Predicting the Need for Vehicle Compressor Repairs Using Maintenance Records and Logged Vehicle Data

Rune Prytz[a], Sławomir Nowaczyk[b], Thorsteinn Rögnvaldsson[b], Stefan Byttner[b]

[a] *Volvo Group Trucks Technology, Advanced Technology & Research, Göteborg, Sweden.*
[b] *Center for Applied Intelligent Systems Research, Halmstad University, Sweden.*

Abstract

Methods and results are presented for applying supervised machine learning techniques to the task of predicting the need for repairs of air compressors in commercial trucks and buses. Prediction models are derived from logged on-board data that are downloaded during workshop visits and have been collected over three years on large number of vehicles. A number of issues are identified with the data sources, many of which originate from the fact that the data sources were not designed for data mining. Nevertheless, exploiting this available data is very important for the automotive industry as means to quickly introduce predictive maintenance solutions. It is shown on a large data set from heavy duty trucks in normal operation how this can be done and generate a profit.

Random forest is used as the classifier algorithm, together with two methods for feature selection whose results are compared to a human expert. The machine learning based features outperform the human expert features, wich supports the idea to use data mining to improve maintenance operations in this domain.

Keywords: Machine Learning, Diagnostics, Fault Detection, Automotive Industry, Air Compressor

1. Introduction

Today, Original Equipment Manufacturers (OEMs) of commercial transport vehicles typically design maintenance plans based on simple parameters such as calendar time or mileage. However, this is no longer sufficient in the market and there is a need for more advanced approaches that provide predictions of future maintenance needs of individual trucks. Instead of selling just vehicles, the sector is heading towards selling complete transport services; for example, a fleet of trucks, including maintenance, with a guaranteed level of availability. This moves some of the operational risk from the customer to the OEM but should lower the overall cost of ownership. The OEM

Email addresses: rune.prytz@volvo.com (Rune Prytz), slawomir.nowaczyk@hh.se (Sławomir Nowaczyk), thorsteinn.rognvaldsson@hh.se (Thorsteinn Rögnvaldsson), stefan.byttner@hh.se (Stefan Byttner)

has the benefit of scale and can exploit similarities in usage and wear between different vehicle operators.

Predicting future maintenance needs of equipment can be approached in many different ways. One approach is to monitor the equipment and detect patterns that signal an emerging fault, which is reviewed by Hines and Seibert (2006), Hines et al. (2008a), Hines et al. (2008b), and Ma and Jiang (2011). A more challenging one is to predict the Remaining Useful Life (RUL) for key systems, which is reviewed by Peng et al. (2010), Si et al. (2011), Sikorska et al. (2011) and Liao and Köttig (2014). For each of these approaches there are several options on how to do it: use physical models, expert rules, data-driven models, or hybrid combinations of these. The models can look for parameter changes that are linked to actual degradation of components, or they can look at vehicle usage patterns and indirectly infer the wear on the components. Data-driven solutions can be based on real-time data streamed during operation or collected historical data.

We present a data-driven approach that combines pattern recognition with the RUL estimation, by classifying if the RUL is shorter or longer than the time to the next planned service visit. The model is based on combining collected (i.e. not real-time) data from two sources: data collected on-board the vehicles and service records collected from OEM certified maintenance workshops. This presents a number of challenges, since the data sources have been designed for purposes such as warranty analysis, variant handling and financial follow-up on workshops, not for data mining. The data come from a huge set of real vehicles in normal operation, with different operators. The challenges include, among others, highly unbalanced datasets, noisy class labels, uncertainty in the dates, irregular readouts and unpredictable number of readouts from individual vehicles. In addition, multiple readouts from the same truck are highly correlated, which puts constraints on how data for testing and training are selected. We specifically study air compressors on heavy duty trucks and the fault complexity is also a challenge; air compressors face many possible types of failures, but we need to consider them all as one since they are not differentiated in the data sources.

Predictive maintenance in the automotive domain is more challenging than in many other domains, since vehicles are moving machines, often operating in areas with low network coverage or travelling between countries. This means few opportunities for continuous monitoring, due to the cost of wireless communication, bandwidth limitations, etc. In addition, both the sensors and computational units need to fulfil rigorous safety standards, which makes them expensive and not worth adding purely for diagnostic purposes. Those problems are amplified due to a large variety of available truck configurations. Finally, heavy duty vehicles usually operate in diverse and often harsh environments.

The paper is structured as follows. A survey of related works introduces the area of data mining of warranty data. This is followed by an overview of the data sets and then a methodology section where the problem is introduced and the employed methods are described. This is finally followed by a results section and a conclusion section.

Related Work

There are few publications where service records and logged data are used for predicting maintenance needs of equipment, especially in the automotive industry, where

wear prediction is almost universally done using models that are constructed before production.

In a survey of artificial intelligence solutions in the automotive industry, Gusikhin et al. (2007) discuss fault prognostics, after-sales service and warranty claims. Two representative examples of work in this area are Buddhakulsomsiri and Zakarian (2009) and Rajpathak (2013). Buddhakulsomsiri and Zakarian (2009) present a data mining algorithm that extracts associative and sequential patterns from a large automotive warranty database, capturing relationships among occurrences of warranty claims over time. Employing a simple IF-THEN rule representation, the algorithm filters out insignificant patterns using a number of rule strength parameters. In their work, however, no information about vehicle usage is available, and the discovered knowledge is of a statistical nature concerning relations between common faults. Rajpathak (2013) presents an ontology based text mining system that clusters repairs with the purpose of identifying best-practice repairs and, perhaps more importantly, automatically identifying when claimed labour codes are inconsistent with the repairs. Related to the latter, but more advanced, is the work by Medina-Oliva et al. (2014) on ship equipment diagnosis. They use an ontology approach applied to mining fleet data bases and convincingly show how to use this to find the causes for observed sensor deviations.

Thus, data mining of maintenance data and logged data has mainly focused on finding relations between repairs and operations and to extract most likely root causes for faults. Few have used them for estimating RUL or to warn for upcoming faults. We presented preliminary results for the work in this paper in an earlier study (Prytz et al., 2013). Furthermore, Frisk et al. (2014) recently published a study where logged onboard vehicle data were used to model RUL for lead-acid batteries. Their approach is similar to ours in the way that they also use random forests and estimate the likelihood that the component survives a certain time after the last data download. Our work is different from theirs in two aspects. First, a compressor failure is more intricate than a battery failure; a compressor can fail in many ways and there are many possible causes. Secondly, they also attempt to model the full RUL curve whereas we only consider the probability for survival until the next service stop.

Recently Choudhary et al. (2009) presented a survey of 150 papers related to the use of data mining in manufacturing. While their scope was broader than only diagnostics and fault prediction, they covered a large portion of literature related to the topic of this paper. Their general conclusion is that the specifics of the automotive domain make fault prediction and condition based maintenance a more challenging problem than in other domains; almost all research considers the case where continuous monitoring of devices is possible.

Jardine et al. (2006) present an overview of condition-based maintenance (CBM) solutions for mechanical systems, with special focus on models, algorithms and technologies for data processing and maintenance decision-making. They emphasize the need for correct, accurate, information (especially event informaton) and working tools for extracting knowledge from maintenance databases. Peng et al. (2010) also review methods for prognostics in CBM and conclude that methods tend to require extensive historical records that include many failures, even "catastrophic" failures that destroy the equipment, and that few methods have been demonstrated in practical applications. Schwabacher (2005) surveys recent work in data-driven prognostics, fault detection

and diagnostics. Si et al. (2011) and Sikorska et al. (2011) present overviews of methods for prognostic modelling of RUL and note that available on-board data are seldom tailored to the needs of making prognosis and that few case studies exist where algorithms are applied to real world problems in realistic operating environments.

When it comes to diagnostics specifically for compressors, it is common to use sensors that continuously monitor the health state, e.g. accelerometers for vibration statistics, see Ahmed et al. (2012), or temperature sensors to measure the compressor working temperature, see Jayanth (2010 (filed 2006). The standard off-board tests for checking the health status of compressors require first discharging the compressor and then measuring the time it takes to reach certain pressure limits in a charging test, as described e.g. in a compressor trouble shooting manual Bendix (2004). All these are essentially model-based diagnostic approaches where the normal performance of a compressor has been defined and then compared to the field case. Similarly, there are patents that describe methods for on-board fault detection for air brake systems (compressors, air dryers, wet tanks, etc.) that build on setting reference values at installment or after repair, see e.g. Fogelstrom (2007 (filed 2006).

In summary, there exist very few published examples where equipment maintenance needs are estimated from logged vehicle data and maintenance data bases. Yet, given how common these data sources are and how central transportation vehicles are to the society, we claim it is a very important research field.

2. Presentation of Data

Companies that produce high value products necessarily have well-defined processes for product quality follow-up, which usually rely on large quantities of data stored in databases. Although these databases were designed for other purposes, e.g. analysing warranty issues, variant handling and workshop follow-up, it is possible to use them also to model and predict component wear. In this work we use two such databases: the *Logged Vehicle Data* (LVD) and the *Volvo Service Records* (VSR). In this work we have used data from approximately 65000 European Volvo trucks, models FH13 and FM13, produced between 2010 and 2013.

LVD

The LVD database contains aggregated information about vehicle usage patterns. The values are downloaded each time a vehicle visits an OEM authorised workshop for service and repair. This happens several times per year, but at intervals that are irregular and difficult to predict *a priori*.

During operation, a vehicle continuously aggregates and stores a number of parameters, such as average speed or total fuel consumption. In general, those are simple statistics of various kinds, since there are very stringent limitations on memory and computing power, especially for older truck models. Most parameters belong to one of the following three categories: *Vehicle Performance and Utilisation*, *Diagnostics* or *Debugging*. This work mainly focused on the first category, which includes over two thousand different parameters. They are associated with various subsystems and components. The following four example ones have been identified as important for

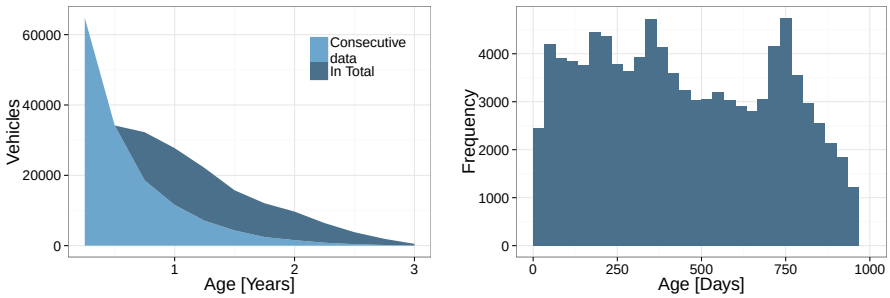

Figure 1: Vehicle age distribution, based on readouts in LVD. The left panel shows the number of vehicles, with data readouts, at different ages: dark blue are vehicles with any data; light blue are vehicles with consecutive data. The right panel shows the age of vehicles at data readouts for the subset of vehicles that have *any* readouts beyond age of two years (warranty period).

predicting air compressor failures by a domain expert: *Pumped air volume since last compressor change*, *Mean compressed air per distance*, *Air compressor duty cycle*, and *Vehicle distance*.

The vehicles in our data set visit a workshop, on average, every 15 weeks. This means that the predictive horizon for the prognostic algorithm must be at least that long. The system needs to provide warnings about components with an increased risk for failing until the next expected workshop visit. However, if the closest readout prior to the failure is 3-4 months, then it is less likely that the wear has had visible effects on the data.

This time sparseness is a considerable problem with the LVD. The readout frequency varies a lot between vehicles and changes with vehicle age, and can be as low as one readout per year. They also become less frequent as the vehicle ages. Figure 1 illustrates how many vehicles have data in LVD at different ages. Some vehicles (dark blue in the Figure) have consecutive data, defined as at least one readout every three months. They are likely to have all their maintenance and repairs done at OEM authorised workshops. Many vehicles, however, only have sporadic readouts (light blue in the Figure).

For data mining purposes are the vehicles with consecutive data most useful. They have readouts in the LVD database and repairs documented in the VSR system. They contribute with sequences of data that can be analysed for trends and patterns. On the other hand, from the business perspective it is important that as many trucks as possible are included in the analysis.

The right panel of Figure 1 illustrates two different maintenance strategies. The three peaks during the first year correspond to the typical times of scheduled maintenance. Repairs then get less frequent during the second year, with the exception of just before the end of it. This is probably the result of vehicles getting maintenance and repairs before the warranty period ends. In general, all vehicles visit the OEM authorised workshops often during the warranty period. After that, however, some vehicles disappear, while the remaining ones continue to be maintained as before, without any significant drop in visit frequency. This loss of data with time is a problematic issue.

Plenty of valuable LVD data is never collected, even after the first year of vehicle operation. A future predictive maintenance solution must address this, either by collecting the logged data and the service information using telematics or by creating incentives for independent workshops to provide data.

Finally, the specification of parameters that are monitored varies from vehicle to vehicle. A core set of parameters, covering basic things like mileage, engine hours or fuel consumption, is available for all vehicles. Beyond that, however, the newer the vehicle is, the more LVD parameters are available, but it also depends on vehicle configuration. For instance, detailed gearbox parameters are only available for vehicles with automatic gearboxes. This makes it hard to get a consistent dataset across a large fleet of vehicles and complicates the analysis. One must either select a dataset with inconsistencies and deal with missing values, or limit the analysis to only vehicles that have the desired parameters. In this work we follow the latter approach and only consider parameter sets that are present across large enough vehicle fleets. Sometimes this means that we need to exclude individual parameters that most likely would have been useful.

VSR

The VSR database contains repair information collected from the OEM authorised workshops around the world. Each truck visit is recorded in a structured entry, labelled with date and mileage, detailing the parts exchanged and operations performed. Parts and operations are denoted with standardised identification codes.

The database contains relevant and interesting information with regards to vehicle failures, information that is sometimes exploited by workshop technicians for diagnostics and to predict future failures. However, the work presented here is only using the dates of historic repairs. Those dates form the supervisory signal for training and validating the classifier, i.e. to label individual LVD readouts as having either faulty or healthy air compressor, based on the time distance to the nearest replacement.

Unfortunately, however, there are no codes for reasons *why* operations are done. In some cases those can be deduced from the free text comments from the workshop personnel, but not always. The quality and level of detail of those comments vary greatly. This is a serious limitation since it introduces a lot of noise into the training data classification labels. In the worst case can a perfectly good part be replaced in the process of diagnosing an unrelated problem.

Undocumented repairs are also a problem. They rarely happen at authorised workshops since the VSR database is tightly coupled with the invoicing systems. On the other hand, there is seldom any information about repairs done in other workshops. Patterns preceding faults that suddenly disappear are an issue, both when training the classifier and later when evaluating it.

Much of the information in the VSR database is entered manually. This results in various human errors such as typos and missing values. A deeper problem, however, are incorrect dates and mileages, where information in the VSR database can be several weeks away from when the matching LVD data was read out. This is partly due to lack of understanding by workshop technicians; for the main purposes, invoicing and component failure statistics, exact dates are not overly important. In addition, the VSR

date is poorly defined. In some cases the date can be thought of as the date of diagnosis, i.e. when the problem was discovered, and it may not be the same as the repair date, i.e. the date when the compressor was replaced.

3. Problem Formulation

Much diagnostics research focuses on predicting Remaining Useful Life (RUL) of a component, and, based on that, deciding when to perform maintenance or component replacement.

The RUL is usually modelled as a random variable that depends on the age of the component, the environment in which it operates, and the partially observable health state, which is continuously monitored or occasionally measured. Using the same notation as Si et al. (2011) we define X_t as the random variable of the RUL at time t, and Y_t is the history of operational profiles and condition monitoring information up to that point. The probability density function of X_t conditional on Y_t is denoted as $f(x_t|Y_t)$.

The usual RUL approach is to estimate $f(x_t|Y_t)$ or the expectation of the RUL. However, the setting we consider in this paper, this approach is unnecessarily complicated; we do not need to calculate the perfect time to perform repair since it is unpractical to do a repair at arbitrary times. Repairs are preferably done during planned maintenance events. The truck is in workshop on a particular date and the decision is a binary one: either to replace the component now or not. It should be replaced if the risk that it will not survive until the next planned maintenance is sufficiently high (in relation to the cost for repairing it in an unplanned service). That is, we need to estimate the posterior probability

$$P(X_t < \Delta|Y_t) = \int_0^{\Delta} f(x_t|Y_t)dx_t \tag{1}$$

where Δ is the time horizon. This is the probability that the RUL does not exceed the time horizon Δ, conditioned on the operation history and maintenance information Y_t. Based on this probability, we can make a decision of whether to flag individual component as *faulty* or *healthy*, during every workshop visit of a given vehicle.

In the following sections we make two simplifying assumptions. First, we use the same prediction horizon Δ for all vehicles. Even though some vehicles are in the workshop more often than others, the main driving factor is the cost of wasting component life and the acceptable level can be defined globally. The other simplification is that Y_t only contains the currently downloaded LVD data. This can be viewed as a form of Markovian condition, we have no memory of previous maintenance events in LVD, VSR or any other database. This latter assumption is for practical reasons; a system like this should be possible to implement in a workshop tool, without the need to access a historical database.

One simplification that we cannot assume is that of a single failure mode. We must determine whether the component (air compressor in this case) will fail or not, regardless of mode. It is impossible to get information on single failure modes from the maintenance records.

As described in the next section, we do not explicitly estimate the posterior probability $P(X_t < \Delta | Y_t)$, but instead use a supervised classifier to predict the *faulty/healthy* decision directly.

4. Methods

Machine learning algorithm and software

All experimental results are averages over 10 runs using the Random Forest (Breiman, 2001) classifier, with 10-fold cross validation. We used the R language (R Core Team, 2014) including `caret`, `unbalanced`, `DMwR` and `ggplot2` libraries[1].

Random Forest are decision trees combined by *bagging* but with an additional layer of randomness ontop of what is added by the boostraping of training data. The additional randomness is added by only considering subset of features at each node split. The considered features are randomly selected at each node and is normally few compared to the available features in the training data.

Evaluation criteria

Supervised machine learning algorithms are typically evaluated using measures like accuracy, area under the Receiver Operating Characteristic (ROC) curve or similar. Most of them, however, are suitable only for balanced datasets, i.e. ones with similar numbers of positive and negative examples. Measures that also work well for the unbalanced case include, e.g., the Positive Predictive Value (PPV) or F_1-score:

$$F_1 = \frac{2\text{TP}}{2\text{TP} + \text{FN} + \text{FP}} \qquad \text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{2}$$

where TP, FP and FN denote *true positives*, *false positives* and *false negatives*, respectively.

However, the prognostic performance must take business aspect into account, where the ultimate goal is to minimise costs and maximise revenue. In this perspective, there are three main components to consider: the initial investment cost, the financial gain from correct predictions, and the cost of false alarms.

The initial investment cost consists of designing and implementing the solution, as well as maintaining the necessary infrastructure. This is a fixed cost, independent of the performance of the method. It needs to be overcome by the profits from the maintenance predictions. In this paper we estimate it to be €150,000, which is approximately one year of full time work.

The financial gains come from correctly predicting failures before they happen and doing something about them. It is reported in a recent white paper by Reimer (2013) that *wrench time* (the repair time, from estimate approval to work completion), is on average about 16% of the time a truck spends at the workshop. Unexpected failures are one of the reasons for this since resources for repairs need to be allocated. All component replacements are associated with some *cost of repair*. However, unexpected

[1] `http://cran.r-project.org/web/packages/`

breakdowns usually cause additional issues, such as *cost of delay* associated with not delivering the cargo in time. In some cases, there are additional costs like *towing*. Fixed *operational costs* correspond to the cost of owning and operating a vehicle without using it. This includes drivers wage, insurances and maintenance. A European long-haul truck costs on average €1,000 per day in fixed costs.

False alarms are the most significant cost, since when good components are flagged by the system as being faulty, an action needs to be taken. At best this results in additional work for workshop personnel, and at worst it leads to unnecessary component replacements.

It is worth noting that the above analysis does not account for *false negatives*, i.e. for cases where actual component failures were not detected. This is somewhat counter-intuitive, in a sense that one can think of them as missed opportunities, and missing opportunities is bad. In the current analysis, however, we focus on evaluating the feasibility of *introducing* a predictive maintenance solution in a market where there is none.

At this stage, our goal is much less finding the best possible method, but rather on presenting a convincing argument that a predictive maintenance solution can improve upon existing situation. In comparison to the current maintenance scheme, where the vehicles run until failure, those false negatives maintain status quo.

This is of course a simplification, since there is certain cost associated with missing a failure. We could elaborate about the value of customer loyalty and quality reputation, but they are very hard to quantify. Therefore, in the results section, we use both the above-defined profit, as well as more "traditional" evaluation metrics (accuracy and F_1-score), and point out some differences between them.

In this respect is the predictive maintenance domain for the automotive industry quite different from many others. For instance, in the medical domain, *false negatives* correspond to patients who are not correctly diagnosed even though they carry the disease in question. This can have fatal consequences and be more costly than *false positives*, where patients get mistakenly diagnosed. It is also similar for the aircraft industry.

Among others, Sokolova et al. (2006) analyse a number of evaluation measures for assessing different characteristics of machine learning algorithms, while Saxena et al. (2008) specifically focus on validation of predictions. They note how lack of appropriate evaluation methods often renders prognostics meaningless in practice.

Ultimately, the criterion for evaluation of the model performance is a cost function based on the three components introduced at the beginning of this section. The function below captures the total cost of the implementation of the predictive maintenance system:

$$\text{Profit} = \text{TP} \times \text{ECUR} - \text{FP} \times \text{CPR} - \text{Investment}, \tag{3}$$

where ECUR stands for *extra cost of unplanned repair* and CPR stands for *cost of planned repair*. Each *true positive* avoids the additional costs of a breakdown and each *false positive* is a repair done in vain, which causes additional costs.

It is interesting to study the ratio between the cost of planned and unplanned repairs. It will vary depending on the component, fleet operator domain and business model, et cetera. On the other hand, the cost for a breakdown for vehicles with and without pre-

dictive maintenance can be used to determine the "break even" ratio required between *true positives* and *false positives*.

Prediction Horizon

We define the Prediction Horizon (PH) as the period of interest for the predictive algorithm. A replacement recommendation should be made for a vehicle for which the air compressor is expected to fail within that time frame into the future. As described earlier, the vehicles visit the workshop on average every 15 weeks and the PH needs to be at least that long. The system should provide warnings about components that are at risk of failing before the next expected workshop visit.

It is expected that the shorter the PH, the more likely it is that there is information in the data about upcoming faults. It is generally more difficult to make predictions the further into the future they extend, which calls for a short PH. However, from a business perspective it is desireable to have a good margin for planning, which calls for a long PH. We experiment with setting the PH up to a maximum of 50 weeks.

Independent data sets for training and testing

A central assumption in machine learning (and statistics) it that of *independent and identically distributed (IID)* data. There are methods that try to lift it to various degrees, and it is well known that most common algorithms work quite well also in cases when this assumption is not fully fulfilled, but it is still important, especially when evaluating and comparing different solutions.

The readouts consist of aggregated data that have been sampled at different times. Subsequent values from any given truck are highly correlated to each other. It is even more profound in case of cumulative values, such as total mileage, a single event of abnormal value will directly affect all subsequent readouts. Even without the aggregation effect, however, there are individual patterns that are specific to each truck, be it particular usage or individual idiosyncrasies of the complex cyber-physical system. It makes all readouts from a single vehicle dependent. This underlying pattern of the data is hard to visualize by analysing the parameter data as such. However, a classifier can learn these patterns and overfit.

A partial way of dealing with the problem is to ensure that the test and train dataset be split on a *per vehicle* basis and not randomly among all readouts. It means that if one or more readouts from a given vehicle belong to the test set, no readouts from the same vehicle can be used to train the classifier. The data sets for training and testing must contain unique, non-overlapping, sets of vehicles in order to guarantee that patterns that are linked to wear and usage are learned, instead of specific usage patterns for individual vehicles.

Feature selection

The data set contains 1,250 unique features and equally many differentiated features. However, only approximately 500 of them are available for the average vehicle. It is clear that not all features should be used as input to the classifier. It is important to find the subset of features that yields the highest classification performance. Additionally, the small overlap of common features between vehicles makes this a research

challenge. It is hard to select large sets of vehicles that each share a common set of parameters. Every new feature that gets added to the dataset must be evaluated with respect to the gain in performance and the decrease in number of examples.

Feature selection is an active area of research, but our setting poses some specific challenges. Guyon and Elisseeff (2003) and Guyon et al. (2006) present a comprehensive and excellent, even if by now somewhat dated, overview of the feature selection concepts. Bolón-Canedo et al. (2013) present a more recent overview of methods. Molina et al. (2002) analyse performance of several fundamental algorithms found in the literature in a controlled scenario. A scoring measure ranks the algorithms by taking into account the amount of relevance, irrelevance and redundance on sample data sets. Saeys et al. (2007) provide a basic taxonomy of feature selection techniques, and discuss their use, variety and potential, from the bioinformatics perspective, but many of the issues they discuss are applicable to the data analysed in this paper.

We use two feature selection methods: a wrapper approach based on the beam search algorithm, as well as a new filter method based on the Kolmogorov-Smirnov test to search for the optimal feature set. The final feature sets are compared against an expert dataset, defined by an engineer with domain knowledge. The expert dataset contains four features, all of which have direct relevance to the age of the vehicle or the usage of the air compressor.

The beam search feature selection algorithm performs a greedy graph search over the powerset of all the features, looking for the subset that maximises the classification accuracy. However, at each iteration, we only expand nodes that maintain the data set size above the given threshold. The threshold is reduced with the number of parameters as shown in equation (4). Each new parameter is allowed to reduce the dataset with a small fraction. This ensures a lower bound on the data set size. The top five nodes, with respect to accuracy, are stored for next iteration. This increased the likelihood of finding the global optimum. The search is stopped when a fixed number of features is found:

$$n_{dataset} = n_{all} \times constraintFactor^{n_{params}}. \tag{4}$$

Many parameters in the LVD dataset are highly correlated and contain essentially the same information, which can potentially lower the efficiency of the beam search. Chances are that different beams may select different, but correlated, feature sets. In this way the requirement for diversity on the "syntactic" level is met, while the algorithm is still captured in the same local maximum. We have not found this to be a significant problem in practice.

With the Kolmogorov-Smirnov method, we are interested in features whose distributions vary in relation to oncoming failures of air compressors. Based on the expertise within OEM company we know that there are two main reasons for such variations. On the one hand, they may be related to different *usage patterns* of the vehicle. As an example, air compressors on long-haul trucks typically survive longer than on delivery trucks, due to factors like less abrupt brakes usage and less often gear changes. On the other hand, early symptoms of *component wear* may also be visible in some of the monitored parameters. For example, as worn compressors are weaker than new ones, it often takes them longer time to reach required air pressure.

To identify these features, we define *normal* and *fault* data sets, and compare their

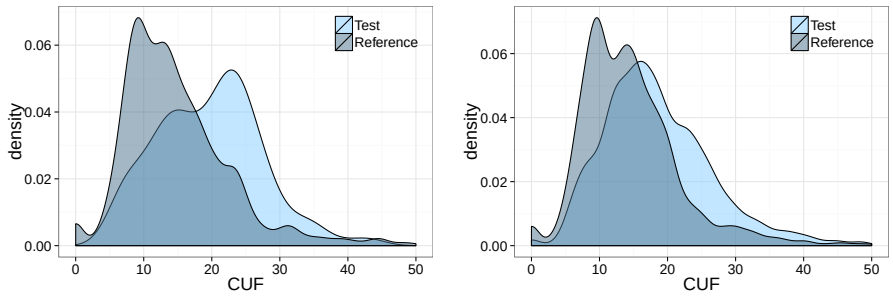

Figure 2: Differences due to usage. The left panel shows the normal and fault distributions for the feature *Compressor Duty Cycle* (CUF) when the fault sample (light blue) is drawn from periods 0–5 weeks prior to the compressor repair. The right panel shows the same thing but when the fault sample (light blue) is drawn from 0–25 weeks prior to the repair. The normal sample (grey) is in both cases selected from vehicles that have not had any air compressor repair.

distributions using the Kolmogorov-Smirnov (KS) test (Hazewinkel, 2001).

The *normal* sample is a random sample of fault-free LVD readouts, while the *fault* sample are LVD readouts related to a compressor repair. The fault sample is drawn from vehicles with a compressor change and selected from times up to PH before the repair. This is done in the same way for both *usage* and *wear* metrics. The normal sample, on the other hand, is drawn either from all vehicles that have not had a compressor change, or from vehicles with a compressor change but outside of the PH time window before the repair. In the first case, the difference in distributions between *normal* and *fault* data corresponds to parameters capturing *usage* difference that is relevant for air compressor failures. In the second case, it is the *wear* difference.

The two samples are compared using a two-sample KS test and a p-value is computed under the null hypothesis that the two samples are drawn from the same distribution. The p-value is a quantification of how likely it is to get the observed difference if the null hypothesis is true and a low p-value indicates that the null hypothesis may not be true. Features with low p-values are therefore considered interesting since the observed difference may indicate a fundamental underlying effect (wear or usage). The lower the p-value, the more interesting the feature. The KS filter search is terminated when a predetermined number of features has been reached.

Figure 2 illustrates the case with the feature Compressor Duty Cycle (CUF) when evaluated as relevant from the usage point of view. There is a clear difference 0–5 weeks before the repair and there is also a difference 0–25 weeks before the repair. Figure 3 illustrates the same case but when evaluated from the wear point of view. This is done for the illustrative purposes; CUF is a parameter that shows both usage and wear metrics, but there are other interesting parameters that are only discovered in one or the other.

It is important to note that the distribution of vehicle ages in the fault set is different from the normal set. We are working with real data, and old vehicles are more likely to fail than newer ones. This difference is clearly important when doing the classification, since the RUL depends on the age of the truck. However, in the feature selection, this

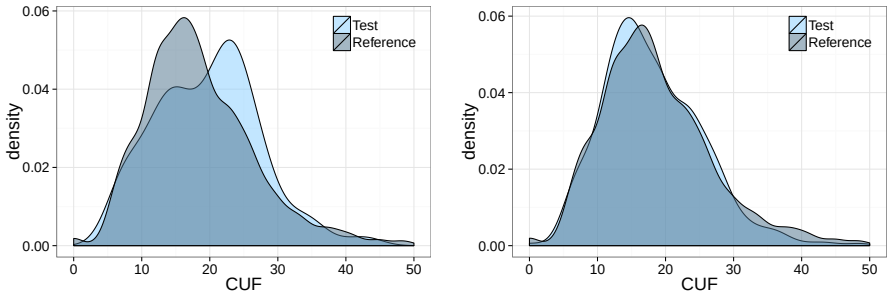

Figure 3: Differences due to wear. The left panel shows the normal and fault distributions for the feature *Compressor Duty Cycle* (CUF) when the fault sample (light blue) is drawn from periods 0–5 weeks prior to the compressor repair. The right panel shows the same thing but when the fault sample (light blue) is drawn from 0–25 weeks prior to the repair. The normal sample (grey) is in both cases selected from vehicles that have had an air compressor repair, but times that are before the PH fault data.

effect is undesirable: we are interested in identifying parameters that differ between healthy and failing vehicles, not those that differ between new and old vehicles.

We propose a methods for reducing the risk for such spurious effects, by re-sampling the normal group so that it has the same mileage or engine hour distribution as the fault group. We call this *age normalisation*, and present evaluation of it in the Results section. The sampling is done in two steps. The first step is to re-sample the reference distribution uniformly. In the second step is the uniform reference distribution sampled again, this time weighted according to the distribution of the test set. In cases with a narrow age distribution for the fault set will only a fraction of the normal data be used. This requires a substantial amount of normal data which, in our case, is possible since the dataset is highly unbalanced and there is much more normal data than fault data. The effect of age normalisation is illustrated in Fig. 4.

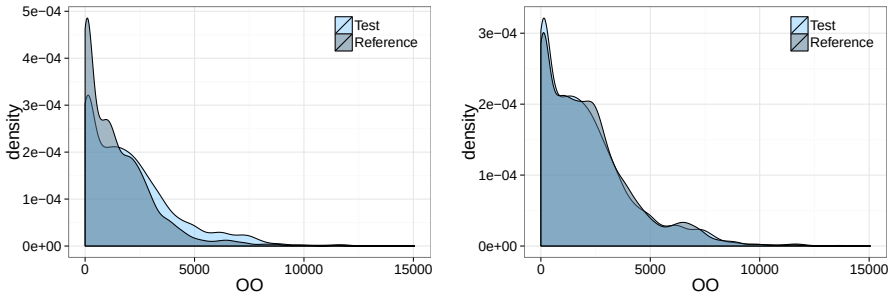

Figure 4: Illustration of the effect of age normalisation. The left panel shows the normal (grey) and the fault (light blue) distributions for a feature without age normalisation. The right panel shows the result after age normalisation. Here age normalisation removed the difference, which was a spurious effect caused by age.

Balancing the dataset

Machine learning methods usually assume a fairly balanced data distribution. If that is not fulfilled, then the results tend to be heavily biased towards the majority class. This is a substantial problem in our case, since only a small percentage of the vehicles experience compressor failure and, for any reasonable value of the PH, only a small subset of their readouts is classified as faulty.

Imbalanced datasets require either learning algorithms that handle this or data preprocessing steps that even out the imbalance. We chose to use the latter. There are many domains where class imbalance is an issue, and therefore a significant body of research is available concerning this. For example, He and Garcia (2009) provide a comprehensive review of the research concerning learning from imbalanced data. They provide a critical review of the nature of the problem and the state-of-the-art techniques. They also highlight the major opportunities and challenges, as well as potential important research directions for learning from imbalanced data. Van Hulse et al. (2007) present a comprehensive suite of experimentation on the subject of learning from imbalanced data. Sun et al. (2007) investigate meta-techniques applicable to most classifier learning algorithms, with the aim of advancing the classification of imbalanced data, exploring three cost-sensitive boosting algorithms, which are developed by introducing cost items into the learning framework of AdaBoost. Napierala and Stefanowski (2012) propose a comprehensive approach, called BRACID, that combines multiple different techniques for dealing with imbalanced data, and evaluate it experimentally on a number of well-known datasets.

We use the Synthetic Minority Over-sampling TEchnique (SMOTE), introduced by Chawla et al. (2002). It identifies, for any given positive example, the k nearest neighbours belonging to the same class. It then creates new, synthetic, examples randomly placed in between the original example and the k neighbours. It uses two design parameters: number of neighbours to take into consideration (k) and the percentage of synthetic examples to create. The first parameter, intuitively, determines how similar new examples should be to existing ones, and the other how balanced the data should be afterwards. SMOTE can be combined with several preprocessing techniques, e.g. introduced by Batista et al. (2004) and some others, aggregated and implemented in a R library by Dal Pozzolo et al.. We tried and evaluated four of them: The Edited Nearest Neighbour (ENN), the Neighbourhood Cleaning Rule (NCL), the Tomek Links (TL), and the Condensed Nearest Neighbour (CNN).

5. Results

Cost function

The *cost of planned repair* CPR, *cost of unplanned repair* CUR, and *extra cost of unplanned repair* ECUR can be split up into the following terms:

$$\text{CPR} = \text{C}_{part} + \text{C}^{P}_{work} + \text{C}^{P}_{downtime} \tag{5}$$

$$\text{CUR} = \text{C}_{part} + \text{C}^{U}_{work} + \text{C}^{U}_{downtime} + \text{C}_{extra} \tag{6}$$

$$\text{ECUR} = \text{CUR} - \text{CPR} \tag{7}$$

Here, C_{part} is the cost of the physical component, the air compressor, that needs to be exchanged. We set this to €1000. It is the same for both planned and unplanned repairs. C_{work} is the labour cost of replacing the air compressor, which takes approximately three hours. We set C^P_{work} to €500 for planned repairs and C^U_{work} to €1,000 for unplanned repairs. If the operation is unplanned, then one needs to account for diagnosis, disruptions to the workflow, extra planning, and so on.

$C_{downtime}$ is the cost for vehicle downtime. Planned component exchanges can be done together with regular maintenance; $C^P_{downtime}$ is therefore set to zero. It is included in equation (5) since it will become significant in the future, once predictive maintenance becomes common and multiple components can be repaired at the same time. The downtime is a crucial issue for unplanned failures, however, especially roadside breakdown scenarios. Commonly at least half a day is lost immediately, before the vehicle is transported to the workshop and diagnosed. After that comes waiting for spare parts. The actual repair may take place on the third day. The resulting 2–3 days of downtime plus a possible cost of towing, $C^U_{downtime}$, is estimated to cost a total of €3,500.

The additional costs, C_{extra}, are things like the delivery delay, the cost for damaged goods, fines for late arrival, and so on. This is hard to estimate, since it is highly dependent on the cargo, as well as on the vehicle operator's business model. The *just in time* principle is becoming more widespread in the logistics industry and the additional costs are therefore becoming larger. We set C_{extra} to €11,000.

Inserting those estimates into equations (5), (6) and (7) yields CPR = €1,500, CUR = €16,500 and ECUR = €15,000. The final Profit function, eq. (3), becomes (in Euros):

$$\text{Profit}(\text{TP}, \text{FP}) = \text{TP} \times 15,000 - \text{FP} \times 1,500 - 150,000. \tag{8}$$

Obviously, the Profit function (8) is an estimate and the numbers have been chosen so that there is a simple relationship between the gain you get from true positives and the loss you take from false positives (here the ratio is 10:1). A more exact ratio is hard to calculate since it is difficult to get access to the data required for estimating it (this type of information is usually considered confidential). Whether the predictive maintenance solution has a profit or loss depends much on the extra cost C_{extra} .

The importance of data independence

The importance of selecting independent data sets for training and testing cannot be overstated. Using dependent data sets will lead to overly optimistic results that never hold in the real application. Figure 5 shows the effects from selecting training and test data sets in three different ways.

The *random* method refers to when samples for training and testing are chosen completely randomly, i.e. when examples from the same vehicle can end up both in the training and the test data set. These data sets are not independent and the out-of-sample accuracy is consequently overestimated.

The *one sample* method refers to when each vehicle provides one positive and one negative example to the training and test data, and there is no overlap of vehicles in the training and test data. This leads to independent data sets that are too limited in size. The out-of-sample performance is correctly estimated but the data set cannot be made

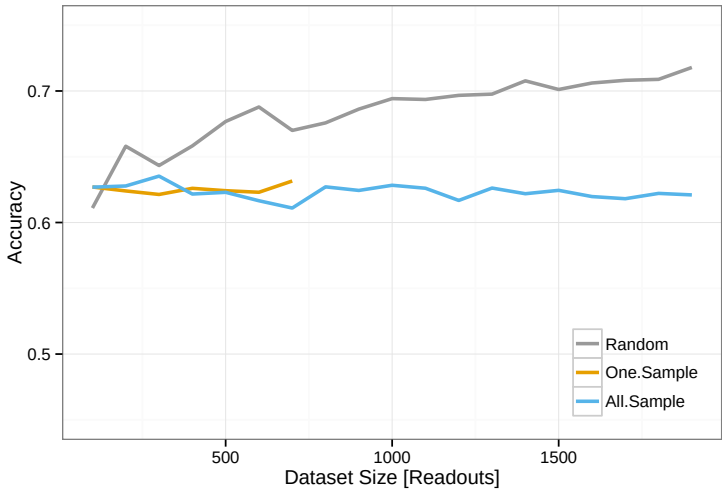

Figure 5: Comparison of strategies for selecting training and test data. The *Expert* feature set was used for all experiments and the data sets were balanced. The x-axis shows the size of the training data set.

large. The *all sample* method refers to the case when each vehicle can contribute with any number of examples but there is no overlap of vehicles between the training and test data. This also yields a correct out-of-sample accuracy but the training data set can be made larger.

Feature selection

The different feature selection approaches, and the age normalisation of the data, described in the Methods section produced six different feature sets in addition to the *Expert* feature set.

The beam search wrapper method was performed with five beams and a size reduction constraint of 10%. The search gave five different results, one from each beam, but four of them were almost identical, differing only by the last included feature. The four almost identical feature sets where therefore reduced to a single one, by including only the 14 common features. The fifth result was significantly different and was kept without any modifications. The two feature sets from the beam search are denoted *Beam search set 1* and *Beam search set 2*, respectively; they each had 14 features (the limit set for the method). Three out of the four features selected by the expert were also found by the beam search.

The KS filter method was used four times, with different combinations of *wear* features, *usage* features, and age normalisation (the reader is referred to the Methods section for details). This gave four feature sets: *Wear*, *Usage*, *Wear with age normalisation*, and *Usage with age normalisation*.

Figure 6 show the results when using the seven different feature sets. The overly optimistic result from using randomly selected data sets is shown for pedagogical reasons, reiterating the importance of selecting independent data sets. The data sets were

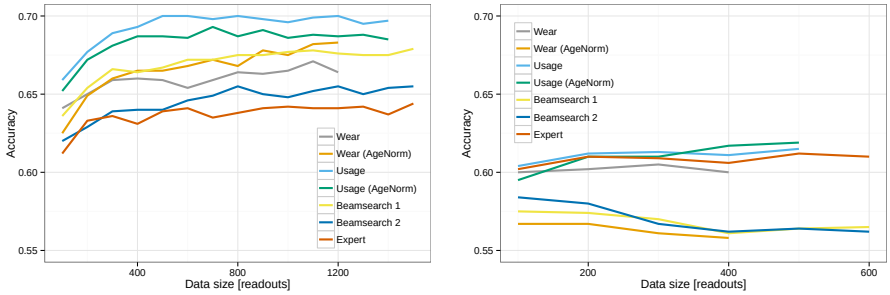

Figure 6: Comparison of feature selection methods when measuring the accuracy of the predictor. The left panel shows the result when training and test data are chosen randomly, i.e. with dependence. The right panel shows the result when the training and test data are chosen with the one sample method, i.e. without dependence.

balanced in the experiments. The *Usage* features performed best and the *Expert* features were second (except when an erroneous method for selecting data was used).

As an example, the following 14 parameters have been included in the *Beamseach 1* feature set:
BIX: Pumped air volume since last compressor change
CFZ: Timestamp at latest error activation
CHJ: Engine time at latest error activation (diff)
CUD: Max volume for air dryer cartridge
KJ: Fuel consumed in *Drive*
MT: Fuel consumed in *PTO*
OA: Total Distance in *PTO* (diff)
OF: Total time in Coasting
OL: Total time using pedal
OQ: Fuel consumed in *Econ* mode (diff)
OR: Fuel consumed in *Pedal* mode (diff)
NDI: Number of Times in Idle Mode (diff)
NDJ: Total Time in Idle Mode *Bumped* (diff)
NDP: Total Time in Idle Mode *Parked* (diff)
Where *(diff)* denote that the parameter has been diffrentiated and reflect the parametric change since the previous readout.

Accuracy vs. Prediction Horizon

Two experiments were done to gauge how the Prediction Horizon (PH) affects the classification results. In the first experiment were all available readouts used, while in the second experiment was the training set size fixed at 600 samples. Balanced data sets were used throughout why the number of available fault readouts were the limiting factor. As PH increased could more samples be used since more readouts were available for inclusion in the fault data set.

Figures 7 and 8 show the result of the two experiments. The accuracy (Fig. 7) is best at lower PH and decreases as the PH increases. This is probably due to the

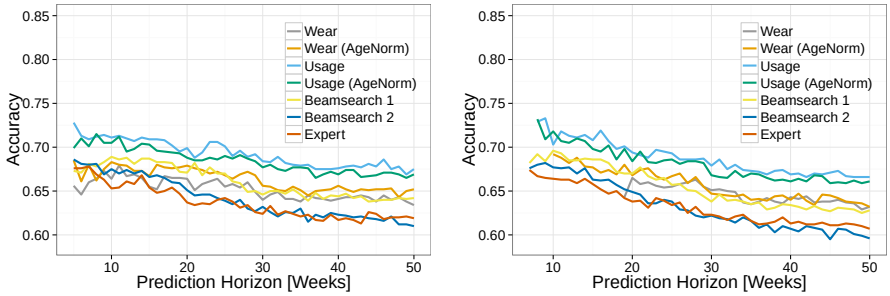

Figure 7: Prediction accuracy vs. prediction horizon. The left panel shows how the prediction accuracy decreases with PH when the training data set size is not limited. The right panel shows how the prediction accuracy decreases with PH when the training data set size is limited to 600 samples.

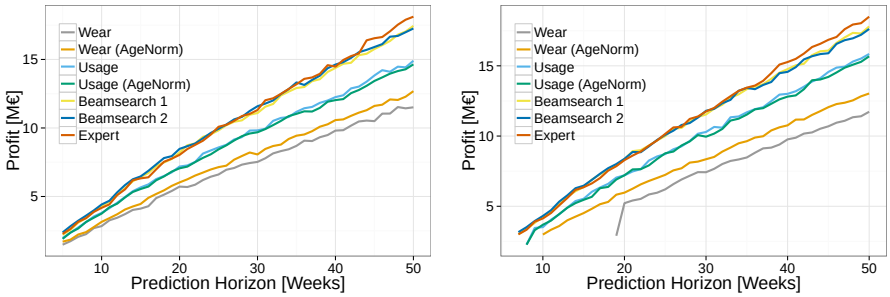

Figure 8: Profit vs. prediction horizon. The left panel shows how the profit increases with PH when the training data is not limited. The right panel shows how the profit increases with PH when the training data is limited to 600 samples. The PH required to achieve 600 samples varies between the datasets, which explains the differences in starting positions of individual lines.

training labels being more reliable closer to the fault. Accuracy decreases somewhat less rapidly in the first experiment with unlimited training data set size (left panel of Fig. 7). Figure 8 shows the result when evaluated with the profit measure. Interestingly, from this point of view, the system performance improves with larger PH. This appears to be, at least partially, caused by a larger number of *false negatives*. In particular, the further away data readout is from compressor replacement, the less indications of problems it contains. Thus a classifier will consider them to be negative examples, but if it was trained on data with sufficiently large prediction horizon, they will be *false negatives*. Large number of them will lower accuracy significantly, but will not affect profit.

SMOTE

The SMOTE oversampling method depends on two parameters: the percentage of synthetic examples to create and the number of neighbours to consider when generating new examples.

Figure 9 shows F_1-score (as defined in equation 2) and Profit (as defined in equation 8) when the SMOTE percentage is varied but k is kept constant at 20, for three different

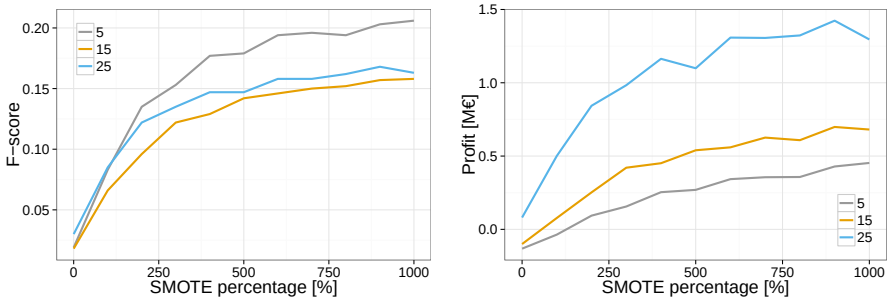

Figure 9: Evaluation of SMOTE percentage settings, using the *Expert* dataset. The number of SMOTE neighbours is fixed to 20.

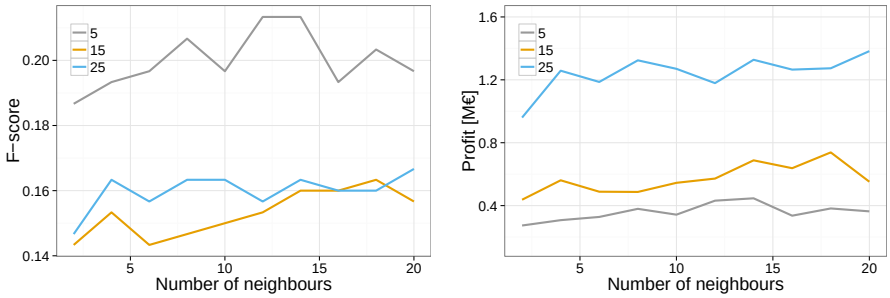

Figure 10: Evaluation of the number of SMOTE neighbours (k) using the *Expert* dataset and with the SMOTE% fixed at 900.

values of the PH. All results improve significantly with the percentage of synthetic examples, all the way up to a ten-fold oversampling of synthetic examples. A lower PH is better from the F_1-score perspective but worse from the Profit perspective. Figure 10 shows the effect of varying k, the number of SMOTE neighbours, when the SMOTE percentage is kept fixed at 900%. The results are not very sensitive to k although a weak increase in performance comes with higher k.

The four SMOTE preprocessing methods mentioned in section 4 were evaluated using a PH of 25 weeks and a SMOTE percentage of 900% (the best average settings found). Nearly all feature sets benefitted from preprocessing but there was no single best method.

Final evaluation

A final experiment was done, using the best settings found for each feature set, in order to evaluate the whole approach. The best SMOTE settings were determined by first keeping k fixed at 20 and finding the best SMOTE%. Then the SMOTE% was kept fixed at the best value and the k value varied between 1 and 20 and the value that produced the best cross-validation Profit was kept. The best SMOTE preprocessing determined in the previous experiments was used for each feature set. The final best

Feature set	Samples	Features	%	k	Prepr	Profit	nProfit
Wear	10,660	20	700	14	TL	1.59	86
Wear AN	10,520	20	1000	12	ENN	0.62	22
Usage	12,440	20	1000	16	TL	1.94	114
Usage AN	12,440	20	1000	20	CNN	1.60	110
Beam search 1	14,500	14	800	20	NCL	1.66	116
Beam search 2	14,500	15	800	16	TL	0.75	54
Expert	14,960	4	900	20	ENN	0.84	64

Table 1: The best settings for each of the feature sets (AN denotes age normalised). The total number of samples (second column) depends on the method used for selecting the data. The columns marked with % and k show the SMOTE parameter settings and the column labelled Prepr shows the SMOTE preprocessing method used. The Profit is evaluated for a PH of 15 weeks and the optimal training data set size (see Fig. 11 and the discussion in the text). The Profit (M€) depends on the test data set size, which depends on the method used for selecting the data. The rightmost column, labeled nProfit, shows per vehicle Profit (in €) which is the Profit normalised with respect to the number of vehicles in the testsets.

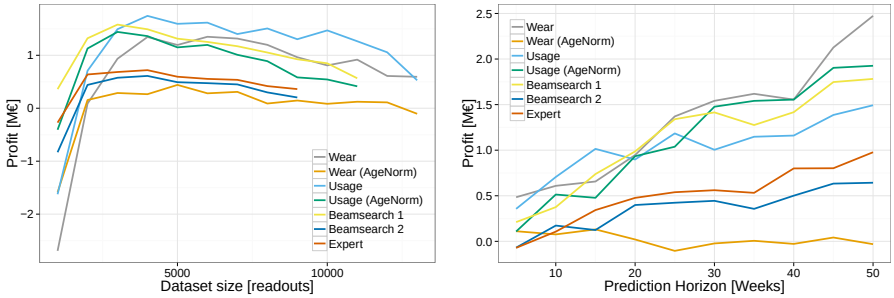

Figure 11: Final evaluation of all feature sets. The left panel shows how the Profit varies with the training data set size using a prediction horizon of 15 weeks. The right panel shows how the Profit changes with PH. The settings for each feature set are listed in Table 1.

settings for each feature set are summarised in Table 1, together with the basic data for each feature set.

The left panel of Fig. 11 shows how varying the training data set size affects the Profit. The PH was set to 15 weeks, which is a practical PH, even though many of the feature sets perform better at higher values of PH. From a business perspective is a PH of 30 weeks considered too long, since it leads to premature warnings when the vehicle is likely to survive one more maintenance period. The ordering of feature selection algorithms is mostly consistent; *Usage* is best, with the exception of very small data sizes where it is beaten by *Beam search 1*.

The left panel of Fig. 11 also shows an interesting phenomenon where profit grows and then drops as the data set size increases. This is unexpected, and we are unable to explain it. It may be related, for example, to the k parameter of the SMOTE algorithm. The right panel of Fig. 11 illustrates how varying the prediction horizon affects the Profit, using all available data for each feature set. In general, the longer the PH the better the Profit. The relative ordering among feature sets is quite consistent, which

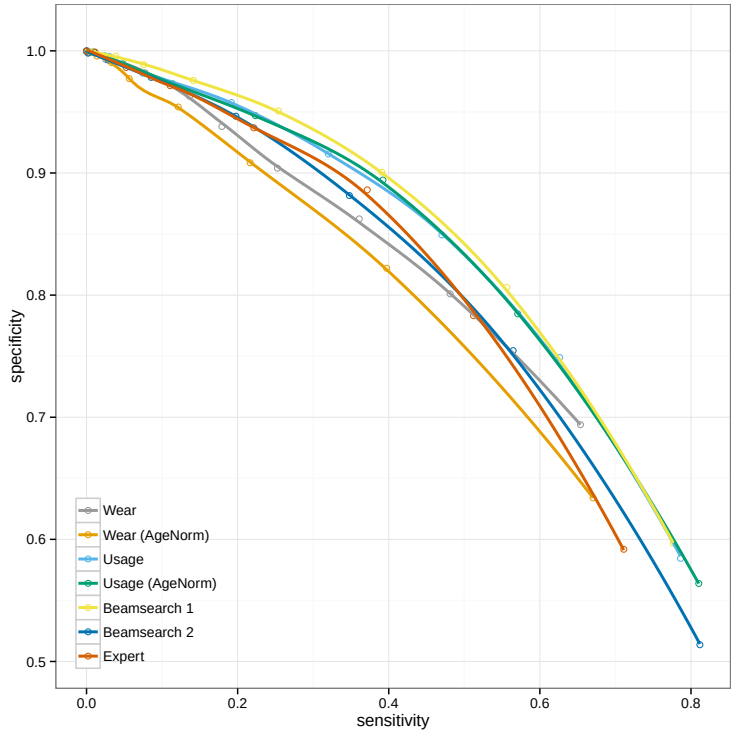

Figure 12: Sensitivity and specifity for the classifiers based on each feature set using the optimal settings in Table 1. Profit increases from lower left towards the upper right.

indicates that neither of them focus solely on patterns of wear. Such features would be expected to perform better at lower PH when the wear is more prominent.

The performances listed in Table 1 are for one decision threshold. However, the classifiers can be made to be more or less restrictive when taking their decision to recommend a repair, which will produce different numbers of true positives, true negatives, false positives and false negatives. Figure 12 shows the sensitivity–specificity relationships for each feature set (i.e. each classifier using each feature set). The perfect classifier, which certainly is unachievable in this case, would have both sensitivity and specificity equal to one. It is, from Fig. 12, clear that the feature sets *Beam search 1* and *Usage*, with or without age normalisation, are the best from the perspective of sensitivity and specificity. All three are better than the *Expert* feature set. Profit is not uniquely defined by specificity and sensitivity; it depends on the data set size and the mix of positive and negative examples. However, Profit increases from low values of specificity and sensitivity to high values.

6. Conclusions

Transportation is a low margin business where unplanned stops quickly turn profit to loss. A properly maintained vehicle reduces the risk of failures and keeps the vehicle operating and generating profit. Predictive maintenance introduces dynamic maintenance recommendations which react to usage and signs of wear.

We have presented a data driven method for predicting upcoming failures of the air compressor of a commercial vehicle. The predictive model is derived from currently available warranty and logged vehicle data. These data sources are in-production data that are designed for and normally used for other purposes. This imposes challenges which are presented, discussed and handled in order to build predictive models. The research contribution is twofold: a practical demonstration on these practical data, which are of a type that is abundant in the vehicle industry, and the techniques developed and tested to handle; feature selection with inconsistent data sets, imbalanced and noisy class labels and multiple examples per vehicle.

The method generalises to repairs of various vehicle components but it is evaluated on one component: the air compressor. The air compressor is a challenge since a failing air compressor can be due to many things and can be a secondary fault caused by other problems (e.g. oil leaks in the engine that cause coal deposits in the air pipes). Many fault modes are grouped into one label. Components with clearer or fewer fault causes should be easier to predict, given that the information needed to predict them is available in the data sources, and given that the fault progresses slow enough. We have not tested it on other components but plan to do so in the near future.

The best features are the *Beam search 1* and the *Usage* sets, with or without age normalisation. All three outperform the *Expert* feature set, which strengthens the arguments for using data driven machine learning algorithms within this domain. There is an interesting difference between the *Wear* and *Usage* feature sets. In the latter, there is little effect of doing age normalisation while on the first the age normalisation removes a lot of the information. This indicates that important wear patterns are linked to age, which in turn is not particularly interesting since age is easily measured using mileage or engine hours. It is possible that trends due to wear are faster than what is detectable given the readout frequency. This could partly explain the low performance of the wear features.

All feature sets show a positive Profit in the final evaluation. However, this depends on the estimated costs for planned and unplanned repair. There are large uncertainties in those numbers and one must view the profits from that perspective. The investment cost can probably be neglected and the important factor is the ratio in cost between unplanned and planned repair.

Acknowledgment

The authors thank Vinnova (Swedish Governmental Agency for Innovation Systems), AB Volvo, Halmstad University, and the Swedish Knowledge Foundation for financial support for doing this research.

Ahmed, M., Baqqar, M., Gu, F., Ball, A.D., 2012. Fault detection and diagnosis using principal component analysis of vibration data from a reciprocating compressor, in: Proceedings of the UKACC International Conference on Control, 3-5 September 2012, IEEE Press.

Batista, G.E.A.P.A., Prati, R.C., Monard, M.C., 2004. A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explorations Newsletter 6, 20–29.

Bendix, 2004. Advanced Troubleshooting Guide for Air Brake Compressors. Bendix Commercial Vehicle Systems LLC.

Bolón-Canedo, V., Sánchez-Maroño, N., Alonso-Betanzos, A., 2013. A review of feature selection methods on synthetic data. Knowledge and Information Systems 34, 483–519.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Buddhakulsomsiri, J., Zakarian, A., 2009. Sequential pattern mining algorithm for automotive warranty data. Computers & Industrial Engineering 57, 137–147.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 16, 321–357.

Choudhary, A.K., Harding, J.A., Tiwari, M.K., 2009. Data mining in manufacturing: a review based on the kind of knowledge. Journal of Intelligent Manufacturing 20, 501–521.

Dal Pozzolo, A., Caelen, O., Bontempi, G., . Comparison of balancing techniques for unbalanced datasets. URL: http://www.ulb.ac.be/di/map/adalpozz/pdf/poster_unbalanced.pdf.

Fogelstrom, K.A., 2007 (filed 2006). Prognostic and diagnostic system for air brakes.

Frisk, E., Krysander, M., Larsson, E., 2014. Data-driven lead-acid battery prognostics using random survival forests, in: Proceedings of the 2:nd European Conference of the PHM Society (PHME14).

Gusikhin, O., Rychtyckyj, N., Filev, D., 2007. Intelligent systems in the automotive industry: applications and trends. Knowledge and Information Systems 12, 147–168. doi:10.1007/s10115-006-0063-1.

Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182.

Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.A., 2006. Feature Extraction: Foundations and Applications (Studies in Fuzziness and Soft Computing). Springer-Verlag New York, Inc.

Hazewinkel, M. (Ed.), 2001. Encyclopedia of Mathematics. Springer.

He, H., Garcia, E., 2009. Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering 21, 1263–1284.

Hines, J., Garvey, D., Seibert, R., , Usynin, A., 2008a. Technical Review of On-Line Monitoring Techniques for Performance Assessment. Volume 2: Theoretical Issues. Technical review NUREG/CR-6895, Vol. 2. U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research. Washington, DC 20555-0001.

Hines, J., Garvey, J., Garvey, D.R., Seibert, R., 2008b. Technical Review of On-Line Monitoring Techniques for Performance Assessment. Volume 3: Limiting Case Studies. Technical review NUREG/CR-6895, Vol. 3. U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research. Washington, DC 20555-0001.

Hines, J., Seibert, R., 2006. Technical Review of On-Line Monitoring Techniques for Performance Assessment. Volume 1: State-of-the-Art. Technical review NUREG/CR-6895. U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research. Washington, DC 20555-0001.

Jardine, A.K., Lin, D., Banjevic, D., 2006. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing 20, 1483–1510.

Jayanth, N., 2010 (filed 2006). Pat.no 7,648,342 b2 - compressor protection and diagnostic system.

Liao, L., Köttig, F., 2014. Review of hybrid prognostics approaches for remaining useful life prediction of engineered systems, and an application to battery life prediction. IEEE Transactions on Reliability 63, 191–207.

Ma, J., Jiang, J., 2011. Applications of fault detection and diagnosis methods in nuclear power plants: A review. Progress in Nuclear Energy 53, 255–266.

Medina-Oliva, G., Voisin, A., Monnin, M., Léger, J.B., 2014. Predictive diagnosis based on a fleet-wide ontology approach. Knowledge-Based Systems 68, 40–57.

Molina, L., Belanche, L., Nebot, A., 2002. Feature selection algorithms: a survey and experimental evaluation, in: Proceedings of IEEE International Conference on Data Mining, pp. 306–313.

Napierala, K., Stefanowski, J., 2012. BRACID: a comprehensive approach to learning rules from imbalanced data. Journal of Intelligent Information Systems 39, 335–373.

Peng, Y., Dong, M., Zuo, M.J., 2010. Current status of machine prognostics in condition-based maintenance: a review. International Journal of Advanced Manufacturing Technology 50, 297–313.

Prytz, R., Nowaczyk, S., Rögnvaldsson, T., Byttner, S., 2013. Analysis of truck compressor failures based on logged vehicle data, in: Proceedings of the 2013 International Conference on Data Mining (DMIN13). URL: `http://worldcomp-proceedings.com/proc/p2013/DMI.html`.

R Core Team, 2014. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. URL: http://www.R-project.org.

Rajpathak, D.G., 2013. An ontology based text mining system for knowledge discovery from the diagnosis data in the automotive domain. Computers in Industry 64, 565–580.

Reimer, M., 2013. Service Relationship Management – Driving Uptime in Commercial Vehicle Maintenance and Repair. White paper. DECISIV. URL: http://tinyurl.com/q6xymfw.

Saeys, Y., Inza, I., Larraaga, P., 2007. A review of feature selection techniques in bioinformatics. Bioinformatics 23, 2507–2517.

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., Schwabacher, M., 2008. Metrics for evaluating performance of prognostic techniques, in: Prognostics and Health Management, 2008. PHM 2008. International Conference on, pp. 1–17. doi:10.1109/PHM.2008.4711436.

Schwabacher, M., 2005. A survey of data-driven prognostics, in: Infotech@Aerospace.

Si, X.S., Wang, W., Hu, C.H., Zhou, D.H., 2011. Remaining useful life estimation – a review on the statistical data driven approaches. European Journal of Operational Research 213, 1–14.

Sikorska, J.Z., Hodkiewicz, M., Ma, L., 2011. Prognostic modelling options for remaining useful life estimation by industry. Mechanical Systems and Signal Processing 25, 1803–1836.

Sokolova, M., Japkowicz, N., Szpakowicz, S., 2006. Beyond accuracy, f-score and ROC: A family of discriminant measures for performance evaluation, in: AI 2006: Advances in Artificial Intelligence. Springer Berlin Heidelberg. volume 4304 of *Lecture Notes in Computer Science*, pp. 1015–1021.

Sun, Y., Kamel, M.S., Wong, A.K., Wang, Y., 2007. Cost-sensitive boosting for classification of imbalanced data. Pattern Recognition 40, 3358–3378.

Van Hulse, J., Khoshgoftaar, T.M., Napolitano, A., 2007. Experimental perspectives on learning from imbalanced data, in: Proceedings of the 24th International Conference on Machine Learning, ACM, New York, NY, USA. pp. 935–942.

Changes to the manuscript (2014-01-18)

Reviewer #2: [...]Thus one interesting and valuable contribution would be a clear and rigorous formalization of the prognosis problem in the original setting considered by the authors. But this formalization is far from being complete. The general idea and general technical choices (ie working with available -but imperfect- data) are interesting, well presented and motivated, but it is rather difficult to have a clear idea of the technical developments made and implemented for this work. Indeed, a clear formulation of the prognostics problem as a classification problem is missing in the paper and would be really required to fully understand the work and assess the contribution: without this clear formulation, it is rather difficult to fully understand and assess the presented work[...]

We have addressed this issue together with remark 3, see below for an explanation.

1) The authors mention several times that "the specifics of the automotive domain make fault prediction and condition based maintenance a more challenging problem than in other domains". It would be interesting if they could further develop what are these "specifics" and explain why and in what these specifics make the problem more challenging.

We have added a paragraph to the Introduction explaining some of the difficulties.

2) Section 2 "Presentation of Data" presents the two used databases (VSR and LVD), but not really the data selected within these databases and used for the work. This presentation is really missing before Section 3 "Methods" and should be added to the paper.

We have added two paragraphs to Section 2, presenting in some more detail what is available in each of those databases, and how do we use this information. However, the data is considered sensitive by Volvo, which limits how descriptive can we be.

3) In this work devoted to RUL prognosis, the authors adopt an original problem setting: they do not want to predict the RUL, but rather to determine whether the item will survive until the next stop and the probability of this survival. The formalization of this original problem setting is not presented in the paper ; it would be an interesting complement to the paper to give a complete formalization of the considered problem, before presenting the methods used to solve it and the obtained results.

We have added a new section, 3 "Problem Formulation", where we present the formal definition of the setting, and present the exact question we aim to answer.

4) Page 7 - I am not fully convinced by the explanation about not taking into account "false negative" : even if from this point of view there is a status quo wrt to a more classical maintenance strategy, not taking them into account does not allow for a complete performance evaluation of the proposed predictive maintenance approach. Would it be difficult to take them into account and why?

We have expanded on the paragraph explaining this choice (now it's three paragraphs), to better rationalise our decision in this regard.

5) Section 3 "Methods" - Paragraph "Machine Learning Algorithm" : the method used (ie Random Forest) could be presented in mode detailed to make the paper more self-content.

We have added a description of the method.

6) Page 10 - 2nd paragraph below Figure 2 : I do not understand clearly the difference between the cases referred to as usage difference and wear difference, respectively, and why they are named in this way. What do the authors want to show by distinguishing these two cases. Further information on this would be welcome.

We have provided additional explanation concerning the rationale for the chosen names, and made it clear that we do not (as of yet, at least, we have some ideas for the future) make any use of this distinction – we consider both kinds of relations interesting.

7) Page 11 : The "age normalisation" effect & objective are not clear to me : does this operation remove the effect of the item age on the prognostic output ? If yes, this could be a problem, as the RUL depends on the age of the item. I am quite sure that my understanding is not correct here, but it could nice if the authors could explain the role of the age normalisation operation.

We have simplified this paragraph a little bit, and in particular made it clear that age normalisation is only done in the feature selection step, not in the actual classification.

8) Figures 5, 6, 7 : How is defined the "Accuracy" plotted in these figures. How does this "accuracy" relates to the prognostics problem considered at the beginning of the paper. Here again a clear formalization of the considered problem is missing to clearly understand the results. A clear formulation of the prognostics problem as a classification problem is required, and a clear explanation on how the performance of the prognosis is assessed (accuracy) is also required. Without this, it is really difficult to have a clear idea of the interest of the proposed approach and of its benefits

We have addressed this issue together with remark 3: we hope that the new section provides a better explanation.

9) Page 14 - Subsection "Feature selection" : it could be interesting to have a description of the selected features in at least one feature set.

We have listed the parameters that are included in the "Beamsearch 1" feature set.

10) Page 16 - Figure 8 : Any comments on figure 8) ? It is rather surprising to see that the "profit" increases with the PH, whereas at the same time the accuracy decreases. These figures would require further explanations. I have not enough explanations both on the prognosis method and on the performance evaluation (profit evaluation) to understand these results.

We have expanded the explanation, and provided our best intuition, but we cannot yet fully explain this result.

Some minor remarks

a) References: Some references are not detailed enough in the reference list, and should be completed to be useful for the reader (eg : Dal Pozzolo, A ; Jayanth, N ; Medina-Oliva, G. ; Reimer, M. ; Schwabacher, M. ; ....). Problem also with the first reference (missing line?)

Unclear references has been updated with suffient information such that they are easier found by the reader.

d) Figure 9 : F-score not defined

We have added reference to equations 2 and 8 to the description of the figure.