	4	4	2
(Jordan et al. 2018)	Other	Ho	Cat	1	1	2	Pt	1	1	5	5	5	5
(Liu et al. 2007)	Regression Analysis	Id	Num	3	1	2	I	2	3	3	3	4	5
(Wang et al. 2018)	Regression Analysis	Ho	Cat	4	1	1	D	2	2	4	4	3	5
(Salvador Palau et al. 2019)	Other	He	Num	1	4	5	Pt	2	2	5	5	1	1
(Voisin et al. 2013)	Vector Machine	He	Sem	2	4	2	D	4	4	1	1	3	4
(Leone et al. 2017; Subbiah and Turrin 2015)	Regression Analysis	He	Num	3	1	3	D	2	1	3	3	4	3
(Poot-Geertman et al. 2015)	Reliability Function	Ho	Cat	3	1	3	D	2	1	3	3	4	3
(Hoffman 2009)	Reliability Function	Ho	No	3	1	3	D	2	1	3	3	4	3
(Raghavan and Frey 2016)	Markov Model	Id	No	4	2	1	I	2	3	2	1	3	4
(Al-Dahidi et al. 2016, 2017a, 2017b)	Ensemble	He	Num	4	2	1	D	2	4	1	1	3	4
(Teixeira et al. 2015)	Markov Model	Ho	No	4	2	1	I	2	3	2	1	3	4
(Peng et al. 2012; Heimes 2008; Trilla et al. 2018)	Neural Net	Ho	No	1	3	1	Pt	5	4	1	2	2	1
(Gebraeel and Lawley 2008; Huang et al. 2007; Gebraeel et al. 2004; Shao and Nezu 2000)	Neural Net	Id	No	1	3	1	Pt	5	4	1	2	2	1
(Hu et al. 2012)	Ensemble	Ho	Num	1	4	1	Pt	5	5	1	1	1	1
(Lim et al. 2014)	Ensemble	Ho	Num	1	4	1	Pt	5	5	1	1	1	1
(Rigamonti et al. 2016; Babu et al. 2016; Riad et al. 2010)	Neural Net	Ho	Num	1	3	1	Pt	5	4	1	2	1	1
(Rigamonti et al. 2018)	Neural Net	Ho	Cat	1	3	1	Pt	4	3	3	2	1	1
(Peel 2008)	Ensemble	Ho	Num	1	3	1	Pt	5	5	1	1	1	1
(Nuhic et al. 2018; Baptista et al. 2016; Nicchiotti and Rüegg 2014)	Vector Machine	Ho	No	2	2	2	Pt	5	3	1	1	1	1
(Razavi-Far et al. 2018)	Neural Net	Ho	No	1	3	1	Pt	5	4	1	1	1	1
(Fagang et al. 2009; Soualhi et al. 2014; Wang et al. 2004)	Neuro-fuzzy System	No	No	2	2	1	Pt	4	4	1	1	2	1

Legend: *FT = Fleet type, Ho = Homogeneous, Id = Identical; FFT = Fleet feature type, No = None, Cat = Categorical, Num = Numeric; M = Missing data; N = Noise; SS = Sample size; O = Output type, D = Distribution, Pt = Point-estimate, I = Interval; A = General accuracy; R = Robustness; T = Time complexity; S = Space complexity; E = Explainability; P = Parameter handling*

Own depiction.

Tab. 4 The Alternative-Criteria Matrix

5 Case Study and Model Evaluation

Now that the artifact, the decision model for the selection of fleet-based prognostic methods, has been constructed, it needs to be justified and evaluated. Especially guideline three of the used design science framework of HEVNER ET AL. must be followed: "The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods" (2004, p. 83). In the first Section of this Chapter the methodology that is used to adhere to this guideline is presented. The second Section introduces the reader to the case study context and the decision model is used to select a best-fit method for the presented case. In the last Section the best-fit model is applied and compared with a worst-fit model. The results are compared in regard to the identified decision-criteria of the previous Chapter.

5.1 Methodology

After the proposition of the decision model, it is tested. As the proposed decision model is generic, it can be used in any prognostics context and on any dataset that contains fleet data. It was deemed important that the dataset is not biased towards a method as this would invalidate the rigor of the research (guideline five of the used research design, cf. Tab. 2). Using a dataset that is already applied within the publications that were incorporated in the decision model might constitute bias towards the implemented method. Optimally the dataset is also seldomly used, as successful implementation would indicate generalizability and applicability of the model on systems that are not yet explored. For the case study, the openly accessible Backblaze dataset (Backblaze 2019) was chosen, which contains PHM-relevant degradation data of hard disk drives (HDD). The dataset represents the largest public dataset for HDDs (Shen et al. 2018, p. 7) and has not been identified in the SLR.

To choose the best- and worst-fit alternative for the Backblaze dataset, the preference of the criteria must be defined first. Data-related criteria weighting is derived by looking at the characteristics of the dataset, while algorithm-related criteria are determined by the requirements of the practitioner. In this case study, the former criterion group is weighted by reasoning through data analysis and the latter by reasoning through the lens of a fictious data storage provider.

To enable method implementation, an appropriate framework for prognostics must be found next. The literature that has been gathered in the previous Chapters is used and reviews and meta-analyses are scanned for frameworks. A comprehensive list of frameworks is depicted in Fig. 20 of Appendix E. Note that frameworks often used diagnostics as a preliminary step to prognostics (Atamuradov et al. 2017, p. 2; Elattar et al. 2016, pp.

132, 137; Lee et al. 2017, p. 12; Pecht and Kumar 2008, p. 6; Voisin et al. 2010, p. 181) and decision support and further steps as succeeding processes (Atamuradov et al. 2017, p. 2; Elattar et al. 2016, p. 132; ISO13381-1 2015; Voisin et al. 2010, p. 181). These steps were omitted, because diagnostics and decision support (on basis of the prognosis) are not in the scope of this research. From these frameworks a simplified version was adapted that comprises the steps 1) data acquisition, 2) data preprocessing, 3) feature extraction and 4) prognostics and is depicted in Fig. 14.

The framework was implemented in the programming language Python and R. Python is a multi-paradigmatic programming language that offers many packages which facilitate data scientific work. Predominantly, the frameworks Pandas, NumPy, Lifelines and Scikit-learn were used. R is a programming language for statistical computing and graphics that also offers many packages. Mostly, the packages Reticulate, Random-ForestSRC, ggplot2 and dplyr were used.

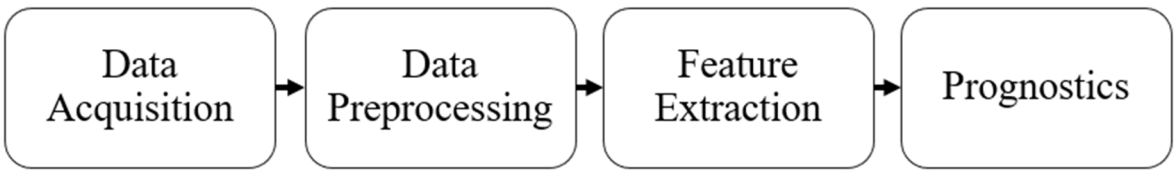
First, the scope of the case study is defined in the next Section (5.2) and the decision model is used for method selection. Then, the implementation of the best- and worst-fit methods is described in Section 5.3.

5.2 Method Selection

Before a method can be selected, a weighting vector must be defined. The weighting vector which comprises a preference score (from 0 for no preference to 10 strong preference) for each data- and algorithm related criterion is fed into the TOPSIS-based decision model and all alternatives are returned with their score that represents the fit to the best possible alternative.

5.2.1 Data Analysis and Weighting of the Data-Related Criteria

Before the dataset can be explained, some knowledge about HDDs must be established first. An HDD is a very complex system that is made up of electrical, magnetic and mechanical components, such as the head, disk and voice coil motor that can each cause a failure (Wang et al. 2011, p. 1) and thus data-driven approaches show a lot of potential. Nowadays, most HDDs are equipped with an embedded failure detection system



Own depiction.

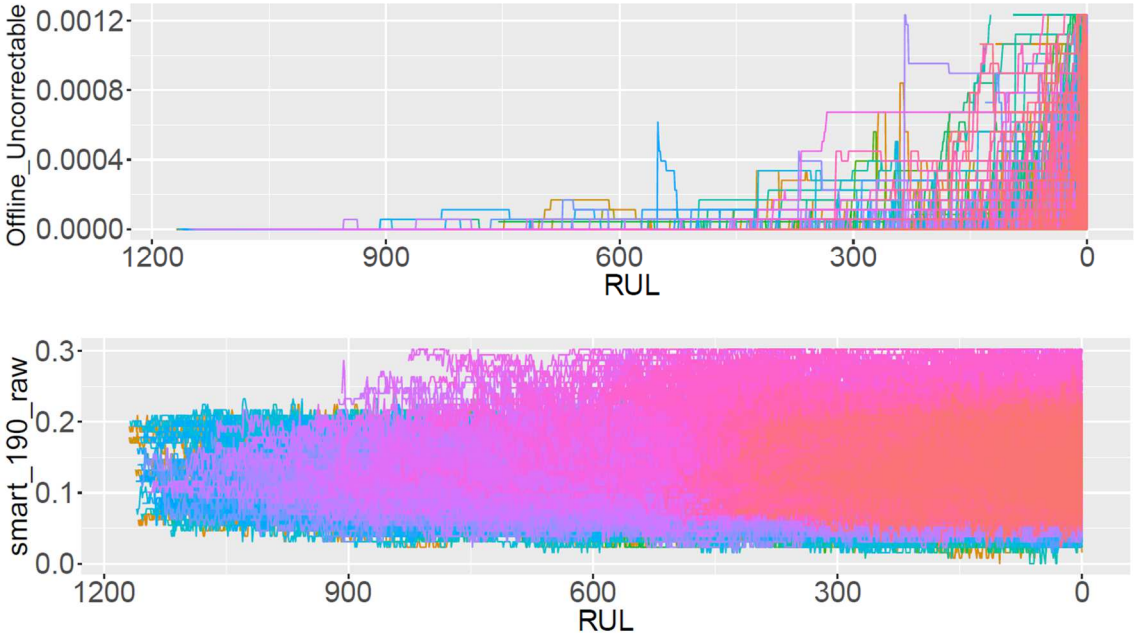
Fig. 14 Simplified Prognostics Framework

named “Self-Monitoring, Analysis and Reporting Technology” (SMART) that records many failure-related measurements (Lima et al. 2018, p. 1), such as temperature, days of service or allocation errors.

SMART stats are also recorded in the Backblaze dataset. As of 2019, Backblaze records daily snapshots of 62 SMART stats, each added to the dataset as a raw and normalized value (Backblaze 2019). It is important to note that “different drives may report different stats based on their model and/or manufacturer” (Backblaze 2019). Each row of a snapshot also contains the date and serial numbers of the HDDs; a serial number / HDD has only one snapshot per day. Further data are the model and capacity in bytes. Furthermore, each snapshot contains a flag that signifies whether the HDD is failed or not. After an HDD is flagged as failed, it is not contained in any of the succeeding snapshots. All in all, a snapshot contains 129 features.

There are some SMART values that are proven to correlate with drive failure. MASHHADI ET AL. calculated the feature importance and identified SMART 9 as the most important variable across all models of HDDs (Mashhadi et al. 2018, p. 1113). This value contains the drive life in hours (Backblaze 2019). Further, SMART 5 contains the reallocated sector count which remaps memory areas, whenever a read, write or verification error occurs (Acronis 2019a). Multiple publications rate the value as an imminent failure indicator (Backblaze 2014, 2016; Pinheiro et al. 2007). An indicator for physical degradation is SMART 10 which records the retry count of the spindle motor until fully operational speed is attained (Acronis 2019b). SMART 184, the end-to-end error was implemented by Hewlett Packard and records errors by comparing a 512 byte parity block that is added in the data path between host and hard drive (Hewlett Packard 2007, p. 2). Further correlation between annual failure rate and SMART value was identified for SMART 187, reported uncorrectable errors (Backblaze 2014), SMART 188, command timeouts (Backblaze 2016), SMART 196, reallocation event counts (Acronis 2019c; Pinheiro et al. 2007, p. 7), SMART 197, current pending sector count and SMART 198, offline uncorrectables (Acronis 2019d; Backblaze 2016; Fujitsu 2019; Pinheiro et al. 2007).

Fig. 15 shows two representative examples of SMART measurements: one is critical (offline uncorrectable, top) and one is non-critical (SMART 190, temperature difference from 100°C, bottom). Each colored line is the measurement of the respective SMART value over the life span of one HDD. On the x-axis is the inverse RUL, i.e. measurements on the right were recorded close to failure and vice versa. The y-axis contains the normalized value of the measurement. The number of offline uncorrectable values increases dramatically, while the temperature difference has an increased swing at the end of the life.

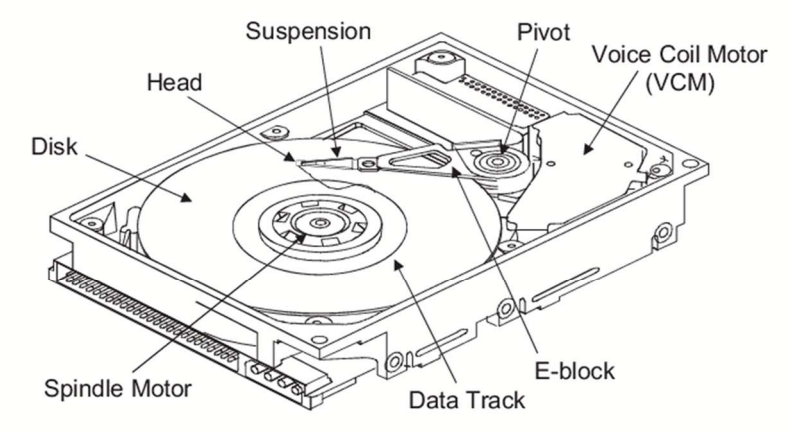


Own depiction.

Fig. 15 Two Exemplary SMART Measurements

Now, that the basics of the dataset were introduced, the data must be closely examined to derive proper weightings for the decision model.

Fleet Type (FT). HDDs fulfill the properties of homogeneous fleets which were defined as follows: The features are similar, however the usage or working conditions are different. Fig. 16 shows that HDDs are composed of eight main components. While some features, such as the disk size, can vary, all HDDs comprise these eight units and that is the reason why the fleet cannot be rated as heterogeneous. Furthermore, the fleet cannot be rated as identical, because the working conditions, SMART stats (Backblaze 2019),



(Felix et al. 2008, p. 755)

Fig. 16 HDD Architecture

failure rates and degradation patterns are proven to vary between models and manufacturers (Shen et al. 2018, p. 7).

Fleet Feature Type (FFT). Because it was already identified how sub-fleets can be distinguished, fleet features must be found that allow distinction of model and manufacturer. Luckily, Backblaze added the field 'model' as a categorical feature to the dataset. The model also makes is possible to group the HDDs by manufacturer. There are no numeric or semantic features that represent models or manufacturers.

Missing Data (M). The heat matrix in Appendix F (Fig. 23) shows that some SMART values are not recorded for some models and/or manufacturers. Green spots indicate a missing data rate of 0%, while red spots indicate 100% missing data. If all of these missing values would be counted, the missing data rate would be immense, but these empty columns cannot be used to make inference of degradation, nor can they be imputed. To measure missing data, the health critical values (i.e. SMART 5, 10, etc.) which were presented above are examined. All in all, the whole dataset comprises 79 missing values, with a total sample size of 2,026,705 observations and thus the missing data rate is 0.0039%. Because this rate is very low, a preference score of one out of ten is assigned.

Noise (N). Different strategies for measuring noise are used in practice. Typically, the mean and standard deviation are used for measuring the noise level, but using these measures assumes that data is normally distributed and does not contain outliers (Leys et al. 2013, p. 764), which is not the case in the Backblaze dataset. The median absolute deviation (MAD) offers a robust alternative. The MAD, when divided by the median (MADM), can be used as a coefficient of dispersion and e.g. is suggested by the National Institute of Standards and Technology (NIST 2019). The MAD to median can adopt any value between zero and infinity and can be calculated as follows:

$$MADM = \frac{median(|X_i - \bar{X}|)}{\bar{X}} \tag{19}$$

Here, $\bar{X}$ is the median and X_i the sample observation. A MADM of zero indicates no noise, while LEYS ET AL. suggest that a value greater than 3.5 can be used to detect outliers, so such a noise level would be abysmal (2013, p. 766).

Because the dataset contains trending, non-stationary data, the MADM was calculated with a sliding time window of ten days. For the different SMART values, the MADM ranges from above zero (e.g. reallocated sector count) to almost 1.5 (e.g. G-sense error rate) with an average MADM of 0.34 (a list per model and manufacturer can be seen in Appendix H.b). Thus, the noise level is moderate, and the preference rated with five out of ten points.

Low Sample Size (SS).

The Backblaze dataset contains 2,026,705 observations and 4812 HDDs, so the sample size is very big. Algorithms that work well on low sample data are not necessarily preferred and thus the preference is rated with zero out of ten.

5.2.2 Business Context and Weighting of the Algorithm-Related Criteria

Now that the data-related criteria are weighted, it is necessary to assign preferences to the algorithm-related criteria. In this case study it is assumed, that the measured HDDs belong to a fictitious data storage provider that wants to predict HDD failure to minimize costs.

Output type (O). The output type could help the DM to cope with uncertainty. As the decommissioning of an HDD is a decision that is made ten to eleven times a day[1] and changing HDDs is a low-investment decision, that costs ~13 cents per day the drive was removed too early[2], uncertainty management is irrelevant, and any output is acceptable.

Accuracy (A). Data loss is not a problem and Backblaze guarantees 99.999999999% reliability through data sharding (Backblaze 2018). However, for cost optimization, the accuracy is most important. Here, the ~13 cents per day can be very cost intensive if an algorithm performs bad all the time. Additionally, the rebuild time of data, if one shard is lost, takes 6.5 days (Backblaze 2018). The drive exchange could be optimized, if failure is anticipated a few days in advance. Moreover, accuracy is the bread and butter of a prognostics algorithm and should be rated highly for almost every scenario. In the case study, accuracy is preferred with ten out of ten points.

Robustness (R). The algorithm should be robust, so that small deviations in measurements or outliers do not lead to completely skewed predictions. Robustness is preferred with five out of ten points.

Time complexity (T). As multiple drives fail each day and some models have a very low sample size, incorporating new failures could improve prognostics. Also, it should be time-efficient to train on data of some past years, because a drive has an average lifetime of 753.91 days. A low time complexity is preferred with five out of ten.

Space complexity (S). Ironically, space should not be a problem for a data storage provider. Ultimately, how much space is provided is a cost-benefit decision. For the sake of this case study, space complexity is disregarded with a preference of zero out of ten.

---

[1] The average number of disk failures per day.

[2] The most used HDD model, the Seagate BarraCuda 4 TB costs ~100€ for private customers. The price divided by the mean lifespan of 753.91 days makes the cost of removing one HDD too early 13.26c/d.

Explainability (E). As replacement is a low-cost decision, it must not be thoroughly analyzed why a drive is failing or which combination of SMART measurements led to a certain decision. Also, failure mechanisms are well studied. WANG ET AL. categorized failures by mode, cause and mechanism (2011) and HUANG ET AL. constructed a reliable classifier for failure modes (2015). Explainability is not preferred (zero out of ten).

Parameter handling (P). Having no or few parameters is always preferred as this reduces the required data scientific knowledge to tune them and minimizes time complexity in case tuning can be automated. However, it is assumed, that the fictious data storage provider is assisted by a master's student with proficient data science knowledge and that tuning time (do not confuse with training time![3]) is abundant. Parameter handling is preferred with a score of three out of ten.

5.2.3 Selection

In the following the TOPSIS algorithm, that is described in Section 2.2.3, is performed.

Step 1. The final weighting vector is $w = (1, 5, 0, 10, 5, 5, 0, 0, 3)$, the normalized vector with sum one is $w_n = (.0345,\ .1724, .0, .3448, .1724, .1724, .0, .0, .1034)$. This vector must now, together with the alternative-criteria matrix A (Tab. 4), be inserted into the algorithm. A Python implementation is found in Appendix D; all intermediate calculations can be found in Appendix G. Before the matrix can be used, it must be filtered by the elimination criteria (FT = Ho, FFT = Cat and O = I ∨ D ∨ Pt), resulting in ten alternatives that can be seen in Equation (20).

$$A = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{matrix} M & N & SS & A & R & T & S & E & P \end{matrix} \\ \begin{bmatrix} 1 & 1 & 2 & 1 & 1 & 5 & 5 & 5 & 5 \\ 3 & 1 & 3 & 1 & 1 & 4 & 4 & 4 & 5 \\ 3 & 1 & 3 & 1 & 1 & 4 & 5 & 5 & 5 \\ 4 & 1 & 1 & 2 & 2 & 4 & 4 & 3 & 5 \\ 4 & 2 & 5 & 3 & 4 & 2 & 3 & 3 & 3 \\ 3 & 1 & 3 & 2 & 1 & 3 & 3 & 4 & 3 \\ 5 & 5 & 4 & 3 & 3 & 3 & 2 & 3 & 4 \\ 1 & 3 & 1 & 4 & 3 & 3 & 2 & 1 & 1 \\ 5 & 5 & 4 & 3 & 3 & 2 & 2 & 3 & 4 \\ 4 & 4 & 2 & 4 & 4 & 3 & 1 & 4 & 3 \end{bmatrix} \end{array} \tag{20}$$

Step 2. The matrix A is then normalized by the Euclidean length of column-vector expressed by the formula $r_{ij} = x_{ij} \Big/ \sqrt{\sum_{i=1}^{n} x_{ij}^2}$. For the M column the vector length is $\sqrt{1^2 + 3^2 + 3^2 + 4^2 + 4^2 + 3^2 + 5^2 + 1^2 + 5^2 + 4^2} \approx 11.2694$ and thus the first normalized value for M and alternative (Jordan et al. 2018) is $\frac{1}{11.2694} \approx 0.0887$. The normalized matrix can be seen in Equation (34) in Appendix G.b.

[3] Please note that parameter tuning must not be executed on each retraining of the model but is a onetime effort at conception of the model.

Step 3. In the next step, the normalized matrix is multiplied by the vector w_n which can be expressed with the formula $v_{ij} = w_j * r_{ij}$. For the first cell, the equation is $0.0887 * 0.0345 \approx 0.0031$. The complete normalized weighted matrix can be seen in Equation (21).

$$W = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{array}{cccccccccc} M & N & SS & A & R & T & S & E & P \end{array} \\ \begin{bmatrix} .00 & .02 & .0 & .04 & .02 & .08 & .0 & .0 & .04 \\ .01 & .02 & .0 & .04 & .02 & .06 & .0 & .0 & .04 \\ .01 & .02 & .0 & .04 & .02 & .06 & .0 & .0 & .04 \\ .01 & .02 & .0 & .08 & .04 & .06 & .0 & .0 & .04 \\ .01 & .04 & .0 & .12 & .08 & .03 & .0 & .0 & .02 \\ .01 & .02 & .0 & .08 & .02 & .05 & .0 & .0 & .02 \\ .02 & .09 & .0 & .12 & .06 & .05 & .0 & .0 & .03 \\ .00 & .05 & .0 & .16 & .06 & .05 & .0 & .0 & .01 \\ .01 & .09 & .0 & .12 & .06 & .03 & .0 & .0 & .03 \\ .01 & .07 & .0 & .16 & .08 & .05 & .0 & .0 & .02 \end{bmatrix} \end{array} \quad (21)$$

Step 4. Now the highest values and lowest values of each criterion are used to compose the PIS and NIS, respectively. For instance, 0.2 is the highest, 0.00 the lowest value for criterion M. The PIS vector is $\underline{v_p} = (.02, .09, .0, .16, .08, .08, .0, .0, .04)$ and the NIS vector $\underline{v_n} = (.00, .02, .0, .04, .02, .03, .0, .0, .01)$.

Step 5. Now, the Euclidean distance from each alternative to the PIS and NIS are calculated. For instance, the distance from the first alternative to the PIS is $d_{1b} = \sqrt{(0 - 0.02)^2 + (0.2 - 0.9)^2 + \cdots + (0.4 - 0.4)^2} \approx 0.16$. The full distance matrix can be found in Equation (36) of Appendix 0.

Step 6. Now the similarity of each alternative to the PIS is calculated by $s_{iw} = d_{iw}/(d_{iw} + d_{ib})$. The final vector can be seen in Equation (22).

$$S = \begin{array}{c} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Wang et al. 2018)} \\ \text{(Jordan et al. 2018)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \end{array} \begin{array}{c} s_b \\ \begin{bmatrix} .79 \\ .68 \\ .68 \\ .65 \\ .55 \\ .35 \\ .27 \\ .26 \\ .22 \\ .22 \end{bmatrix} \end{array} \quad (22)$$

The works (Voronov et al. 2018) and (Frisk et al. 2014) achieve the highest similarity value with ~79%, while the methods (Wolak 2018), (Wang et al. 2012) and (Jardine et al. 2001) perform worst. To test whether the choices of the decision model are correct, the best-fit publication of FRISK ET AL. (2014) will be compared to the worst-fit of JARDINE ET AL. (2001). The former implemented a random forest for survival (from now on addressed by RSF), the latter a proportional hazard model (addressed as PrHM). Both methods are implemented by the prognostics framework that was presented above (Fig. 14).

5.3 The Experiment

Now that the best- and worst fit algorithms have been identified, they are implemented and compared. Up until the third step, feature extraction, the implementation of both methods is identical for both methods, which guarantees a higher level of comparability.

5.3.1 Data Acquisition

The dataset was extracted with a total of 62 SMART stats as raw and normalized values. These, plus date, serial number, model, capacity and failure resulted in 129 features. Data was collected from January 1, 2016 to March 31, 2019 and only HDDs with a failure in this timespan were selected. The data thus does not contain censored data; however, its initial conditions are unknown if the drive was employed before January 1, 2016. Then, only the power on hours (SMART 9) provide insight into the past of the HDD.

In total, the data comprised 2,026,705 observations and 4812 HDDs. The descriptive statistics of the critical values in the Backblaze dataset can be seen in Tab. 5. As measurements of failed drives are also recorded, they are compared to statistics of healthy snapshots. It is shown, that failed HDDs have a standard deviation that is at least one magnitude higher than the mean, while the contrary is true for running HDDs. This corresponds to the low fluctuations at the beginning and high fluctuations at the end that can also be seen graphically (e.g. Fig. 15). Tab. 6 additionally depicts the descriptive statistics of the days in service. Originally, the “RAW value of SMART 9 is the number of hours a drive has been in service up to that point” (Backblaze 2019), but in the following, the value is divided by 24 to determine the days of service.

The dataset comprises the manufacturers Seagate (4071 HDDs, 23 models), Hitachi (433 HDDs, 11 models), Western Digital (186 HDDs, 22 models), Toshiba (99 HDDs, six models) and Samsung (10 HDDs, one model).

5.3.2 Preprocessing

The first step of preprocessing is the fleet identification (stage 1). Which identification method is suitable is a difficult problem and domain knowledge is generally required. As stated above, it is already proven by multiple works, that failure rates and degradation are highly correlated with models and manufacturers (Pinheiro et al. 2007, p. 4; Shen et al. 2018, p. 7). Additionally, most drives do not report values for all SMART stats and some

	Mean		Standard deviation		Minimum		Maximum	
Measure	Failed	Running	Failed	Running	Failed	Running	Failed	Running
SMART 5	8.03e+02	6.17e+01	4.29e+03	1.18e+03	0.00e+00	0.00e+00	6.17e+04	6.53e+04
SMART 10	8.62e+02	1.12e+01	2.08e+04	1.63e+03	0.00e+00	0.00e+00	1.11e+06	1.11e+06
SMART 184	2.14e-01	2.11e-02	2.60e+00	5.36e-01	0.00e+00	0.00e+00	0.76e+02	6.80e+01
SMART 187	4.25e+01	7.39e-01	1.47e+03	1.12e+02	0.00e+00	0.00e+00	6.55e+04	6.55e+04
SMART 188	4.60e+08	1.35e+08	8.21e+09	2.96e+10	0.00e+00	0.00e+00	4.51e+11	8.93e+12
SMART 196	8.24e+01	2.82e+01	4.61e+02	2.57e+02	0.00e+00	0.00e+00	9.11e+03	9.03e+03
SMART 197	1.74e+02	1.71e+00	2.48e+03	8.49e+01	0.00e+00	0.00e+00	1.43e+05	5.58e+04
SMART 198	1.64e+02	1.66e+00	2.45e+03	8.49e+01	0.00e+00	0.00e+00	1.43e+05	5.58e+04

Own depiction.

Tab. 5 Descriptive Statistics of Backblaze Dataset

	Count	Mean	Std	Min	25%	Median	75%	Max
SMART 9	4812	753.91	502.93	0	348.04	696.67	1131.36	3769.88

Own depiction.

Tab. 6 Descriptive Statistics about Days in Service (SMART 9)

stats “can vary in meaning based on the drive manufacturer and the drive model” (Backblaze 2019). This is also shown in the heat matrix in Appendix F (Fig. 23) that was explained in Section 5.2.1. The distinction between fleets was possible through the categorical feature ‘model’ that is included in the Backblaze dataset. Due to the choice of having prognostic methods with categorical fleet identification components, the two publications of FRISK ET AL. (2014) and JARDINE ET AL. (2001) were returned earlier. Both use similar approaches to identify fleets. The former differentiated five fleets of vehicles by their selling market (2014, p. 2), the latter analyzed two fleets from the same mine site, but with different numbers of wheel motors (2001, p. 287). FRISK ET AL. implemented a RF, JARDINE ET AL. a PrHM for each fleet. This approach is easily adaptable to the Backblaze dataset. Because it is not known if fleet identifaction is optimal by model or manufacturer, model fleets (63 fleets) and manufacturer fleets (five fleets) are constructed. The identification of the models could be done by directly looking at the field ‘model’ (e.g. ST4000DM000). Because there was no field for the manufacturer, it was derived by the first two letters of the model (e.g. ‘Sa’ for Samsung, ‘ST’ for Seagate). All following steps were conducted per each fleet.

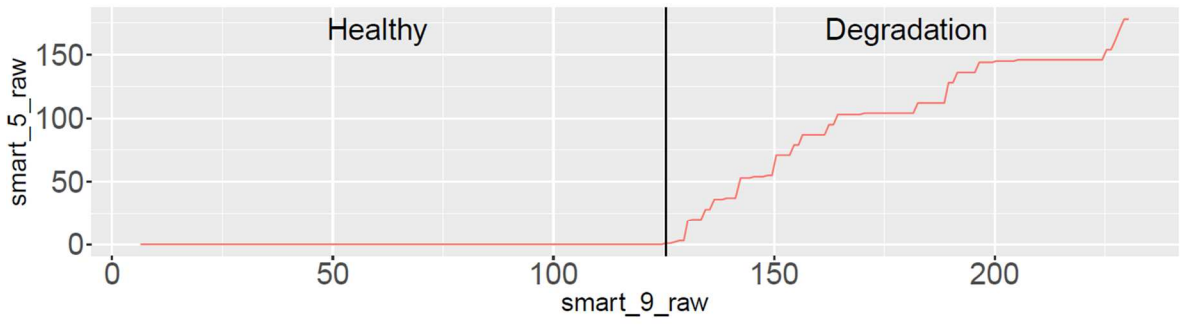
First, the RUL was added as a label by counting upwards from the entry where the feature ‘failure’ signifies ‘1’ (for failure). The last entry has a RUL of zero, the penultimate a RUL of one, etc.

Afterwards, data cleaning was conducted as proposed by PECHT AND KUMAR (2008, p. 6) and ATAMURADOV ET AL. (2017, p. 5). The 62 normalized SMART values were omitted, because the analyzed data is only a subset of the whole data and it is unknown how the manufacturers normalized the data. Spurious data, which comprise “negative counts or

data values that are clearly impossible" (Pinheiro et al. 2007, p. 3) were dropped. For instance, there were drives with reported temperatures hotter than the surface of the sun or with negative power cycles. The filtering was done manually and only few entries were dropped. More realistic, but still extreme outliers were left in the data, because they could potentially indicate failure. No statistical outlier detection was applied. Some HDDs were only employed for one day before failure; these were dropped too. Moreover, columns were dropped, if they contained only constant values and if more than 95% of data was missing. This high threshold verifies, that only columns were dropped, where one or multiple models or even the whole manufacturer did not measure the SMART stat at all. After the columns were dropped, a missing data rate of 0.0039% which was reported earlier could be achieved.

Like many PHM problems, the life of an HDD can be separated into a healthy and degradation state. In the healthy state, none of the critical values which were explained in Section 5.2.1 and shown in Tab. 5 have a non-zero value. As soon as one critical value increases, degradation is initiated (an example can be seen in Fig. 17). As proposed by AL-DAHIDI ET AL., only trajectories after onset of degradation are further analyzed (2016, p. 119). Unfortunately, this led to 33% of HDDs being dropped completely, as they did not record any non-zero values for critical SMART values at all. This is a known problem for HDD prognostics and in line with BACKBLAZE, which report that 23.3% of failed drives show no warning from six critical SMART stats (2016), with PINHEIRO ET AL. which even reported 36% (2007, p. 10) and with MASHHADI ET AL. which removed over one third of empty rows (2018, p. 1109). Because a lot of data was dropped, the 63 model fleets shrank to 21, and the five manufacturer fleets to four (Samsung was omitted).

Ultimately, the SMART values were normalized as proposed by FRISK ET AL. (2014, p. 2) and PECHT AND KUMAR (Pecht and Kumar 2008, p. 6). Contrary to the works on the C-MAPSS turbofan dataset, the models are not collected "under variable operational conditions", so a multi-regime normalization was not conducted as proposed by WANG (2010). Instead feature data was min-max normalized per fleet with the Equation (23).



Own depiction.

Fig. 17 Degradation Onset of HDD ZCH072P4

$$\frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)} \tag{23}$$

x_{ij} is the ith feature value of SMART stat j; $\min(x_j)$ and $\max(x_j)$ are the minimum and maximum values of the complete SMART stat.

5.3.3 Feature Extraction

Typically, RFs and PrHMs cannot address sequential data and thus feature extraction was conducted as proposed by ZHAO ET AL. (2017, p. 283). Feature extraction allows enrichment of data by adding new, relevant features. It can be distinguished between four types of feature extraction methods in PHM: Frequency-based, residual-based, time-slice statistics and time statistics (Lee et al. 2017, p. 14). As the time series is known, the third type was applied to the dataset with a rolling time window of nine past days. For each observation, the Equations (24) to (31) were applied.

Root Mean Square (RMS)
$$\mathrm{rms}_t = \sqrt{\frac{1}{r+1}\sum_{i=0}^{r} x_{t-i}^2} \tag{24}$$

Minimum
$$\min_t = \min(z) \tag{25}$$

Maximum
$$\max_t = \max(z) \tag{26}$$

Peak-to-Peak
$$ptp_t = \max_t - \min_t \tag{27}$$

Mean
$$\bar{x_t} = \frac{1}{r+1}\sum_{i=0}^{r} x_{t-i} \tag{28}$$

Variance
$$\sigma_t^2 = \frac{1}{r+1}\sum_{i=0}^{r} (x_{t-i} - x_t)^2 \tag{29}$$

Skewness
$$\mathrm{skew}_t = E\left[\left(\frac{z - \bar{x_t}}{\sigma_t}\right)^3\right] \tag{30}$$

Kurtosis
$$\mathrm{kurt}_t = E\left[\left(\frac{z - \bar{x_t}}{\sigma_t}\right)^4\right] \tag{31}$$

In these Equations, x_t is the SMART value x of time index t and r the rolling time window (i.e. 9). All calculations therefore include the focal observation, as well as nine past observations ($x_t, x_{t-1}, \dots, x_{t-r}$); this is denoted as z. This also implies, that feature engineering could not be applied to the first ten observations of each HDD.

In conclusion, “Time-domain based feature extraction techniques (e.g. root mean square, kurtosis etc.) are used to analyze the global characteristics of data” (Atamuradov et al.

2017, p. 4), and thus counter the sequential data problem of RFs and PrHMs. However, the data was extended to 563 features, a number that is high-dimensional. When presented with high dimensionality, methods usually choke because a) the training time increases exponentially and b) models overfit (Gheyas and Smith 2010, p. 5). To counter these drawbacks, feature selection must be applied to reduce dimensionality.

Up to this point, the preparation of the data was identical for each of the two selected methods. However, the publications by FRISK ET AL. (2014) and JARDINE ET AL. (2001) have different approaches for feature selection.

ALTERNATIVE RSF BY FRISK ET AL. (2014). The best-fit method suggests to implement feature selection proposed by ISHWARAN ET AL. (2007; 2008). The basic principle is, that RFs are implemented with bootstrap sampling. An out-of-bag (OOB) sample, i.e. a sample that does not contain the feature x, is dropped down an in-bag RF, i.e. a RF that has been trained with x. When a split node with x is encountered, a random child is chosen (Ishwaran et al. 2008, p. 849). The prediction error of this ensemble is compared with the original RF prediction (Frisk et al. 2014, p. 7). The difference of the errors is the feature importance of x. This method is repeated for each feature. Alike the publication of FRISK ET AL. (2014), the 30 most important features for each fleet were chosen.

ALTERNATIVE PRHMBY JARDINE ET AL. (2001). Using the same feature selection for the PrHM by JARDINE ET AL. (2001) is problematic. The hazard function for PrHM is estimated by partial likelihood, which is optimized by the Newton-Raphson approach, that does not converge when there is a high collinearity between features (González et al. 2008, p. 513). To prevent non-convergence, features with a Pearson correlation coefficient of $\rho > 0.9$ were removed. Congruently to the method of JARDINE ET AL., "Covariate significance is tested by the Wald statistic, the square of the standardize estimate of the parameter which follows a chi square distribution with one degree of freedom" (2001, p. 297).

5.3.4 Prognostics

The final dataset comprised more than 35,000 observations and 3,700 serials in the 21 model fleets and more than 1,780,000 observations and 4,278 serials in the 4 manufacturer fleets. The data was split into 80% training and 20% test data (holdout principle).

Alternative RSF by FRISK ET AL. (2014). FRISK ET AL. implemented a RSF that extends the RFs famously proposed by LEO BREIMAN (2001). RSFs are a survival analysis extension of RFs that can be used as classifiers or regressors. Vast empirical evidence indicates that RFs are highly accurate models that can compete against state-of-the-art methods

(Ishwaran et al. 2008, pp. 841–842). The basic idea is to map the feature space X into the outcome space Y non-linearly. Within RFs, this is done by partitioning X into binary subtrees, where at the terminal node, the outcome is s.t. a cost function that must be optimized (Voronov et al. 2018, p. 626). For larger problems, two random elements are introduced, namely bootstrap sampling (bagging) and random split feature selection. The former samples a subset of the data and uses it to grow a tree, while the latter randomly selects a subset of splitting features in each node (Ishwaran et al. 2008, p. 841).

While RSF can also work on single right-censored snapshots, they are inferior if the system to-predict is tracked over its lifetime (Voronov et al. 2018, p. 636). Because the Backblaze dataset does contain information about major parts of the lifecycle of HDDs, a regressive RSF is used. The only difference is, that instead of an RSF predicting a survival function with the log-rank test (Voronov et al. 2018, p. 627), a regressive RSF predicts a continuous value for its given input with the mean squared error (MSE) as the cost function (Kogalur and Ishwaran 2016). FRISK ET AL. uses the R package "Random Forests for Survival, Regression and Classification" (2014, p. 4) and is also employed in this research (cf. Ishwaran and Kogalur 2019).

FRISK ET AL. state that three important parameters must be optimized: the number of trees, number of splitting variables in each node and terminal node size (2014, p. 4; Voronov et al. 2018, p. 630). Luckily the R package includes a function that allows for parameter tuning of the three variables (Ishwaran and Kogalur 2019, pp. 89–91). Beyond the feature importance, an optimal triple of parameters was determined per each fleet. The tree size was held constant at 200 trees, because the OOB error could not be improved beyond this number. The number of splitting variables was tuned by iteratively growing trees. The initial value was 15 per node and the value increased by 1.25 each step. If the error did not improve by at least 0.001 in three succeeding iterations or when 25 iterations were reached, the optimization was stopped and the parameter value with the lowest error was chosen. The node size was tuned by trying all integer values from one to ten and integer values from ten to 100 with a step factor of five. The optimal parameter triple for each fleet can be seen in Appendix H.a.

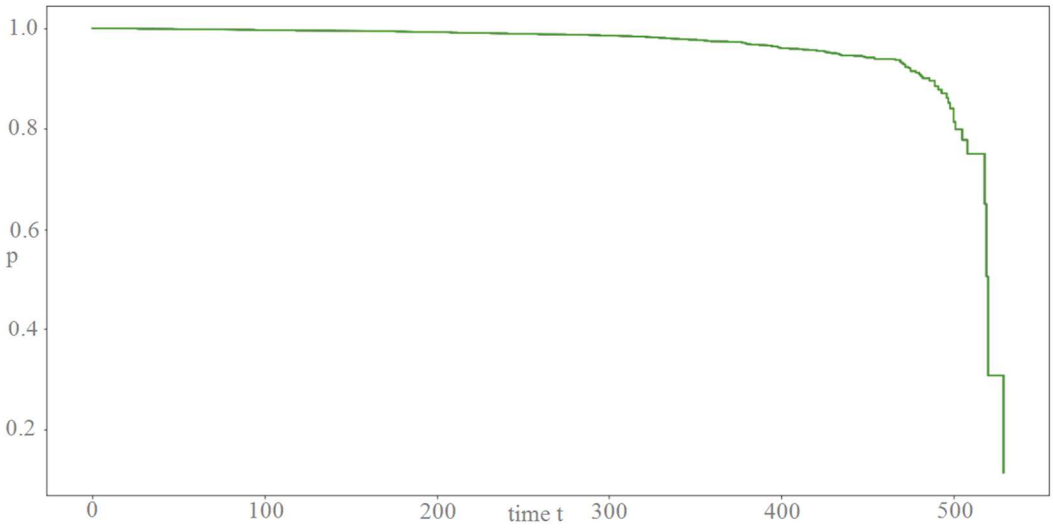
Alternative PrHM by JARDINE ET AL. (2001). The PrHM, also named Cox regression, was proposed by DAVID R. COX (1972). JARDINE ET AL. adapted the approach for fleets (2001). The PrHM function consists of a baseline function $\lambda_0(t)$, and a feature term $e^{\gamma_1 Z_1(t)+\gamma_2 Z_2(t)+\cdots+\gamma_m Z_m(t)}$ (Jardine et al. 2001, p. 288). The baseline function and covariate terms are estimated by MLE which is applied to the fleet data. However, because the PrHM works with right-censored data (where many observations are not failed), the baseline hazard function could vanish and therefor no failure is predicted if traditional MLE

is used. To solve this issue, COX proposed partial MLE (1972, pp. 190–192). Partial MLE can be maximized using the iterative Newton-Raphson method, that converges towards an optimal solution (González et al. 2008, p. 514).

When the Newton-Raphson method estimates the parameters, the different features are weighted within the covariate term (cf. above). In contrast to simple regression methods, the covariates have a multiplicative instead of an additive effect on the function (Sikorska et al. 2011, p. 1825). The exponent of the exponential covariate function can be split into m parts, where m denotes the index of the feature. One covariate monomial can be expressed as $\gamma_m Z_m(t)$. $Z_m(t)$ is the value of feature m (i.e. the SMART value) at time t and γ_m is the coefficient. The bigger the absolute value of the coefficient, the bigger the effect of the feature on the hazard function.

JARDINE ET AL. implemented the PrHM with a software called EXAKT, that was developed by the university of Toronto (2001, p. 288). Unfortunately, EXAKT is a commercialized software (Jardine 2019). Instead of EXAKT, the lifelines package of Python is used (Davidson-Pilon 2019a). The PrHM allows only few parameters to be tuned, that empirically have low impact on the prediction outcome. A penalizer can be added that tackles overfitting. Different values have been tried and the value 0.5 was chosen. Additionally, the step size for the partial MLE was chosen to be 0.01. The parameters are kept fixed for each fleet.

After the PrHM has been fitted to the training data, it can be used to predict the hazard function (Fig. 18). In the depicted example, the x-axis contains the predicted lifetime of the HDD and the y-axis plots the probability that the HDD survives up to that point. Like



Own depiction.

Fig. 18 Survival Function for a Random Sample

the critical SMART values that dramatically increase at the end of life of the unit, the survival function increases proportionately. To be sure that the HDD life is not overestimated, the chance of survival is chosen to be 90%. The life time of the HDD can be found by plotting a horizontal line through the 90%tile of the y-axis. The x-value of the intersection of the line with the function is the predicted life time. To calculate the RUL, this value then must be subtracted by the days in service (SMART 9).

5.4 Results

In terms of accuracy, both models performed moderate on most fleets. As mentioned earlier, HDD failures are hard to predict and topic of on-going research. Nevertheless, it can be seen, that both algorithms correctly incorporate the SMART values that were deemed critical by expert literature. Both algorithms confirm the findings of past studies, and this can be seen by the feature selection (RSF) and covariate coefficients (PrHM), that all underline the importance of the critical values.

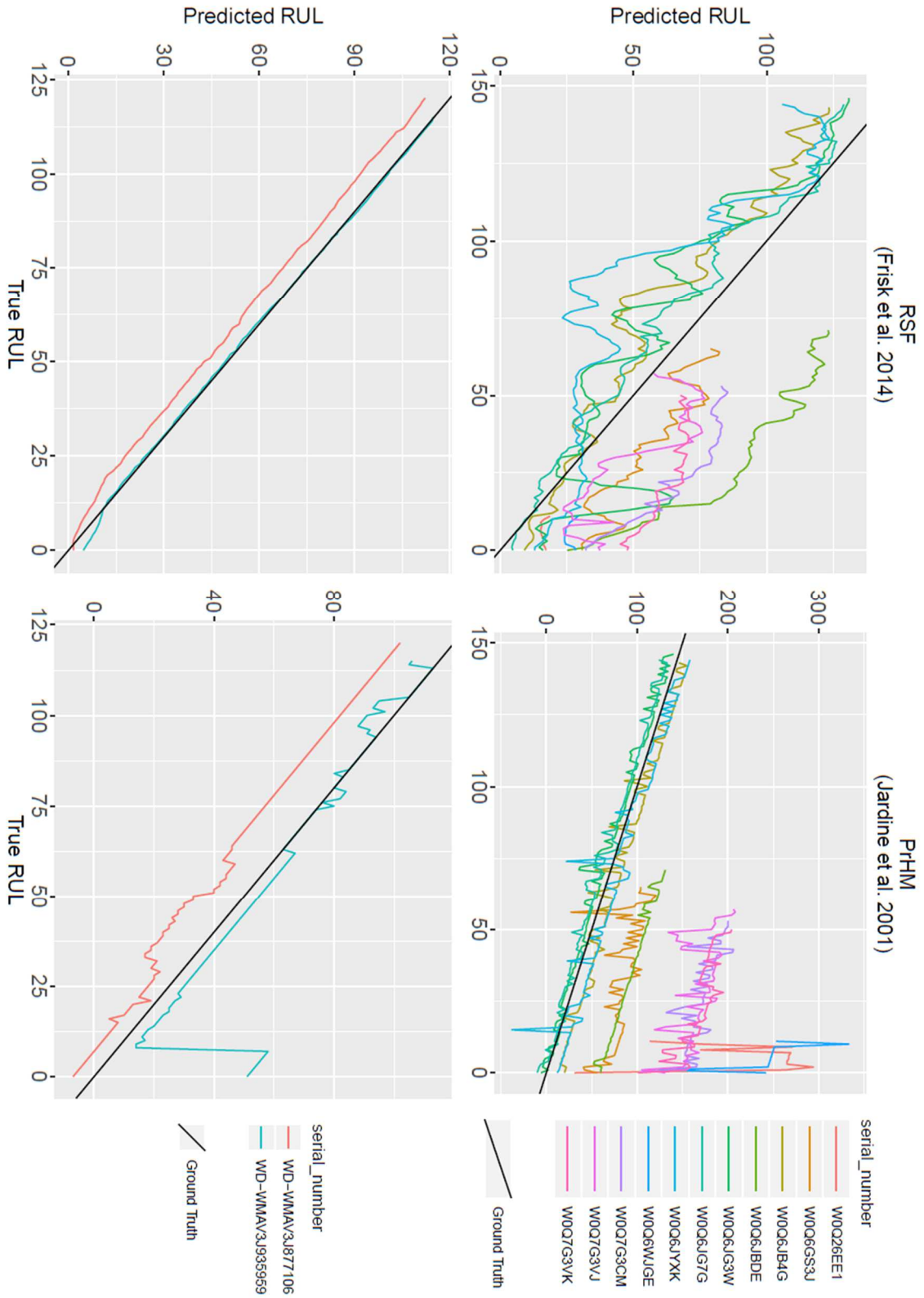
In Fig. 19, predictions for two representative model fleets, “ST320LT007” and “WD800AAJS” are presented. On the y-axis the true RUL is depicted, while the x-axis contains the predicted RUL at each time step. In the best case, all colored lines, each representing one HDD, should match the black-lined ground truth. Both methods recognize the downwards trend of the HDDs, but the predictions for the first HDD model have a high error. While the RSF underestimates the RUL at the beginning and overestimates at the end of “ST320LT007”, the PrHM recognizes the correct slope of the degradation, but assumes a wrong intercept for some of the HDDs. The HDDs of model “WD800AAJS” is forecast with a high accuracy for both models.

Lastly, the goal of the research was to see, how the different algorithms perform against each other regarding the decision criteria. The performance of each criterion is compared in the following. At the beginning of each criterion, the assigned scores from the decision model (Chapter 4) are repeated.

5.4.1 Data-Related Evaluation

Missing Data (M). *RSF = 4; PrHM = 3*

As the missing data rate was especially low in this dataset (0.0039%), it was of no interest for the selection of the optimal method. Both, the best- and worst-fit method have tolerance for medium (3) to moderately high levels (4) of missing data and both could cope with the missing values. There was no indication that missing values hindered the prediction accuracy of the algorithms. For future research it would be interesting to see how



Own depiction.

Fig. 19 Predicted RUL for Models ‚ST320LT007‘ (Top) and ‚WDC WD800AAJS‘ (Bottom)

well methods perform for each criterion if they are rated with totally different scores, but as only two methods are benchmarked and are deemed to overlap in some criteria, it is difficult to come to a meaningful result for this criterion.

Noise (N). *RSF = 4; PrHM = 1*

The MADM ratio was moderate overall (0.34) and some fleets exerted a lot of noise. For all fleets whose noise level was above the mean (ST8000DM002, ST8000NM0055, ST12000NM0007, ST4000DM000, Hitachi, Seagate), the RSF performed always better in terms of accuracy (cf. Appendix H.b). Accuracy was measured by the root mean square error (RMSE), as proposed by SAXENA, CELAYA, ET AL. (2008, p. 9). On noisy fleets, the RSF performed 150.75% better than the PrHM on average. This can also be seen in Fig. 19, where the top HDD model has a MADM of 0.32. The PrHM (top right) shows a lot of jumps in the predictions whenever noise is encountered, while the RSF exerts smoother predictions.

Low Sample Size (SS). *RSF = 2; PrHM = 3*

RSF and PrHM are suited for medium (3) to moderately high (2) sample sizes. As explained earlier, the data has a high sample size of multiple million observations and even the smallest fleet contains 900 observations (model ST3160318AS). The RSF performs better for 90% of model fleets and for 75% of manufacturer fleets. It only performs worse for the models “Hitachi HDS5C4040ALE630”, “ST10000NM0086” and the manufacturer “Toshiba”. All have small sample sizes, with only 13, 5 and 32 HDDs in their training set, respectively. Surprisingly, on other low sample fleets, RSF performs better.

All in all, the results seem plausible, because both methods are rated close to each other, and the superiority of the RSF can also be attributed to a much higher general accuracy.

5.4.2 Algorithm-Related Evaluation

General Accuracy (A). *RSF = 4; PrHM = 1*

The RSF is rated much higher in terms of general accuracy and this is reflected in the results. The accuracy of the algorithm is measured by the RMSE (Saxena, Celaya, et al. 2008, p. 9). For the different model fleets, the RSF delivers improvements of 211.21% over the PrHM and 131.37% improvement within the manufacturer fleets. As seen in Fig. 19 the RSF predicts closer to the ground truth most of the time.

Robustness (R). *RSF = 4; PrHM = 1*

Robustness is measured by the sensitivity that was proposed by SAXENA, CELAYA, ET AL. (2008, p. 10) and VACHTSEVANOS ET AL. (2006, p. 394) and can be seen in Equation (32).

Δ_i^{Out} is difference between two successive outputs and Δ_i^{In} between two inputs. Output is in this case the RMSE as suggested by SAXENA, CELAYA, ET AL. (2008, p. 10) and inputs

$$SN = \frac{1}{N}\sum_{i=1}^{N} \frac{\Delta_i^{Out}}{\Delta_i^{In}} \tag{32}$$

are the different SMART values. Through this measurement, it can be seen how sensitive an algorithm reacts to small changes. Optimally, the RUL should be relatively constant for two successive measurements. If the method is robust, the sensitivity is low.

The RSF performs better in 68.42% of the model fleets and is lower for 100% of the manufacturer fleets. The average improvement of sensitivity is 2,508.11% over the PrHM. The robustness can also be seen in both examples of Fig. 19. A robust RUL prediction is linear (i.e. ground truth). The PrHM exerts a lot of small spikes in the predictions, whereas the RSF predicts more smoothly. A full list of sensitivity scores can be seen in Appendix H.d.

Time Complexity (T). *RSF = 3; PrHM = 4*

To measure time complexity, the total time between training or fitting the model until returning the final prediction was measured in milliseconds. Both models performed similar on a compute-optimized Amazon Web Server with an eight core Xeon Platinum 8000, 3.5 GH and 21 GB RAM. The RSF is faster 61.90% and 50% of the time, for model and manufacturer fleets respectively. While this contradicts the rating that was derived by literature, it must be noted that the RSF was trained on only 30 most important variables and training was even reduced by the node size parameter, that used only a subset of all variables. It can also be seen, that the time for fitting the PrHM increases exponentially, the more observations the fleet contains. The time complexity for each fleet is listed in Appendix H.e and it is a metric that should be investigated further in future research.

Space Complexity (S). *RSF = 1; PrHM = 4.*

The trained RSF and the fitted PrHM were saved as a file and their uncompressed file size was compared. The models do not include the data, but they do include the default elements that are defined by the packages RandomForestSRC and lifelines. For instance, the former includes the total of 200 random trees, but also feature importance weights and OOB errors. The latter includes the Cox regression function, but also Wald statistics and

confidence intervals of the coefficients. The total size per fleet in KB can be seen in Appendix H.f. The PrHM is always smaller than the RSF; on average it is only 10.02% the size of its counterpart and thus the given scores are plausible.

Explainability (E). *RSF = 4; PrHM = 5.*

Explainability cannot be quantified, instead it is qualitatively assessed. The RSF was moderately transparent. The feature importance scores gave insight into which variables were most important for measuring degradation in each fleet and the decision trees are easily understood. Merely the randomness of the tree structure is not transparent, but nervertheless RSFs offer one of the most explainable AI. The PrHM is more explainable. The coefficients directly represent the importance of the covariates and the survival function that can be plotted for each observation is easy to understand visually (cf. Fig. 18).

Parameter Handling (P). *RSF = 3; PrHM = 5.*

Parameter handling is also not quantifiable. While the number of parameters can indicate the complexity in some way, it is not sufficient. For instance, the RSF comprises three variables, but all have a big impact on the prediction error and tuning them took a long time. Also, the node size or number of splitting variables depend on the number of observations and features and are thus dependent on the problem at hand. Even though the PrHM contained one less parameter, the two parameters were much less influential on the error and the range of possible values is independent of the data.

Summary. All in all, the results indicate, that each decision-relevant criterion was correctly rated. Unfortunately, the two methods are rated equally for some criteria, such as missing data, low sample size, time complexity and explainability so that meaningful inductions cannot be made. Additionally, the characteristics of the dataset do not allow comparison of certain criteria, e.g. missing data. It is important to test the decision model with further datasets and case studies to validate it, but for now, the results look promising.

6 Discussion

Research Objective 1: Fleet-Based Prognostic Methods

The first objective, the exploration of the current state-the-art in fleet PHM has been completed by carrying out a SLR. Pivotal fleet-data-based approaches for fault prognostics were identified. The methods could be separated into the three steps fleet identification, fleet usage and prognostics and their aptness to specific fleet types could be analyzed.

The approach of using existing methodologies is suggested by the employed design science framework of HEVNER ET AL. (2004). For the literature review, the methodology of VOM BROCKE ET AL. (2009) offered a tool for unbiased and exhaustive research of the publications in the targeted field.

Unfortunately, the methodology had to be slightly changed, because the journal search was not feasible as a first step. PHM is a multi-disciplinary field and it is impossible to narrow down relevant journals. It would be beneficial if a meta-analysis of different journals for PHM is conducted that includes journals from the various disciplines, such as engineering, computer and materials science, reliability, artificial intelligence, physics, and economics.

Moreover, the key word section “Fleet” that can be seen in Fig. 21 of Appendix B.a is problematic, as most researchers do not use the same terminology for fleets. This was also the reason why some works could be only found via a backwards search. Research towards a PHM taxonomy is desired, but most importantly, researchers must clarify what type of fleets they use and how.

Also, while the alternatives of the proposed decision model can be generally divided into the three stages proposed in Chapter 3, the alternatives of the decision model are tightly coupled. There is potential to split up the stages of the methods (e.g. fleet identification) and recombine them to offer more flexible methods.

Lastly, NFS were not identified in fleet prognostics, even though they are widely applied for non-fleet data. The fuzzy pendant to the NN offers great potential and it is must be analyzed why they have not been adapted in fleet prognostics research yet.

Research Objective 2: The Decision Model

The second research objective was fulfilled by constructing the principal artifact of this research: the decision support model for the selection of fleet-based prognostics methods. MCDM techniques generally work with alternatives (i.e. prognostic methods) and criteria. While the former has been identified in research objective 1, the latter still had to be determined before an artifact could be constructed. Because the identified publications generally do not disclose many criteria and seldomly measure more than a couple, secondary overview literature has been gathered. The combination of the works from the structured literature review and the summary publications allowed an objective, rigorous and exhaustive research and rating of the alternatives.

The MCDM method TOPSIS was used to enable the decision model and it fulfills the conditions of the employed research design framework which requires that the knowledge base must be transformed into a tangible and actionable artifact.

Of course, criteria are heavily abstracted to facilitate a generic model. For instance, missing data is a very broad criterion. It is not stated whether the data is left-, right or interval-censored or whether the observed systems have different initial wear conditions. While the generalization of the decision model reduces the complexity, it must be further analyzed whether critical criteria are missing.

Additionally, none of the identified prognostic methods stated how well their algorithm performs in each of the different criteria. Most algorithms only reported on a few criteria at most and a lot had to be derived second-handedly from literature that deals with the different types of algorithms that were employed. This was especially difficult, because the methods often comprised multiple algorithms (e.g. k-means clustering plus multi-regime normalization plus feature engineering plus neural net ensembles). In the same way in which all publications recognized the need to test their method within a case study or an experiment, it should also be natural that the basic criteria like time complexity, accuracy, robustness, etc. are always reported. In the end, the given rating per each model was an “educated guess” that is based on substantial amount of theoretical and empirical research. Through further testing of the model through case study research, the scores can be confirmed or rejected (and subsequently adapted). The decision model thus offers the possibility to adapt it, both by adding new, but also by changing existing alternatives whenever new evidence is brought to light.

Research Objective 3: Case Study and Model Evaluation

The proposed artifact was tested with the real-life dataset by Backblaze, that contains measurements of HDDs over the span of more than three years. The ratings of a best- and worst-fit method towards the decision criteria could be confirmed in a case study and the applicability and generalizability of the decision model looks promising.

Through a review of prognostic frameworks, a four-step framework was derived to implement the two methods. The Backblaze dataset was not used in the fleet-based approaches that have been identified prior and thus the objectiveness of the research is facilitated. The case study was evaluated with measures that are scientifically reputable (e.g. Saxena, Celaya, et al. 2008).

The weights that are put into the decision model have been defined by examining real-life data and introducing a fictious, but realistic business case study. In the future, the evidence of the model evaluation could be enhanced by applying it in a real business case. Additionally, the weighting of the decision criteria could then be rated by domain experts, e.g. within a Delphi study.

Additionally, TOPSIS was used as an underlying MCDM framework, because the number of alternatives and criteria was big. TOPSIS is criticized for allowing inconsistent decision-making and AHP in combination with TOPSIS could potentially be used to counteract this (example cf. Kirubakaran and Ilangkumaran 2016).

It is a common misunderstanding that case study research cannot be used to generalize (Flyvbjerg 2006, p. 228), and this case study is central to demonstrate the generalization of the proposed decision model. Nevertheless, the model has only been tested within one case study and further examples help to further solidify the designed artifact.

Lastly, the results of the prognostics method were far from optimal. While the decision model guarantees to return the best method for RUL prediction, it does not necessarily mean that the overall result is good. The Backblaze dataset is relatively young and HDD failure prognostics is evidently difficult, nevertheless using a new “untouched” dataset was deemed necessary to guarantee a low level of bias in the selection of the optimal method.

7 Conclusion

All in all, the research was conducted with scientific rigor through the design science IS research framework by HEVNER ET AL. (2004) under considerations of the design science guidelines. Tab. 7 summarizes how the different guidelines have been incorporated in the research.

The design science approach was used to construct a tested, tangible and actionable decision model for the selection of fleet-based prognostic methods. This was achieved through three research objectives.

First, pivotal fleet-data-based approaches for fault prognostics were identified and the current state-the-art in fleet PHM was analyzed by carrying out a structured literature review.

Secondly, the artifact, a decision support model for the selection of fleet-based methods for a given prognostics scenario was constructed. The decision model offers synthesis of

Guideline	Description
Guideline 1: Design as an Artifact	The result of the design-scientific research is a viable artifact in the form of a decision model for fleet-based prognostic methods.
Guideline 2: Problem Relevance	The decision model is based on technological, state-of-the-art methods and algorithms and offers a solution for the anticipation of engineered system failures.
Guideline 3: Design Evaluation	The decision model is based on a structured literature review. The utility, quality and efficacy of the artifact is demonstrated within a case study that was implemented under rigorous research frameworks and is evaluated by scientifically proven methods and measures.
Guideline 4: Research Contributions	The contribution of the model as a design artifact is clear and verifiable. It contributes to research by generalizing and synthesizing existing literature within a decision model that can also be easily by practitioners.
Guideline 5: Research Rigor	For each of the three research objectives, rigorous research frameworks have been used, such as the framework of VOM BROCKE ET AL. (2009) or various prognostic frameworks.
Guideline 6: Design as a Search Process	The design of the artifact has been constructed as a search process. Means from various domains, such as engineering, mathematics, decision-making, statistics and artificial intelligence have been utilized to construct a model that can be used for any engineered system.
Guideline 7: Communication of Research	The research is presented in a manner that is understandable for management-oriented people, while it also facilitates effective and efficient implementation of prognostic methods for a technology-oriented audience.

Adapted from (Hevner et al. 2004, p. 83)

Tab. 7 Realization of the Design Science Guidelines

the methods that have been identified in the first step and allows for a simple selection and actionable implementation of best-fit methods.

Lastly, the decision model was tested within a case study and it was shown that the model performs well.

Nevertheless, this thesis offers potential for future research and the following research agenda is proposed.

PHM terminology. An extensive terminology for PHM is required. Especially fleets are novel concepts that lack a common understanding. A lot of approaches do not communicate well which type of fleet was used and how.

Decoupling of steps. As of now, fleet-based prognostic methods are tightly coupled processes. This makes current literature specific to the problem at hand, even though many components (i.e. fleet identification, usage or prognostic steps) could be reused and combined.

Neuro-fuzzy systems and fleets. It must be analyzed why NFSs are less predominantly used in a fleet-context than they are in general prognostics.

Performance measurement. In this work, twelve decision relevant criteria are proposed. Future research of prognostic methods should always show for which fleet type, etc. it is suited and how well it performs in regard to low sample sizes, missing data, general accuracy etc. Overall more benchmarking should be performed.

Validation of the decision model. Lastly, the decision model must be applied in further case studies and its validity be proven.

References

Acronis. 2019a. “9105: S.M.A.R.T. Attribute: Reallocated Sectors Count.” (https://kb.acronis.com/content/9105, accessed June 15, 2019).

Acronis. 2019b. “9110: S.M.A.R.T. Attribute: Spin Retry Count.” (https://kb.acronis.com/content/9110, accessed June 15, 2019).

Acronis. 2019c. “9132: S.M.A.R.T. Attribute: Reallocation Event Count.” (https://kb.acronis.com/content/9132, accessed June 15, 2019).

Acronis. 2019d. “9264: Acronis Drive Monitor: Disk Health Calculation.” (https://kb.acronis.com/content/9264, accessed June 25, 2019).

Al-Dahidi, S., Di Maio, F., Baraldi, P., and Zio, E. 2016. “Remaining Useful Life Estimation in Heterogeneous Fleets Working under Variable Operating Conditions,” *Reliability Engineering and System Safety* (156:2016), pp. 109–124.

Al-Dahidi, S., Di Maio, F., Baraldi, P., and Zio, E. 2017a. “A Locally Adaptive Ensemble Approach for Data-Driven Prognostics of Heterogeneous Fleets,” *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* (231:4), Department of Energy, Politecnico di Milano, Via La Masa 34, Milan, 20156, Italy: SAGE Publications Ltd, pp. 350–363.

Al-Dahidi, S., Di Maio, F., Baraldi, P., and Zio, E. 2017b. “A Switching Ensemble Approach for Remaining Useful Life Estimation of Electrolytic Capacitors,” in *26th European Safety and Reliability Conference, ESREL 2016*, W. L., R. M., and B. T. (eds.), Energy Department, Politecnico di Milano, Milan, Italy: CRC Press/Balkema, p. 325.

de Almeida, A. T., and Bohoris, G. A. 1995. “Decision Theory in Maintenance Decision Making,” *Journal of Quality in Maintenance Engineering* (3:1), pp. 39–45.

Assari, A., Mahesh, T. M., and Assari, E. 2012. “Role of Public Participation in Sustainability of Historical City: Usage of TOPSIS Method,” *Indian Journal of Science and Technology* (5:3), pp. 2289–2294.

Atamuradov, V., Medjaher, K., Dersin, P., Lamoureux, B., and Zerhouni, N. 2017. “Prognostics and Health Management for Maintenance Practitioners-Review, Implementation and Tools Evaluation,” *International Journal of Prognostics and Health Management* (8:31), pp. 1–31. (https://www.researchgate.net/publication/321747601).

Babu, G. S., Zhao, P., and Li, X. L. 2016. “Deep Convolutional Neural Network Based Regression Approach for Estimation of Remaining Useful Life,” in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*.

Backblaze. 2014. “Hard Drive SMART Stats.” (https://www.backblaze.com/blog/hard-drive-smart-stats/, accessed June 15, 2019).

Backblaze. 2016. “What SMART Stats Tell Us About Hard Drives.” (https://www.backblaze.com/blog/what-smart-stats-indicate-hard-drive-failures/, accessed June 15, 2019).

Backblaze. 2018. “Backblaze Durability Is 99.999999999% — And Why It Doesn’t Matter.” (https://www.backblaze.com/blog/cloud-storage-durability/, accessed June 17, 2019).

Backblaze. 2019. “Hard Drive Data and Stats.” (https://www.backblaze.com/b2/hard-drive-test-data.html, accessed June 14, 2019).

Baker, M. J. 2000. “Writing a Literature Review,” *The Marketing Review* (1:2), pp. 219–247.

Baptista, M., De Medeiros, I. P., Malere, J. P., Prendinger, H., Nascimento, C. L., and Henriques, E. 2016. “Improved Time-Based Maintenance in Aeronautics with Regressive Support Vector Machines,” in *Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM* (Vol. 2016-Octob), pp. 1–10.

Bevilacqua, M., and Braglia, M. 2000. “Analytic Hierarchy Process Applied to Maintenance Strategy Selection,” *Reliability Engineering and System Safety* (70), p. 7183.

Bezdek, J. C., Ehrlich, R., and Full, W. 1984. “FCM: The Fuzzy c-Means Clustering Algorithm,” *Computers and Geosciences* (10:2–3), pp. 191–203.

Blancke, O., Tahan, A., Komljenovic, D., Amyot, N., Lévesque, M., and Hudon, C. 2018. “A Holistic Multi-Failure Mode Prognosis Approach for Complex Equipment,” *Reliability Engineering and System Safety* (180), École de technologie supérieure, Montréal, QC H3C 1K3, Canada: Elsevier Ltd, pp. 136–151.

Bracke, S. 2014. “RAPP: A New Approach for Risk Prognosis on Technical Complex Products in Automotive Engineering,” in *European Safety and Reliability Conference, ESREL 2013*, Dept. of Safety Engineering and Risk Management, University of Wuppertal, Wuppertal, Germany: shers, pp. 1333–1338.

Bracke, S., and Sochacki, S. 2015. “The Estimation and Prognosis of Failure Behaviour in Product Fleets within the Usage Phase—RAPP Method,” in *25th European Safety and Reliability Conference, ESREL 2015*, Z. E., P. L., K. W., Sudret B., and Stojadinovic B. (eds.), Safety Engineering and Risk Management, University of Wuppertal, Wuppertal, Germany: CRC Press/Balkema, pp. 1141–1148.

Breiman, L. 2001. “Random Forrests,” *Machine Learning* (45:1), pp. 5–32.

vom Brocke, J., Simons, A., Niehaves, B., Riemer, K., Plattfaut, R., and Cleven, A. 2009. “Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process,” *ECIS* (9), pp. 2206–2217.

Caruana, R., and Niculescu-Mizil, A. 2006. “An Empirical Comparison of Supervised Learning Algorithms,” in *Proceedings of the 23rd IInternational Conference on*

Machine Learning, pp. 161–168. (www.cs.cornell.edu).

Chemweno, P., Pintelon, L., Van Horenbeek, A., and Muchiri, P. 2015. "Development of a Risk Assessment Selection Methodology for Asset Maintenance Decision Making: An Analytic Network Process (ANP) Approach," *International Journal of Production Economics* (170), pp. 663–676.

Chen, C. T. 2000. "Extensions of the TOPSIS for Group Decision-Making under Fuzzy Environment," *Fuzzy Sets and Systems*.

Chen, S.-J., and Hwang, C.-L. 1992. "Fuzzy Multiple Attribute Decision Making: Methods and Applications," *Lecture Notes in Economics and Mathematical Systems*, New York: Springer-Verlag.

Cooper, H. M. 1988. "Organizing Knowledge Synthesis: A Taxonomy of Literature Reviews," *Knowledge in Society* (1:1), pp. 104–126.

Cox, D. R. 1972. "Regression Models and Life-Tables," *Journal of the Royal Statistical Society. Series B (Methodological)* (34:2), pp. 187–220.

Davidson-Pilon, C. 2019a. "Lifelines." (https://lifelines.readthedocs.io/en/latest/, accessed June 19, 2019).

Davidson-Pilon, C. 2019b. *Lifelines Documentation*. (https://github.com/CamDavidsonPilon/lifelines/blob/master/docs/Survival Regression.rst).

Duong, P. L. T., Park, H., and Raghavan, N. 2018. "Application of Multi-Output Gaussian Process Regression for Remaining Useful Life Prediction of Light Emitting Diodes," *Microelectronics Reliability* (88–90), Engineering Product Development Pillar, Singapore University of Technology and Design487372, Singapore: Elsevier Ltd, pp. 80–84.

Duong, P. L. T., and Raghavan, N. 2019. "Prognostic Health Management for LED with Missing Data: Multi-Task Gaussian Process Regression Approach," in *2018 Prognostics and System Health Management Conference, PHM-Chongqing 2018*, D. P., L. C., Y. S., D. P., and S. R.-V. (eds.), Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design (SUTD), Singapore, 487372, Singapore: Institute of Electrical and Electronics Engineers Inc., pp. 1182–1187.

Elattar, H. M., Elminir, H. K., and Riad, A. M. 2016. "Prognostics: A Literature Review," *Complex & Intelligent Systems* (2:2), pp. 125–154.

Fagang, Z., Jin, C., Lei, G., and Xinglin, L. 2009. "Neuro-Fuzzy Based Condition Prediction of Bearing Health," *JVC/Journal of Vibration and Control* (15:7), pp. 1079–1091.

Felix, S., Kon, S., Nie, J., and Horowitz, R. 2008. "Strain Sensing With Piezoelectric Zinc Oxide Thin Films for Vibration Suppression in Hard Disk Drives," in *ASME 2008 Dynamic Systems and Control Conference, Parts A and B*, ASME, pp. 755–762.

Flyvbjerg, B. 2006. “Five Misunderstandings About Case-Study Research,” *Qualitative Inquiry* (12:2), pp. 219–245.

Frisk, E., Krysander, M., and Larsson, E. 2014. “Data-Driven Lead-Acid Battery Prognostics Using Random Survival Forests,” in *PHM 2014 - Proceedings of the Annual Conference of the Prognostics and Health Management Society 2014*, pp. 1–10.

Fujitsu. 2019. “MHT2080AT, MHT2060AT, MHT2040AT MHT2030AT, MHT2020AT DISK DRIVES PRODUCT MANUAL.” (https://www.fujitsu.com/downloads/COMP/fcpa/hdd/discontinued/mht20xxat_prod-manual.pdf, accessed June 15, 2019).

Gebraeel, N., Lawley, M., Liu, R., and Parmeshwaran, V. 2004. “Residual Life Predictions from Vibration-Based Degradation Signals: A Neural Network Approach,” *IEEE Transactions on Industrial Electronics*.

Gebraeel, N. Z., and Lawley, M. A. 2008. “A Neural Network Degradation Model for Computing and Updating Residual Life Distributions,” *IEEE Transactions on Automation Science and Engineering*.

Gheyas, I. A., and Smith, L. S. 2010. “Feature Subset Selection in Large Dimensionality Domains,” *Pattern Recognition* (43:1), pp. 5–13.

González, D., Piña, M., and Torres, L. 2008. “Estimation of Parameters in Cox’s Proportional Hazard Model: Comparisons between Evolutionary Algorithms and the Newton-Raphson Approach,” in *Gelbukh A., Morales E.F. (Eds) MICAI 2008: Advances in Artificial Intelligence. MICAI 2008. Lecture Notes in Computer Science, Vol 5317.*, Berlin: Springer, pp. 513–523.

Guillén, A. J., Crespo, A., Macchi, M., and Gómez, J. 2016. “On the Role of Prognostics and Health Management in Advanced Maintenance Systems,” *Production Planning and Control* (46:9), pp. 991–1004.

Heimes, F. O. 2008. “Recurrent Neural Networks for Remaining Useful Life Estimation,” in *2008 International Conference on Prognostics and Health Management, PHM 2008*.

Heng, A., Zhang, S., Tan, A. C. C., and Mathew, J. 2009. “Rotating Machinery Prognostics: State of the Art, Challenges and Opportunities,” *Mechanical Systems and Signal Processing* (23:3), pp. 724–739.

Hevner, March, Park, and Ram. 2004. “Design Science in Information Systems Research,” *MIS Quarterly* (28:1), pp. 75–105.

Hewlett Packard. 2007. “SMART IV Technology on HP Business Desktop Hard Drives,” pp. 1–4. (http://h10032.www1.hp.com/ctg/Manual/c01159621, accessed June 15, 2019).

Hoffman, P. C. 2009. “Fleet Management Issues and Technology Needs,” *International Journal of Fatigue* (31:11–12), Airframe Reliability and Risk Assessment Team Lead, NAVAIR 4.3.3 Structures Division, Bldg 2187, Suite 2340A, 48110 Shaw

Rd., Patuxent River, MD 20670-1906, United States, pp. 1631–1637.

Hu, C., Youn, B. D., Wang, P., and Taek Yoon, J. 2012. "Ensemble of Data-Driven Prognostic Algorithms for Robust Prediction of Remaining Useful Life," *Reliability Engineering and System Safety*.

Hu, Y., Baraldi, P., Di Maio, F., and Zio, E. 2015. "A Particle Filtering and Kernel Smoothing-Based Approach for New Design Component Prognostics," *Reliability Engineering and System Safety* (134), Politecnico di Milano, Department of Energy, via Ponzio 34/3, Milan, 20133, Italy: Elsevier Ltd, pp. 19–31.

Huang, R., Xi, L., Li, X., Richard Liu, C., Qiu, H., and Lee, J. 2007. "Residual Life Predictions for Ball Bearings Based on Self-Organizing Map and Back Propagation Neural Network Methods," *Mechanical Systems and Signal Processing*.

Huang, S., Fu, S., Zhang, Q., and Shi, W. 2015. "Characterizing Disk Failures with Quantified Disk Degradation Signatures: An Early Experience," in *2015 IEEE International Symposium on Workload Characterization*, IEEE, October, pp. 150–159.

Hubbard, C., Bavlsik, J., Hegde, C., and Hu, C. 2016. "Data-Driven Prognostics of Lithium-Ion Rechargeable Battery Using Bilinear Kernel Regression," in *Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM* (Vol. 2016-Octob).

Huber, P. J. 1981. *Robust Statistics*, (2nd ed.), New Jersey: John Wiley & Sons.

Hwang, C.-L., and Yoon, K. 1981. "Multiple Attribute Decision Making: Methods and Applications A State-of-the-Art Survey," *Springer-Verlag*, New York.

Ilangkumaran, M., and Kumanan, S. 2009. "Selection of Maintenance Policy for Textile Industry Using Hybrid Multi-Criteria Decision Making Approach," *Journal of Manufacturing Technology Management* (20:7), pp. 1009–1022.

Ishwaran, H. 2007. "Variable Importance in Binary Regression Trees and Forests," *Electronic Journal of Statistics* (1), pp. 519–537.

Ishwaran, H., and Kogalur, U. B. 2019. *Package 'RandomForestSRC,'* pp. 1–103. (https://cran.r-project.org/web/packages/randomForestSRC/randomForestSRC.pdf).

Ishwaran, H., Kogalur, U. B., Blackstone, E. H., and Lauer, M. S. 2008. "Random Survival Forests," *The Annals of Applied Statistics* (2:3), pp. 841–860.

ISO13381-1. 2015. "Condition Monitoring and Diagnostics of Machines —Prognostics —Part 1: General Guidelines," *International Standard, ISO*.

Jardine, A. K. S. 2019. "EXAKT Condition-Based Optimization Software." (http://www.banak-inc.com/exakt_factsheet.pdf, accessed June 19, 2019).

Jardine, A. K. S., Banjevic, D., Wiseman, M., Buck, S., and Joseph, T. 2001.

"Optimizing a Mine Haul Truck Wheel Motors' Condition Monitoring Program: Use of Proportional Hazards Modeling," *Journal of Quality in Maintenance Engineering* (7:4), Condition-based Maintenance Laboratory, Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, Ont., Canada, pp. 286–301.

Jardine, A. K. S., Lin, D., and Banjevic, D. 2006. "A Review on Machinery Diagnostics and Prognostics Implementing Condition-Based Maintenance," *Mechanical Systems and Signal Processing* (20:7), pp. 1483–1510.

Jordan, D. C., Deline, C., Kurtz, S. R., Kimball, G. M., and Anderson, M. 2018. "Robust PV Degradation Methodology and Application," *IEEE Journal of Photovoltaics* (8:2), National Renewable Energy Laboratory, Golden, CO 80401, United States: IEEE Electron Devices Society, pp. 525–531.

Kammoun, M. A., and Rezg, N. 2018. "Toward the Optimal Selective Maintenance for Multi-Component Systems Using Observed Failure: Applied to the FMS Study Case," *International Journal of Advanced Manufacturing Technology* (96:1–4), Industrial Engineering, Production and maintenance Laboratory, University of Lorraine, Metz, France: Springer London, pp. 1093–1107.

Kan, M. S., Tan, A. C. C., and Mathew, J. 2015. "A Review on Prognostic Techniques for Non-Stationary and Non-Linear Rotating Systems," *Mechanical Systems and Signal Processing* (62–63:2015), pp. 1–20.

Kirubakaran, B., and Ilangkumaran, M. 2016. "Selection of Optimum Maintenance Strategy Based on FAHP Integrated with GRA–TOPSIS," *Annals of Operations Research* (245:1–2), pp. 285–313.

Kogalur, U., and Ishwaran, H. 2016. *Random Forests for Survival, Regression, and Classification*. (https://kogalur.github.io/randomForestSRC/theory.html).

Kotsiantis, S. B. 2007. "Supervised Machine Learning: A Review of Classification Techniques," *Informatica (Ljubljana)* (31:3).

LeCun, Y., Bengio, Y., and Hinton, G. 2015. "Deep Learning," *Nature* (521:28 May 2015), pp. 436–444.

Lee, J., Jin, C., Liu, Z., and Ardakani, H. D. 2017. "Introduction to Data-Driven Methodologies for Prognostics and Health Management," in *Probabilistic Prognostics and Health Management of Energy Systems*.

Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., and Siegel, D. 2014. "Prognostics and Health Management Design for Rotary Machinery Systems - Reviews, Methodology and Applications," *Mechanical Systems and Signal Processing* (42:1–2), Elsevier, pp. 314–334. (http://dx.doi.org/10.1016/j.ymssp.2013.06.004).

Lei, Y., Li, Naipeng, Guo, L., Li, Ningbo, Yan, T., and Lin, J. 2018. "Machinery Health Prognostics: A Systematic Review from Data Acquisition to RUL Prediction," *Mechanical Systems and Signal Processing*.

Leone, G., Cristaldi, L., and Turrin, S. 2017. "A Data-Driven Prognostic Approach

Based on Statistical Similarity: An Application to Industrial Circuit Breakers," *Measurement: Journal of the International Measurement Confederation* (108), Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano, Piazza Leonardo da Vinci, 32, Milano, 20133, Italy: Elsevier B.V., pp. 163–170.

Levy, Y., and Ellis, T. J. 2006. "A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research," *Informing Science* (9:8), pp. 171–181.

Leys, C., Ley, C., Klein, O., Bernard, P., and Licata, L. 2013. "Detecting Outliers: Do Not Use Standard Deviation around the Mean, Use Absolute Deviation around the Median," *Journal of Experimental Social Psychology* (49:4), pp. 764–766.

Lim, P., Goh, C. K., Tan, K. C., and Dutta, P. 2014. "Estimation of Remaining Useful Life Based on Switching Kalman Filter Neural Network Ensemble," in *PHM Conference*, pp. 1–8.

Lima, F. D. S., Pereira, F. L. F., Leite, L. G. M., Gomes, J. P. P., and Machado, J. C. 2018. "Remaining Useful Life Estimation of Hard Disk Drives Based on Deep Neural Networks," in *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE, July, pp. 1–7.

Ling, Y., and Mahadevan, S. 2011. "Integration of Structural Health Monitoring and Fatigue Damage Prognosis," in *8th International Workshop on Structural Health Monitoring 2011: Condition-Based Maintenance and Intelligent Structures* (Vol. 1), Department of Civil and Environmental Engineering, Vanderbilt University, TN 37235, United States, pp. 1341–1348.

Ling, Y., and Mahadevan, S. 2012. "Integration of Structural Health Monitoring and Fatigue Damage Prognosis," *Mechanical Systems and Signal Processing* (28), Department of Civil and Environmental Engineering, Vanderbilt University, TN 37235, United States, pp. 89–104.

Liu, J., Djurdjanovic, D., Ni, J., Casoetto, N., and Lee, J. 2007. "Similarity Based Method for Manufacturing Process Performance Prediction and Diagnosis," *Computers in Industry* (58:6), pp. 558–566.

Lukens, S., and Markham, M. 2018. "Data-Driven Application of PHM to Asset Strategies," in *2018 Annual Conference of the Prognostics and Health Management Society*, pp. 1–12.

Mahdaoui, R., and Mouss, L. H. 2012. "A TSK-Type Recurrent Neuro-Fuzzy Systems for Fault Prognosis," *Journal of Software Engineering and Applications* (05:07), pp. 477–482.

Mardani, A., Jusoh, A., and Zavadskas, E. K. 2015. "Fuzzy Multiple Criteria Decision-Making Techniques and Applications - Two Decades Review from 1994 to 2014," *Expert Systems with Applications* (42:8), pp. 4126–4148.

Mashhadi, A. R., Cade, W., and Behdad, S. 2018. "Moving towards Real-Time Data-Driven Quality Monitoring: A Case Study of Hard Disk Drives," *Procedia*

Manufacturing (26), pp. 1107–1115.

Medina-Oliva, G., Voisin, A., Monnin, M., Peysson, F., Leger, J., and Léger, J.-B. 2012. "Prognostics Assessment Using Fleet-Wide Ontology," in *Annual Conference of the Prognostics and Health Management Society 2012, PHM Conference 2012*, pp. 1–10.

Michau, G., Palmé, T., and Fink, O. 2018. "Fleet PHM for Critical Systems: Bi-Level Deep Learning Approach for Fault Detection," in *Proceedings of the Annual Conference of the Prognostics and Health Management Society 2018*, pp. 1–10.

Nicchiotti, G., and Rüegg, J. 2014. *Data-Driven Prediction of Unscheduled Maintenance Replacements in a Fleet of Commercial Aircrafts*, pp. 1–10.

NIST. 2019. "MAD TO MEDIAN." (https://www.itl.nist.gov/div898/software/dataplot/refman2/auxillar/madmed.htm, accessed June 16, 2019).

Nuhic, A., Bergdolt, J., Spier, B., Buchholz, M., and Dietmayer, K. 2018. "Battery Health Monitoring and Degradation Prognosis in Fleet Management Systems," *World Electric Vehicle Journal* (9:3), Daimler AG, Stuttgart, D-70546, Germany: MDPI AG.

Ossadnik, W., and Lange, O. 1999. "AHP-Based Evaluation of AHP-Software," *European Journal of Operational Research* (118:3), pp. 578–588.

Palau, A. S., Liang, Z., Lütgehetmann, D., and Parlikad, A. K. 2019. "Collaborative Prognostics in Social Asset Networks," *Future Generation Computer Systems* (92), Department of Engineering, Institute for Manufacturing, University of Cambridge, Cambridge, CB3 0FS, United Kingdom: Elsevier B.V., pp. 987–995.

Papathanasiou, J., and Ploskas, N. 2018. "TOPSIS," in *Multiple Criteria Decision Aid*, Cham: Springer International Publishing, pp. 1–30.

Pecht, M., and Kumar, S. 2008. "Data Analysis Approach for System Reliability, Diagnostics and Prognostics," in *Pan Pacific Microelectronics Symposium*, pp. 1–9.

Peel, L. 2008. "Data Driven Prognostics Using a Kalman Filter Ensemble of Neural Network Models," in *2008 International Conference on Prognostics and Health Management, PHM 2008*.

Peng, Y., Wang, H., Wang, J., Liu, D., and Peng, X. 2012. "A Modified Echo State Network Based Remaining Useful Life Estimation Approach," in *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*.

Pinheiro, E., Weber, W.-D., and Barroso, L. A. 2007. "Failure Trends in a Large Disk Drive Population," in *USENIX Conference on File and Storage Technologies*, pp. 1–13.

Poot-Geertman, P., Huisman, B., and van Rijn, C. F. H. 2015. “Application of a Maintenance Engineering Decision Method for Railway Operation: Managing Fleet Performance, Cost, and Risk,” in *25th European Safety and Reliability Conference, ESREL 2015*, Z. E., P. L., K. W., Sudret B., and Stojadinovic B. (eds.), NedTrain—Fleet Services, Netherlands: CRC Press/Balkema, pp. 1863–1870.

Raghavan, N., and Frey, D. D. 2016. “Real-Time Update of Multi-State System Reliability Using Prognostic Data-Driven Techniques,” in *Annual Reliability and Maintainability Symposium, RAMS 2016* (Vol. 2016-April), Singapore University of Technology and Design, 8 Somapah Road, Singapore, 487 372, Singapore: Institute of Electrical and Electronics Engineers Inc.

Razavi-Far, R., Chakrabarti, S., Saif, M., Zio, E., and Palade, V. 2018. “Extreme Learning Machine Based Prognostics of Battery Life,” *International Journal on Artificial Intelligence Tools* (27:8), Department of Electrical and Computer Engineering, University of Windsor Windsor, Canada: World Scientific Publishing Co. Pte Ltd.

Riad, A. M., Elminir, H. K., and Elattar, H. M. 2010. “Evaluation of Neural Networks in the Subject of Prognostics as Compared to Linear Regression Model,” *International Journal of Engineering & Technology*.

Rigamonti, M., Baraldi, P., Alessi, A., Zio, E., Astigarraga, D., and Galarza, A. 2018. “An Ensemble of Component-Based and Population-Based Self-Organizing Maps for the Identification of the Degradation State of Insulated-Gate Bipolar Transistors,” *IEEE Transactions on Reliability* (67:3), Politecnico di Milano, Milan, 20156, Italy: Institute of Electrical and Electronics Engineers Inc., pp. 1304–1313.

Rigamonti, M., Baraldi, P., Zio, E., Roychoudhury, I., Goebel, K., and Poll, S. 2016. “Echo State Network for the Remaining Useful Life Prediction of a Turbofan Engine,” in *EUROPEAN CONFERENCE OF THE PROGNOSTICS AND HEALTH MANAGEMENT SOCIETY 2016*.

Saaty, T. L. 1980. “The Analytic Hierarchy Process.,” *Decision Analysis*, New York: McGraw-Hill.

Saaty, T. L. 1987. “The Analytic Hierarchy Process-What It Is and How It Is Used,” *Mathematical Modelling* (9:3–5), pp. 161–176.

Saaty, T. L. 1990. “How to Make a Decision: The Analytic Hierarchy Process,” *European Journal of Operational Research* (48:1), pp. 9–26.

Saaty, T. L. 1996. “Decision Making with Dependence and Feedback: The Analytic Network Process.,” *RWS Publications*, Pittsburgh, Pennsylvania: RWS Publications.

Saaty, T. L. 1999. “Fundamentals of the Analytic Network Process,” in *Proceedings of the 5th International Symposium on the Analytic Hierarchy Process*, pp. 1–14.

Saaty, T. L. 2005. *Theory and Applications of the Analytic Network Process: Decision*

Making with Benefits, Opportunities, Costs, and Risks, (3rd ed.), Pittsburgh, Pennsylvania: RWS Publications.

Saaty, T. L., and Tran, L. T. 2007. "On the Invalidity of Fuzzifying Numerical Judgments in the Analytic Hierarchy Process," *Mathematical and Computer Modelling* (46:7–8), pp. 962–975.

Saaty, T. L., and Vargas, L. G. 2006. "Decision Making with the Analytic Network Process.Economic, Political, Social and Technological Applications with Benefits, Opportunities, Costs and Risks," *International Series in Operations Research Management Science*. (https://doi.org/10.1007/978-1-4614-7279-7).

Sadeghi, A., and Manesh, R. A. 2012. "The Application of Fuzzy Group Analytic Network Process to Selection of Best Maintenance Strategy- A Case Study in Mobarakeh Steel Company, Iran," *Procedia - Social and Behavioral Sciences* (62:2012), pp. 1378 – 1383.

Saxena, A. 2014. "International Journal of Prognostics and Health Management."

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., and Schwabacher, M. 2008. "Metrics for Evaluating Performance of Prognostic Techniques," in *2008 International Conference on Prognostics and Health Management, PHM 2008*, pp. 1–17.

Saxena, A., and Goebel, K. 2008. "PHM08 Challenge Data Set," Moffett Field, CA: NASA Ames Research Center. (http://ti.arc.nasa.gov/project/prognostic-data-repository, accessed June 25, 2019).

Saxena, A., Goebel, K., Simon, D., and Eklund, N. 2008. "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation," in *2008 International Conference on Prognostics and Health Management, PHM 2008*.

Schwabacher, M., and Goebel, K. 2007. "A Survey of Artificial Intelligence for Prognostics," in *AAAI Fall Symposium*, pp. 107–114.

Shahin, A., Pourjavad, E., and Shirouyehzad, H. 2012. "Selecting Optimum Maintenance Strategy by Analytic Network Process with a Case Study in the Mining Industry," *International Journal of Productivity and Quality Management* (10:4).

Shao, Y., and Nezu, K. 2000. "Prognosis of Remaining Bearing Life Using Neural Networks," *Proceedings of the Institution of Mechanical Engineers. Part I, Journal of Systems and Control Engineering*.

Sharma, S. 2008. "Toward an Integrated Knowledge Discovery and Data Mining Process Model," Virginia Commonwealth University.

Shen, J., Wan, J., Lim, S. J., and Yu, L. 2018. "Random-Forest-Based Failure Prediction for Hard Disk Drives," *International Journal of Distributed Sensor Networks* (14:11), pp. 1–15.

Si, X. S., Wang, W., Hu, C. H., and Zhou, D. H. 2011. "Remaining Useful Life

Estimation - A Review on the Statistical Data Driven Approaches," *European Journal of Operational Research* (213:2011), pp. 1–14.

Sikorska, J. Z., Hodkiewicz, M., and Ma, L. 2011. "Prognostic Modelling Options for Remaining Useful Life Estimation by Industry," *Mechanical Systems and Signal Processing* (25:5), pp. 1803–1836.

Singh, A., Thakur, N., and Sharma, A. 2016. "A Review of Supervised Machine Learning Algorithms," in *2016 3rd International Conference on Computing for Sustainable Global Development*, pp. 1310–1315.

Singh, S., Subramania, H. S., Holland, S. W., and Davis, J. T. 2012. "Decision Forest for Root Cause Analysis of Intermittent Faults," *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews* (42:6), Samsung India RandDCenter, Bengaluru 560043, India, pp. 1818–1827.

Subbiah, S., and Turrin, S. 2015. "Extraction and Exploitation of R&M Knowledge from a Fleet Perspective," in *61st Annual Reliability and Maintainability Symposium, RAMS 2015* (Vol. 2015-May), ABB Corporate Research Center Germany, Industrial Software and Applications, Life Cycle Science, Wallstadter Str. 59, Ladenburg, 68526, Germany: Institute of Electrical and Electronics Engineers Inc.

Sutharssan, T., Stoyanov, S., Bailey, C., and Yin, C. 2015. "Prognostic and Health Management for Engineering Systems: A Review of the Data-Driven Approach and Algorithms," *The Journal of Engineering* (2015:7), pp. 215–222.

Teixeira, R. E., Morris, K. E., and Sautter, F. C. 2015. "Accurate Health Estimates from HUMS Vibration Data," in *IEEE Conference on Prognostics and Health Management, PHM 2015*, Reliability and Failure Analysis Laboratory, UAH Research Institute, Huntsville, AL 35899, United States: Institute of Electrical and Electronics Engineers Inc., pp. 1–6.

Thor, J., Ding, S., and Kamaruddin, S. 2013. "Comparison of Multi Criteria Decision Making Methods From The Maintenance Alternative Selection Perspective," *International Journal Of Engineering And Science (IJES)* (2:6), pp. 27–34.

Tinga, T., and Loendersloot, R. 2014. "Aligning PHM, SHM and CBM by Understanding the Physical System Failure Behaviour," in *Proceedings of the European Conference of the Prognostics and Health Management Society*, pp. 162–171.

Tipping, M. 2001. "Sparse Bayesian Learning and the Relevance Vector Machine.," *Journal of Machine Learning Research* (1:2001), pp. 211–244.

Torraco, R. J. 2005. "Writing Integrative Literature Reviews: Guidelines and Examples," *Human Resource Development Review* (4:3), pp. 356–367.

Trapani, N., Macchi, M., and Fumagalli, L. 2015. "Risk Driven Engineering of Prognostics and Health Management Systems in Manufacturing," *IFAC-PapersOnLine* (48:3), pp. 995–1000.

Triantaphyllou, E., Kovalerchuk, B., Mann, L., and Knapp, G. M. 1997. "Determining the Most Important Criteria in Maintenance Decision Making," *Journal of Quality in Maintenance Engineering* (3:1), pp. 16–28.

Trilla, A., Dersin, P., and Cabr, X. 2018. "Estimating the Uncertainty of Brake Pad Prognostics for High-Speed Rail with a Neural Network Feature Ensemble," in *Annual Conference of the Prognostics and Health Management Society*.

Umiliacchi, P., Lane, D., and Romano, F. 2011. "Predictive Maintenance of Railway Subsystems Using an Ontology Based Modelling Approach," in *Proceedings of 9th World Conference on Railway Research*.

Vachtsevanos, G., Lewis, F. L., Roemer, M., Hess, A., and Wu, B. 2006. "Intelligent Fault Diagnosis and Prognosis for Engineering Systems," *Engineering*, New Jersey: Wiley.

Vapnik, V. N. 2009. "An Overview of Statistical Learning Theory," *IEEE Transactions on Neural Networks* (10:5), pp. 1–12.

Venkatasubramanian, V. 2005. "Prognostic and Diagnostic Monitoring of Complex Systems for Product Lifecycle Management: Challenges and Opportunities," *Computers and Chemical Engineering* (29:2005), pp. 1253–1263.

Voisin, A., Levrat, E., Cocheteux, P., and Iung, B. 2010. "Generic Prognosis Model for Proactive Maintenance Decision Support: Application to Pre-Industrial e-Maintenance Test Bed," *Journal of Intelligent Manufacturing* (2010:21), pp. 177–193.

Voisin, A., Medina-oliva, G., Monnin, M., Voisin, A., and Medina-oliva, G. 2013. *Fleet-Wide Diagnostic and Prognostic Assessment To Cite This Version : Fleet-Wide Diagnostic and Prognostic Assessment*.

Voronov, S., Frisk, E., and Krysander, M. 2018. "Data-Driven Battery Lifetime Prediction and Confidence Estimation for Heavy-Duty Trucks," *IEEE Transactions on Reliability* (67:2), Department of Electrical Engineering, Linköping University, Linköping, S-581 83, Sweden: Institute of Electrical and Electronics Engineers Inc., pp. 623–629.

Wagner, C., and Hellingrath, B. 2017. "Fleet Knowledge for Prognostics and Health Management – Identifying Fleet Dimensions and Characteristics for the Categorization of Fleets," in *PHM Society*.

Walstrom, K. A., and Hardgrave, B. C. 2001. "Forums for Information Systems Scholars: III," *Information and Management* (39:2), pp. 117–124.

Wang, T. 2010. "Trajectory Similarity Based Prediction for Remaining Useful Life Estimation," University of Cincinnati.

Wang, W., Hussin, B., and Jefferis, T. 2012. "A Case Study of Condition Based Maintenance Modelling Based upon the Oil Analysis Data of Marine Diesel Engines Using Stochastic Filtering," *International Journal of Production Economics* (136:1), Dongling School of Economics and Management, University

of Science and Technology Beijing, Beijing, China, pp. 84–92.

Wang, W. Q., Golnaraghi, M. F., and Ismail, F. 2004. "Prognosis of Machine Health Condition Using Neuro-Fuzzy Systems," *Mechanical Systems and Signal Processing* (18:4), pp. 813–831.

Wang, Y., Miao, Q., and Pecht, M. 2011. "Health Monitoring of Hard Disk Drive Based on Mahalanobis Distance," in *2011 Prognostics and System Health Management Conference, PHM-Shenzhen 2011*, pp. 1–8.

Wang, Z.-Q., Hu, C.-H., and Fan, H.-D. 2018. "Real-Time Remaining Useful Life Prediction for a Nonlinear Degrading System in Service: Application to Bearing Data," *IEEE/ASME Transactions on Mechatronics* (23:1), High-Tech Institute of Xian, Xian, 710025, China: Institute of Electrical and Electronics Engineers Inc., pp. 211–222.

Wayne, M., and Arres, M. M. 2013. "Modeling Rate of Occurrence of Failures with Log-Gaussian Process Models: A Case Study for Prognosis and Health Management of a Fleet of Vehicles," *International Journal of Performability Engineering* (9:6), University of Maryland, Center for Risk and Reliability, Department of Mechanical Engineering, College Park, MD 20742, United States, pp. 701–713.

Wolak, A. 2018. "TBN Performance Study on a Test Fleet in Real-World Driving Conditions Using Present-Day Engine Oils," *Measurement: Journal of the International Measurement Confederation* (114), Department of Industrial Commodity Science, Cracow University of Economics, ul. Sienkiewicza 4, Kraków, 30-033, Poland: Elsevier B.V., pp. 322–331.

Wu, D., Jennings, C., Terpenny, J., Gao, R. X., and Kumara, S. 2017. "A Comparative Study on Machine Learning Algorithms for Smart Manufacturing: Tool Wear Prediction Using Random Forests," *Journal of Manufacturing Science and Engineering* (139:7), pp. 1–9.

Yang, D., Wang, H., Feng, Q., Ren, Y., Sun, B., and Wang, Z. 2018. "Fleet-Level Selective Maintenance Problem under a Phased Mission Scheme with Short Breaks: A Heuristic Sequential Game Approach," *Computers and Industrial Engineering* (119), School of Reliability and Systems Engineering, Beihang University, Beijing, China: Elsevier Ltd, pp. 404–415.

Zaidan, M. A. Bin. 2014. "Bayesian Approaches for Complex System Prognostics," University of Sheffield.

Zaidan, M. A., Harrison, R. F., Mills, A. R., and Fleming, P. J. 2015. "Bayesian Hierarchical Models for Aerospace Gas Turbine Engine Prognostics," *Expert Systems with Applications* (42:1), Automatic Control and Systems Engineering, University of Sheffield, Mappin StreetS1 3JD, United Kingdom: Elsevier Ltd, pp. 539–553.

Zaidan, M. A., Mills, A. R., Harrison, R. F., and Fleming, P. J. 2016. "Gas Turbine Engine Prognostics Using Bayesian Hierarchical Models: A Variational Approach," *Mechanical Systems and Signal Processing* (70–71), Department of

Automatic Control and Systems Engineering, University of Sheffield, Mappin StreetS1 3JD, United Kingdom: Academic Press, pp. 120–140.

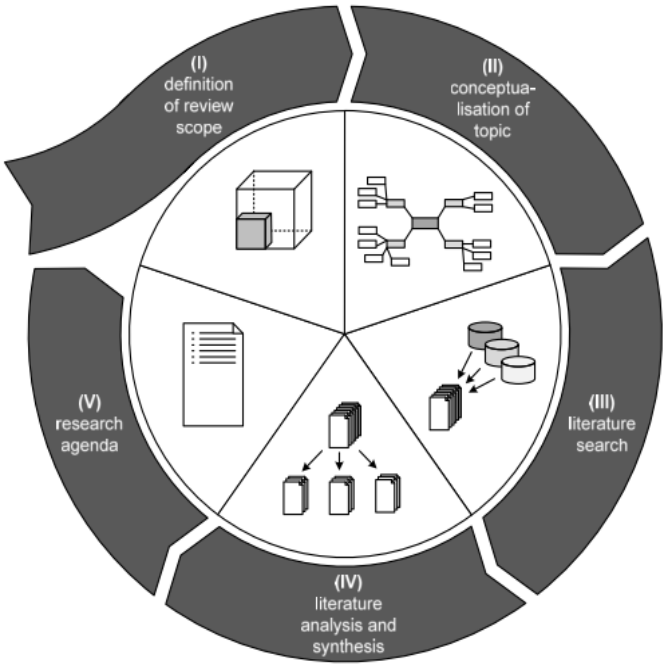
Zaidan, M. A., Relan, R., Mills, A. R., and Harrison, R. F. 2015. “Prognostics of Gas Turbine Engine: An Integrated Approach,” *Expert Systems with Applications* (42:22), Rolls-Royce University Technology Centre, United Kingdom: Elsevier Ltd, pp. 8472–8483.

Zhao, R., Yan, R., Wang, J., and Mao, K. 2017. “Learning to Monitor Machine Health with Convolutional Bi-Directional LSTM Networks,” *Sensors (Switzerland)* (17:2), pp. 273–291.

Appendix

A Framework for Literature Reviews

A.a Methodology of Structured Literature Review



(vom Brocke et al. 2009, p. 2211)

Fig. 20 Framework for Literature Reviews

A.b Definition of Review Scope

a) Focus. It is only important to examine what the outcome of the reviewed literature is (i.e. the algorithms) and in what circumstances they can be applied (i.e. type of data, fleet types).

b) Goal. The findings should be synthesized by summarizing them in concept matrices. Dimensions must be generalized appropriately to construct a universal decision model.

c) Perspective. The review is represented in a neutral manner.

d) Coverage. To make decisions, it is mandatory to know all alternatives. Thus, algorithms were identified by an exhaustive search. Especially the choice of databases and key words is crucial for that. To grant a simpler overview, the results were cited selectively.

e) Organization. The work is structured conceptionally (i.e. by type of algorithm).

f) Audience. The review and its resulting decision model are aimed at practitioners and general scholars.

B Literature Review

B.a Keywords (Figure)

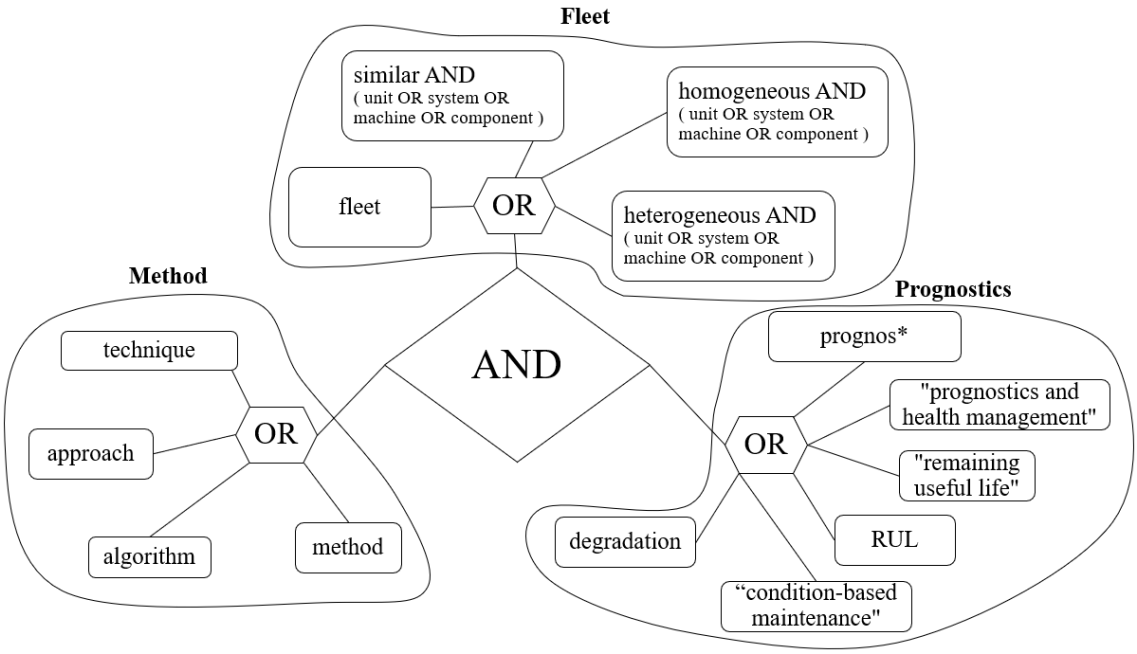


Fig. 21 Key Words

B.b Key Words String

TITLE-ABS-KEY ((prognos* OR "prognostics and health management" OR "remaining useful life" OR RUL OR "Condition-based maintenance" OR degradation) AND (fleet OR ("similar * unit") OR ("similar * system") OR ("similar * machine") OR ("similar * component") OR ("homogeneous * unit") OR ("homogeneous * system") OR ("homogeneous * machine") OR ("homogeneous * component") OR ("heterogeneous * unit") OR ("heterogeneous * system") OR ("heterogeneous * machine") OR ("heterogeneous * component")) AND (method OR algorithm OR approach OR technique))

B.c Included Conference Proceedings

Conference	Hits
Conference of the Prognostics and Health Management Society	26
European Safety and Reliability Conference	11
AHS International Condition Based Maintenance Specialists Meeting	6
Annual Reliability and Maintainability Symposium	5
IEEE International Conference on Prognostics and Health Management	5
Topical Meeting on Probabilistic Safety Assessment and Analysis	4
International Workshop on Structural Health Monitoring	3
Risk, Reliability and Safety: Innovating Theory and Practice	3
Meeting of the Society for Machinery Failure Prevention Technology	3
Prognostics and System Health Management Conference	3
Conference on Probabilistic Safety Assessment and Management	3
Safety and Reliability - Safe Societies in a Changing World	3
International Conference on Condition Monitoring and Diagnosis	3
Workshop on Technical Diagnostics	2
International Conference on Condition Assessment Techniques in Electrical Systems	1
International Instrumentation Symposium	1
International Conference on Condition Monitoring and Machinery Failure Prevention Technologies	1
International Conference on Reliability and Quality in Design	1

Tab. 8 List of Included Conference Proceedings

C Fleet-based Prognostic Methods

C.a Fleet Identification Types (Stage 1)

Source	Count	Fleet Identification
(Al-Dahidi et al. 2016, 2017b, 2017a; Babu et al. 2016; Duong et al. 2018; Duong and Raghavan 2019; Hu et al. 2012; Kammoun and Rezg 2018; Leone et al. 2017; Lim et al. 2014; Liu et al. 2007; Palau et al. 2019; Peel 2008; Riad et al. 2010; Rigamonti et al. 2016; Subbiah and Turrin 2015)	16	Clustering by Categorical Features
(Frisk et al. 2014; Jordan et al. 2018; Lukens and Markham 2018; Poot-Geertman et al. 2015; Rigamonti et al. 2018; Voronov et al. 2018; Wang et al. 2018; Wolak 2018; Zaidan, Relan, et al. 2015; Zaidan, Harrison, et al. 2015; Zaidan et al. 2016)	11	Clustering by Categorical Features
(Voisin et al. 2013)	1	Clustering by Categorical Features
(Baptista et al. 2016; Bracke and Sochacki 2015; Gebraeel et al. 2004; Gebraeel and Lawley 2008; Heimes 2008; Hoffman 2009; Huang et al. 2007; Hubbard et al. 2016; Jardine et al. 2001; Ling and Mahadevan 2011, 2012; Nicchiotti and Rüegg 2014; Nuhic et al. 2018; Peng et al. 2012; Raghavan and Frey 2016; Razavi-Far et al. 2018; Shao and Nezu 2000; Teixeira et al. 2015; Trilla et al. 2018; Wang et al. 2012; Wayne and Arres 2013)	21	No Identification

Tab. 9 Fleet Identification Types and Sources

C.b Fleet Usage Types (Stage 2)

Source	Count	Fleet Usage
(Hoffman 2009; Hubbard et al. 2016; Jardine et al. 2001; Leone et al. 2017; Ling and Mahadevan 2011, 2012; Lukens and Markham 2018; Poot-Geertman et al. 2015; Raghavan and Frey 2016; Subbiah and Turrin 2015; Teixeira et al. 2015; Wang et al. 2012, 2018; Wolak 2018; Zaidan, Harrison, et al. 2015; Zaidan, Relan, et al. 2015; Zaidan et al. 2016)	17	Parameter estimation
(Al-Dahidi et al. 2016, 2017b, 2017a; Baptista et al. 2016; Duong et al. 2018; Duong and Raghavan 2019; Frisk et al. 2014; Nicchiotti and Rüegg 2014; Nuhic et al. 2018; Razavi-Far et al. 2018; Voronov et al. 2018)	11	Training on all fleet data
(Gebraeel et al. 2004; Gebraeel and Lawley 2008; Huang et al. 2007; Jordan et al. 2018; Liu et al. 2007; Rigamonti et al. 2018; Shao and Nezu 2000; Voisin et al. 2013; Wayne and Arres 2013)	9	Regression aggregation
(Babu et al. 2016; Heimes 2008; Hu et al. 2012; Peel 2008; Peng et al. 2012; Riad et al. 2010; Rigamonti et al. 2016; Trilla et al. 2018)	8	Fleet as feature
(Babu et al. 2016; Hu et al. 2012; Lim et al. 2014; Peel 2008; Riad et al. 2010; Rigamonti et al. 2016)	6	Fleet normalization
(Bracke and Sochacki 2015; Kammoun and Rezg 2018; Palau et al. 2019)	3	Other

Tab. 10 Fleet Usage Types and Sources

C.c Publications of Statistical Methods (Stage 3)

Source	Method	Class
(Palau et al. 2019)	Social asset network	Other
(Wang et al. 2018)	Wiener process	Regression Analysis
(Wolak 2018)	Linear regression	Regression Analysis
(Kammoun and Rezg 2018)	K-means and degradation coefficients	Other
(Jordan et al. 2018)	Year-on-year regression	Other
(Lukens and Markham 2018)	Weibull distribution	Reliability Function
(Leone et al. 2017)	Monte Carlo of RUL distribution	Regression Analysis
(Zaidan et al. 2016)	Bayesian hierarchical model	Bayesian Statistics
(Raghavan and Frey 2016)	Markov model with particle filter	Markov Model
(Hubbard et al. 2016)	Robust sparse regression	Regression Analysis
(Zaidan, Relan, et al. 2015)	Bayesian hierarchical model	Bayesian Statistics
(Zaidan, Harrison, et al. 2015)	Bayesian hierarchical model (with MCMC)	Bayesian Statistics
(Teixeira et al. 2015)	Sequential Monte Carlo	Markov Model
(Subbiah and Turrin 2015)	Monte Carlo simulation	Regression Analysis
(Bracke and Sochacki 2015)	Classical statistics	Regression Analysis
(Poot-Geertman et al. 2015)	Weibull distribution and Monte Carlo simulations	Reliability Function
(Ling and Mahadevan 2012)	Bayesian	Bayesian Statistics
(Wang et al. 2012)	Weibull distribution	Reliability Function
(Ling and Mahadevan 2011)	Bayesian	Bayesian Statistics
(Hoffman 2009)	Bayesian updating of RUL distribution and Monte Carlo simulation	Reliability Function
(Liu et al. 2007)	Match matrix	Regression Analysis
(Jardine et al. 2001)	Cox regression	Reliability Function

Tab. 11 Summary of Statistical Methods

C.d Publications of Artificial Intelligence Methods (Stage 3)

Source	Method	Class
(Duong and Raghavan 2019)	Multi-output Gaussian process regression	Gaussian Process Regression
(Razavi-Far et al. 2018)	Extreme learning	Artificial Neural Networks
(Rigamonti et al. 2018)	Self-organizing maps	Artificial Neural Network
(Duong et al. 2018)	Multi-output Gaussian process regression	Gaussian Process Regression
(Voronov et al. 2018)	Random surival forest	Random Forest
(Trilla et al. 2018)	Feed forward back-propagation network	Artificial Neural Network
(Nuhic et al. 2018)	Support vector regression	Vector Machine
(Rigamonti et al. 2016)	Echo state network	Artificial Neural Network
(Babu et al. 2016)	Deep convolutional network	Artificial Neural Network
(Baptista et al. 2016)	Support vector machine	Vector Machine
(Frisk et al. 2014)	Random surival forest	Random Forest
(Nicchiotti and Rüegg 2014)	Support vector machine	Vector Machine
(Wayne and Arres 2013)	Log-Gaussian process regres-sion	Gaussian Process Regression
(Voisin et al. 2013)	Relevance vector machine	Vector Machine
(Peng et al. 2012)	Echo state network	Artificial Neural Network
(Riad et al. 2010)	Multi-layer perceptron	Artificial Neural Network
(Gebraeel and Lawley 2008)	Feed forward back-propagation network	Artificial Neural Network
(Heimes 2008)	Recurrent neural network	Artificial Neural Network
(Huang et al. 2007)	Feed forward back-propagation network	Artificial Neural Network
(Gebraeel et al. 2004)	Feed forward back-propagation network	Artificial Neural Network
(Shao and Nezu 2000)	Feed forward back-propagation network	Artificial Neural Network

Tab. 12 Summary of Artificial Intelligence Methods

C.e Publications of Ensemble Methods (Stage 3)

Source	Method	Class
(Al-Dahidi et al. 2017a)	Discrete-Time Finite-State Semi-Markov Model + Neuro-Fuzzy Ensemble	Ensemble
(Al-Dahidi et al. 2017b)	Discrete-Time Finite-State Semi-Markov Model + Neuro-Fuzzy Ensemble	Ensemble
(Al-Dahidi et al. 2016)	Discrete-Time Finite-State Semi-Markov Model + Neuro-Fuzzy Ensemble	Ensemble
(Lim et al. 2014)	Ensemble of MLPs	Ensemble
(Hu et al. 2012)	Ensemble of RVM, SVM, SBI, BLR, RNN	Ensemble
(Peel 2008)	Multi-layer perceptron and RBF ensemble	Ensemble

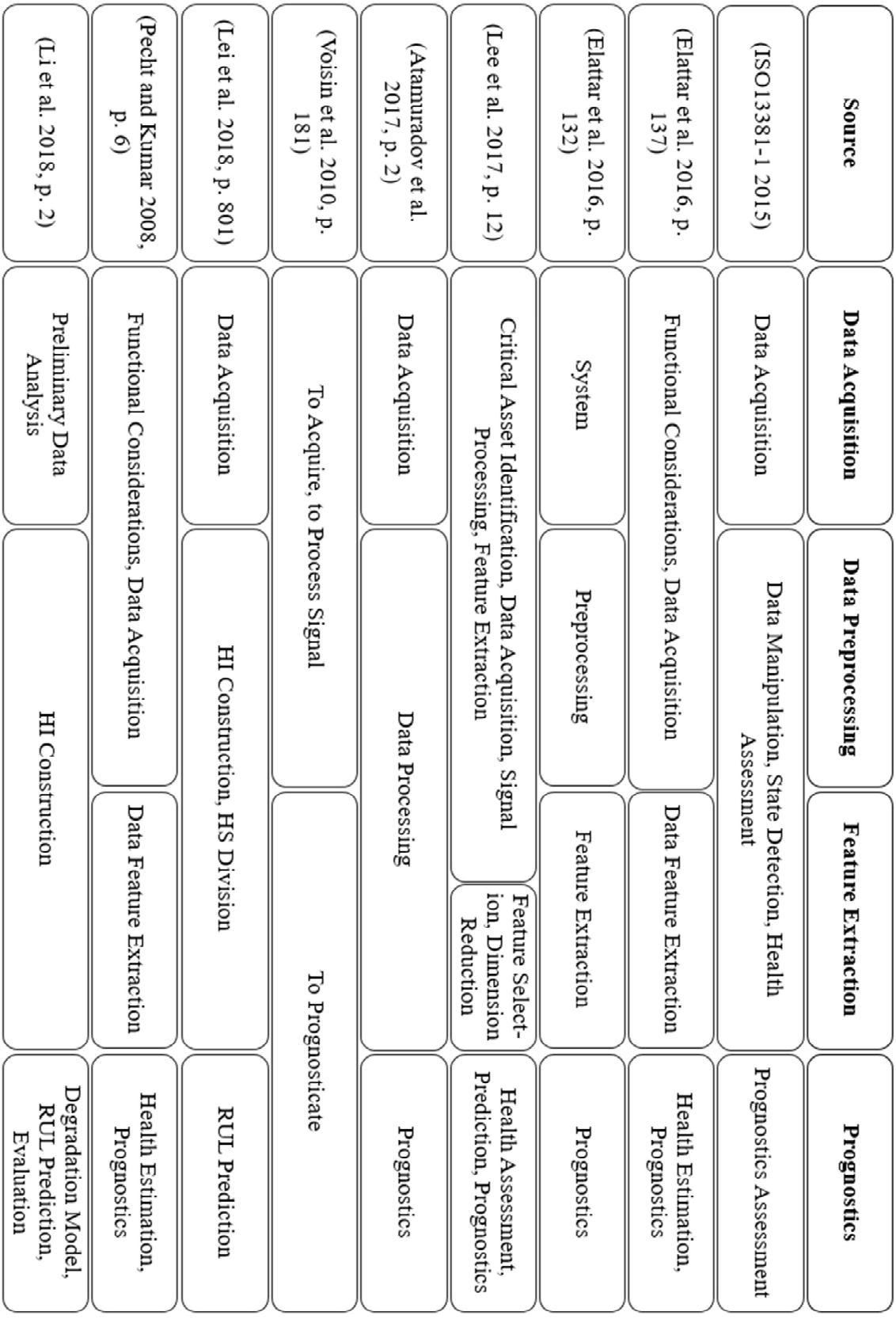
Tab. 13 Summary of Ensemble Methods

D TOPSIS Python Program

```
import pandas as pd
import numpy as np
import math
from scipy.spatial import distance

### Step 1: Alternative matrix and criteria weights

# Possible fleet_type = ['Identical', 'Homogeneous', 'Heterogeneous']
fleet_type = ['Homogeneous']
# Possible fleet_feature_type = ['Numerical', 'Categorical', 'Seman-
tics', 'None']
fleet_feature_type = ['Categorical']
# Possible output_type = ['Point-estimate', 'Interval', 'Distribu-
tion']
output_type = ['Point-estimate', 'Interval', 'Distribution']
# Alternative/Criteria matrix
data = pd.read_csv("Results.csv", sep = ";")
# Filter on elimination criteria
data = data[(data['Fleet Type'].isin(fleet_type)) &
            (data['Output Type'].isin(output_type)) &
            (data['Fleet Feature Type'].isin(fleet_fea-
ture_type))].copy().reset_index(drop=True)
A = data.iloc[:,4:13].values.astype(float)

# Criteria weight
#-------- Adjust weight according to use case --------#
w = np.array([1, 5, 0, 10, 5, 5, 0, 0, 3]).astype(float)
#-----------------------------------------------------#

# Normalize criteria
w = w / w.sum()

### Step 2: Normalize matrix
A_numrows = len(A)
A_numcols = len(A[0])

N = A.copy()
for j in range(0, A_numcols):
    A_column_length = 0
    # Calculate column vector length
    for i in range(0, A_numrows):
        A_column_length = (A[i][j]) ** 2 + A_column_length
    A_column_length = math.sqrt(A_column_length)
    # Divide each element by its column vector length
    for i in range(0, A_numrows):
        N[i][j] = (A[i][j]) / A_column_length

### Step 3: Weigh matrix
W = N.copy()
for j in range(0, A_numcols):
    for i in range(0, A_numrows):
        W[i][j]= w[j]* N[i][j]

### Step 4: Compute PIS and NIS
pis = np.max(W, axis = 0)
nis = np.min(W, axis = 0)

### Step 5: Compute distances from each alternative to PIS and NIS
dist_mat = np.zeros((A_numrows, 2))
```

```
for i in range(0, A_numrows):
    dist_mat[i, 0] = distance.euclidean(W[i, :], pis)
    dist_mat[i, 1] = distance.euclidean(W[i, :], nis)

### Step 6 Calculate similarity to PIS and NIS
similarity_mat = np.zeros((A_numrows, 1))
for i in range(0, A_numrows):
    similarity_mat[i] = dist_mat[i, 1]/(dist_mat[i, 1] + dist_mat[i,
0])

result = pd.concat([data.iloc[:,0], pd.DataFrame(similarity_mat)],
axis=1)
result.columns = ['Alternative', 'Score']
print(result.sort_values(by=['Score'], ascending = False))
```

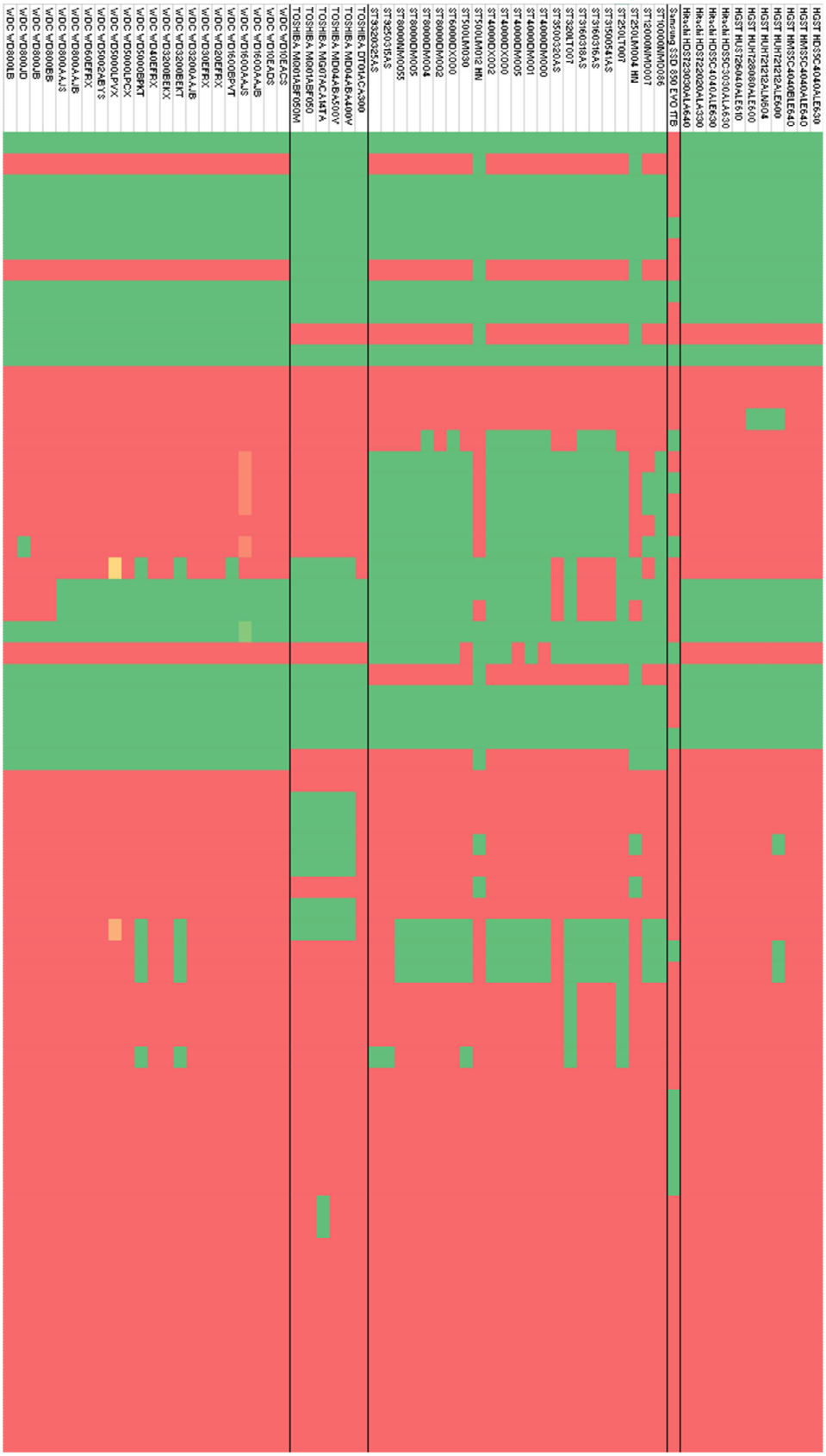
E Overview of Prognostics Frameworks



Own depiction

Fig. 22 Prognostic Frameworks

F Missing Values per Model and Manufacturer



Own depiction.

Fig. 23 Missing Values per Model and Manufacturer

G Method Selection Equations

G.a Alternative-Criteria Matrix

$$A = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{matrix} M & N & SS & A & R & T & S & E & P \end{matrix} \\ \begin{bmatrix} 1 & 1 & 2 & 1 & 1 & 5 & 5 & 5 & 5 \\ 3 & 1 & 3 & 1 & 1 & 4 & 4 & 4 & 5 \\ 3 & 1 & 3 & 1 & 1 & 4 & 5 & 5 & 5 \\ 4 & 1 & 1 & 2 & 2 & 4 & 4 & 3 & 5 \\ 4 & 2 & 5 & 3 & 4 & 2 & 3 & 3 & 3 \\ 3 & 1 & 3 & 2 & 1 & 3 & 3 & 4 & 3 \\ 5 & 5 & 4 & 3 & 3 & 3 & 2 & 3 & 4 \\ 1 & 3 & 1 & 4 & 3 & 3 & 2 & 1 & 1 \\ 5 & 5 & 4 & 3 & 3 & 2 & 2 & 3 & 4 \\ 4 & 4 & 2 & 4 & 4 & 3 & 1 & 4 & 3 \end{bmatrix} \end{array} \tag{33}$$

G.b Normalized Alternative-Criteria Matrix

$$N = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{matrix} M & N & SS & A & R & T & S & E & P \end{matrix} \\ \begin{bmatrix} .09 & .11 & .21 & .12 & .12 & .46 & .47 & .43 & .4 \\ .27 & .11 & .31 & .12 & .12 & .37 & .38 & .34 & .4 \\ .27 & .11 & .31 & .12 & .12 & .37 & .47 & .43 & .4 \\ .35 & .11 & .1 & .24 & .24 & .37 & .38 & .26 & .4 \\ .35 & .22 & .52 & .36 & .49 & .18 & .28 & .26 & .24 \\ .27 & .11 & .31 & .24 & .12 & .28 & .28 & .34 & .24 \\ .44 & .55 & .41 & .36 & .37 & .28 & .19 & .26 & .32 \\ .09 & .33 & .1 & .48 & .37 & .28 & .19 & .09 & .08 \\ .44 & .55 & .41 & .36 & .37 & .18 & .19 & .26 & .32 \\ .35 & .44 & .21 & .48 & .49 & .28 & .09 & .34 & .24 \end{bmatrix} \end{array} \tag{34}$$

G.c Weighted Normalized Alternative-Criteria Matrix

$$W = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{matrix} M & N & SS & A & R & T & S & E & P \end{matrix} \\ \begin{bmatrix} .00 & .02 & .0 & .04 & .02 & .08 & .0 & .0 & .04 \\ .01 & .02 & .0 & .04 & .02 & .06 & .0 & .0 & .04 \\ .01 & .02 & .0 & .04 & .02 & .06 & .0 & .0 & .04 \\ .01 & .02 & .0 & .08 & .04 & .06 & .0 & .0 & .04 \\ .01 & .04 & .0 & .12 & .08 & .03 & .0 & .0 & .02 \\ .01 & .02 & .0 & .08 & .02 & .05 & .0 & .0 & .02 \\ .01 & .09 & .0 & .12 & .06 & .05 & .0 & .0 & .03 \\ .00 & .05 & .0 & .16 & .06 & .05 & .0 & .0 & .01 \\ .01 & .09 & .0 & .12 & .06 & .03 & .0 & .0 & .03 \\ .01 & .07 & .0 & .16 & .08 & .05 & .0 & .0 & .02 \end{bmatrix} \end{array} \tag{35}$$

G.d Distances to Positive and Negative Ideal Solutions

$$D = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{array}{cc} d_b & d_w \end{array} \\ \begin{bmatrix} .16 & .06 \\ .16 & .05 \\ .16 & .05 \\ .12 & .07 \\ .09 & .11 \\ .13 & .05 \\ .06 & .12 \\ .06 & .14 \\ .07 & .12 \\ .04 & .15 \end{bmatrix} \end{array} \tag{36}$$

G.e Similarities to Positive Ideal Solution

$$S = \begin{array}{c} \\ \text{(Jordan et al. 2018)} \\ \text{(Wang et al. 2012; Jardine et al. 2001)} \\ \text{(Wolak 2018)} \\ \text{(Wang et al. 2018)} \\ \text{(Duong et al. 2018; Duong and Raghavan 2019)} \\ \text{(Poot-Geertman etl al. 2015)} \\ \text{(Zaidan et al. 2015, 2016)} \\ \text{(Rigamonti et al. 2018)} \\ \text{(Zaidan, Harrison, et al. 2015)} \\ \text{(Voronov et al. 2018; Frisk et al. 2014)} \end{array} \begin{array}{c} \begin{array}{c} s_b \end{array} \\ \begin{bmatrix} .27 \\ .22 \\ .22 \\ .35 \\ .55 \\ .26 \\ .68 \\ .68 \\ .65 \\ .79 \end{bmatrix} \end{array} \tag{37}$$

H Results of Prognostics

H.a Optimal Number of Trees, Number of Splitting Variables and Terminal Node Size

Model	Number of Trees	Number of Splitting Variables	Terminal Node Size
WDC WD800AAJS	200	16	1
HGST HMS5C4040ALE640	200	5	2
HGST HMS5C4040BLE640	200	4	1
Hitachi HDS5C3030ALA630	200	9	1
Hitachi HDS5C4040ALE630	200	11	1
Hitachi HDS722020ALA330	200	6	1
Hitachi HDS723030ALA640	200	9	1
ST320LT007	200	20	1
ST500LM012 HN	200	13	1
ST500LM030	200	5	1
ST4000DM001	200	5	1
ST4000DX000	200	16	1
ST6000DX000	200	11	1
ST8000DM002	200	25	1
ST8000NM0055	200	13	1
ST10000NM0086	200	8	1
ST12000NM0007	200	9	1
ST3160318AS	200	9	1
WDC WD30EFRX	200	20	6
TOSHIBA MQ01ABF050	200	20	1
ST4000DM000	200	16	1
Manufacturer	Number of Trees	Number of Splitting Variables	Terminal Node Size
Toshiba	200	16	1
Hitachi	200	11	1
Western Digital	200	9	1
Seagate	200	30	1

Own depticion.

Tab. 14 Optimal Parameters of the Random Survival Forest

H.b MAD to Median and Improvement in RMSE

Model	MADM	Difference in RMSE
WDC WD800AAJS	0.1158281	+151.47%
HGST HMS5C4040ALE640	0.01002627	+556.30%
HGST HMS5C4040BLE640	0.1279465	+402.81%
Hitachi HDS5C3030ALA630	0.2128736	+138.54%
Hitachi HDS5C4040ALE630	0.1059	-19.64%
Hitachi HDS722020ALA330	0.09582741	+354.03%
Hitachi HDS723030ALA640	0.05768517	+181.38%
ST320LT007	0.3198834	+179.42%
ST500LM012 HN	0.2310385	+224.17%
ST500LM030	0.1315901	+409.96%
ST4000DM001	0.00	+75,21%
ST4000DX000	0.1673488	+129.21%
ST6000DX000	0.1918157	+368.90%
ST8000DM002	0.4314592	+113.81%
ST8000NM0055	0.4205356	+71.84%
ST10000NM0086	0.2998602	-14.62%
ST12000NM0007	0.4469317	+31.90%
ST3160318AS	0.314082	+435.04%
WDC WD30EFRX	0.2099889	+115.12%
TOSHIBA MQ01ABF050	0.2552959	+169.23%
ST4000DM000	0.5447604	+225.28%
Manufacturer	MADM	Difference in Accuracy
Toshiba	0.09675122	-42.50%
Hitachi	0.4077924	+370.65%
Western Digital	0.289493	+107.75%
Seagate	0.3539884	+91.03%

Own depiction.

Tab. 15 Noise Level and Improvement in Accuracy

H.c Results: Accuracy

Model	Test Sample Size	Random Forest RMSE	Cox regression RMSE	Difference
WDC WD800AAJS	2	5.09	12.80	+151.47%
HGST HMS5C4040ALE640	5	126.03	827.14	+556.30%
HGST HMS5C4040BLE640	12	123.39	620.42	+402.81%
Hitachi HDS5C3030ALA630	7	137.55	328.11	+138.54%
Hitachi HDS5C4040ALE630	3	260.91	209.67	-19.64%
Hitachi HDS722020ALA330	6	43.10	195.685	+354.03%
Hitachi HDS723030ALA640	4	38.19	107.46	+181.38%
ST320LT007	12	27.06	75.61	+179.42%
ST500LM012 HN	6	97.09	314.74	+224.17%
ST500LM030	2	8.94	45.59	+409.96%
ST4000DM001	1	38.73	67.86	+75,21%
ST4000DX000	6	101.52	232.69	+129.21%
ST6000DX000	4	130.91	613.84	+368.90%
ST8000DM002	32	145.30	310.66	+113.81%
ST8000NM0055	33	132.79	228.19	+71.84%
ST10000NM0086	1	38.17	32.59	-14.62%
ST12000NM0007	64	67.46	88.98	+31.90%
ST3160318AS	1	39.81	213.00	+435.04%
WDC WD60EFRX	2	125.06	269.03	+115.12%
TOSHIBA MQ01ABF050	7	76.07	204.80	+169.23%
ST4000DM000	373	216.07	702.83	+225.28%
Model	Test HDDs	Random Forest RMSE	Cox regression RMSE	Difference
Toshiba	8	161.38	92.79	-42.50%
Hitachi	40	145.29	683.81	+370.65%
Western Digital	30	267.11	554.92	+107.75%
Seagate	777	215.66	411.98	+91.03%

Own depiction.

Tab. 16 Results: Accuracy

H.d Results: Robustness

Model	Sensitivity RSF	Sensitivity PrHM	Difference
WDC WD800AAJS	1.39	5.49	294.96%
HGST HMS5C4040ALE640	10.40	587.73	5554.40%
HGST HMS5C4040BLE640	9.76	162.24	1561.44%
Hitachi HDS5C3030ALA630	10.59	42.71	303.47%
Hitachi HDS5C4040ALE630	5.70	165.74	2806.43%
Hitachi HDS722020ALA330	10.63	73.53	591.51%
Hitachi HDS723030ALA640	3.74	80.37	2051.51%
ST320LT007	11.16	71.56	541.40%
ST500LM012 HN	NaN	NaN	NaN
ST500LM030	150.33	3.01	-97.99%
ST4000DM001	NaN	NaN	NaN
ST4000DX000	5.66	331.79	5758.17%
ST6000DX000	5.03	353.28	6924.34%
ST8000DM002	1.25e+12	392.33	-100.00%
ST8000NM0055	6291.89	151.69	-97.59%
ST10000NM0086	1021.50	0.71	-99.93%
ST12000NM0007	1944.12	111.90	-94.24%
ST3160318AS	77.48	0.00	-100.00%
WDC WD30EFRX	7.32	96.16	1213.17%
TOSHIBA MQ01ABF050	7.84	267.09	3305.45%
ST4000DM000	5.35	163.91	2961.47%
Manufacturer	Sensitivity RSF	Sensitivity PrHM	Difference
Toshiba	3.25	344.97	10526.15%
Hitachi	6.26	178.52	2749.57%
Western Digital	3.71	231.85	6141.21%
Seagate	5.68	289.45	4991.74%

Own depiction.

Tab. 17 Results: Robustness

H.e Results: Time Complexity

Model	Training Time RSF (s)	Training Time PrHM (s)	Difference
WDC WD800AAJS	1.33	1.35	1.50%
HGST HMS5C4040ALE640	3.59	3.33	-7.24%
HGST HMS5C4040BLE640	4.56	3.76	-17.54%
Hitachi HDS5C3030ALA630	7.07	4.88	-30.98%
Hitachi HDS5C4040ALE630	3.08	1.84	-40.26%
Hitachi HDS722020ALA330	2.43	1.46	-39.92%
Hitachi HDS723030ALA640	3.2	2.69	-15.94%
ST320LT007	5.25	3.95	-24.76%
ST500LM012 HN	1.1	1.7	54.55%
ST500LM030	0.31	1.16	274.19%
ST4000DM001	0.33	0.36	9.09%
ST4000DX000	6.94	6.18	-10.95%
ST6000DX000	3.26	3.66	12.27%
ST8000DM002	16.02	23.86	48.94%
ST8000NM0055	15.54	25.56	64.48%
ST10000NM0086	1.25	1.55	24.00%
ST12000NM0007	15.45	20.93	35.47%
ST3160318AS	1.02	2.44	139.22%
WDC WD30EFRX	0.34	1.79	426.47%
TOSHIBA MQ01ABF050	2	2.07	3.50%
ST4000DM000	229.97	4,841.4	2005.23%
Manufacturer	Training Time RSF (s)	Training Time PrHM (s)	
Toshiba	2.33	1.64	-29.61%
Hitachi	24.38	18.92	-22.40%
Western Digital	48.86	150.77	208.58%
Seagate	1,549.68	39,465.25	2446.67%

Own depiction.

Tab. 18 Results: Time Complexity

H.f Results: Space Complexity

Model	Size RSF (KB)	Size PrHM (KB)	Difference
WDC WD800AAJS	2,768	274	-90.10%
HGST HMS5C4040ALE640	3,907	873	-77.66%
HGST HMS5C4040BLE640	8,451	1,021	-87.92%
Hitachi HDS5C3030ALA630	14,636	1,417	-90.32%
Hitachi HDS5C4040ALE630	4,881	691	-85.84%
Hitachi HDS722020ALA330	5,244	548	-89.55%
Hitachi HDS723030ALA640	5,269	699	-86.73%
ST320LT007	17,915	1,085	-93.94%
ST500LM012 HN	2,805	312	-88.88%
ST500LM030	1,958	192	-90.19%
ST4000DM001	1,564	211	-86.51%
ST4000DX000	14,511	1,469	-89.88%
ST6000DX000	7,784	841	-89.20%
ST8000DM002	71,444	3,046	-95.74%
ST8000NM0055	70,248	2,979	-95.76%
ST10000NM0086	3,518	386	-89.03%
ST12000NM0007	64,294	2,913	-95.47%
ST3160318AS	2,702	311	-88.49%
WDC WD30EFRX	647	131	-79.75%
TOSHIBA MQ01ABF050	4,672	474	-89.85%
ST4000DM000	828,219	40,591	-95.10%
Manufacturer	Size RSF (KB)	Size PrHM (KB)	Difference
Toshiba	5,753	554	-90.37%
Hitachi	84,581	4,554	-94.62%
Western Digital	178,360	9,127	-94.88%
Seagate	1,014,008	63,889	-93.70%

Own depiction.

Tab. 19 Results: Space Complexity

Declaration of Authorship

I hereby declare that, to the best of my knowledge and belief, this Master Thesis titled “A Decision Model for the Selection of Fleet-Based Prognostic Methods” is my own work. I confirm that each significant contribution to and quotation in this thesis that originates from the work or works of others is indicated by proper use of citation and references.

Münster, 26 June 2019

Kevin Martin Wesendrup

Consent Form

for the use of plagiarism detection software to check my thesis

Name: Wesendrup **Given Name:** Kevin Martin

Student number: 447082 **Course of Study:** Information Systems

Address: Franz-Mülder-Straße 10, 48282 Emsdetten

Title of the thesis: A Decision Model for the Selection of Fleet-Based Prognostic Methods

What is plagiarism? Plagiarism is defined as submitting someone else's work or ideas as your own without a complete indication of the source. It is hereby irrelevant whether the work of others is copied word by word without acknowledgment of the source, text structures (e.g. line of argumentation or outline) are borrowed or texts are translated from a foreign language.

Use of plagiarism detection software. The examination office uses plagiarism software to check each submitted bachelor and master thesis for plagiarism. For that purpose the thesis is electronically forwarded to a software service provider where the software checks for potential matches between the submitted work and work from other sources. For future comparisons with other theses, your thesis will be permanently stored in a database. Only the School of Business and Economics of the University of Münster is allowed to access your stored thesis. The student agrees that his or her thesis may be stored and reproduced only for the purpose of plagiarism assessment. The first examiner of the thesis will be advised on the outcome of the plagiarism assessment.

Sanctions. Each case of plagiarism constitutes an attempt to deceive in terms of the examination regulations and will lead to the thesis being graded as "failed". This will be communicated to the examination office where your case will be documented. In the event of a serious case of deception the examinee can be generally excluded from any further examination. This can lead to the exmatriculation of the student. Even after completion of the examination procedure and graduation from university, plagiarism can result in a withdrawal of the awarded academic degree.

I confirm that I have read and understood the information in this document. I agree to the outlined procedure for plagiarism assessment and potential sanctioning.

Münster, 26 June 2019

Kevin Martin Wesendrup