# ARISE

**Analytical Root-cause Identification in data Streams for detection of Emerging quality issues**


**A project within Big Automotive Data Analytics (BADA)**

Authors: Reza Khoshkangini, Peyman Mashadi, Claes Pihl, Tobias Nicklasson, Sepideh Pashami, Sławomir Nowaczyk, Peter Berck, Saeed Gholami Shahbandi, Adam Stahl, Parivash Pirasteh

Date: Thursday 15th August, 2019

FFI Fordonsstrategisk Forskning och Innovation

VINNOVA Energimyndigheten TRAFIKVERKET FKG VOLVO SCANIA VOLVO

# Contents

# 1 Summary

Product quality is one of the top priorities for commercial vehicle manufacturers. The ARISE project developed machine learning approaches for the early detection of quality issues and their analysis, integrating multiple available data sources.
Original Equipment Manufacturers (OEMs) are required to engineer increasingly specialised and custom-built machines, as customers expect more products with higher uptime and individualised functionality. While on the one hand product diversity fosters a brands desirability, on the other hand, it burdens the manufacturer with more complex challenges in quality assurance, maintenance and customer service. The higher uptime expectation from the customers amplifies this complexity. Under these circumstances, manufacturers can greatly benefit from an early detection of potential vehicle configuration and component quality issues.
Nowadays most operators continuously monitor the state of their vehicles through sensors, wireless communications and telematic equipment. Quality problems can be detected earlier by analysing this data, i.e., identifying emerging patterns, discovering trends, and detecting anomalies. Better understanding of the issues will also allow for more precise solutions to be applied, for example by choosing between vehicle recalls, redesigning or updates to usage guidelines.

# 2 Sammanfattning (in Swedish)

Produktkvalitet är en av de viktigaste prioriteringarna för tillverkare av kommersiella fordon. ARISE-projektet utvecklade maskininlärningsmetoder för tidig upptäckt av kvalitetsbrister och tillhörande analys, genom att integrera flera tillgängliga datakällor.
OEM-tillverkare (Original Equipment Manufacturers) förväntas utveckla alltmer specialiserade och specialbyggda maskiner, detta eftersom kunderna förväntar sig fler produkter med högre drifttid och individualiserad funktionalitet. Medan produktdiversitet åt ena sidan främjar attraktiva varumärken, belastar det åt andra sidan tillverkaren med mer komplexa utmaningar när det gäller kvalitetssäkring, underhåll och kundservice. Högre förväntan från kunderna på drifttiden förstärker denna komplexitet. Under sådana förutsättningar kan tillverkarna i stor utsträckning dra nytta av en tidig upptäckt av potentiell konfigurering av fordonen och brister kring kvaliteten hos komponenter.
Numera övervakar de flesta operatörer kontinuerligt sina fordons status genom sensorer, trdlös kommunikation och telematiklösningar. Kvalitetsproblem kan upptäckas tidigare genom att analysera dessa data, dvs. identifiera nya mönster, samt upptäcka trender och avvikelser. Bättre förstå else av problemen kommer också att möjliggöra mer exakta lösningar, till exempel i valet mellan att återkalla fordon, designa om eller uppdatera riktlinjerna för användningen.

# 3 Background

Product quality is a top priority for commercial vehicle manufacturers as it links to nearly all aspects of the ownership and operation of a fleet of vehicles. The key aspects of ownership are safety, productivity and maintenance, all of which are crucial to the customer, be it a hauler, bus operator or taxi service. Hence, keeping track of failure rates of different components is important. Growth in the number of failures for certain components is often an indicator of a quality issue. This will also translate into an increase in costs that a manufacturer has to pay on warranty claims and a decrease in customers' trust and satisfaction. Therefore, it is important to detect the imminent increase of the claim rate as quickly as possible, or even to predict it before it happens. Analysing WCs [Karim and Suzuki, 2005, Kalbfleisch et al., 1991] with the aim of deriving useful knowledge from products during their operations enables the manufacturers to increase awareness of the quality problems. It also supports Original Equipment Manufacturers (OEMs) in making decisions to initiate corrective actions as soon as possible. Predicting warranty claims, however, is a challenging task. The on-going demand for higher vehicle productivity requires increasingly specialised vehicles. This increasing diversity makes it harder to predict and detect quality problems, as they are usually a result of the combination of a specific use case, vehicle configuration and ambient conditions. Hence, a failure can be caused by different factors, across different components and be due to different usage patterns. In complex systems such as modern heavy-duty trucks there are thousands of potential components to monitor, with complex inter-dependencies. Sensors, wireless communication, and instrumentation such as telematic equipment support OEMs to continuously monitor the vehicles during their operation. This has enabled a variety of services such fleet monitoring systems and predictive maintenance.

Over the past decades significant efforts have been undertaken among researchers and manufacturers to develop various types of algorithms in order to decrease the amount of quality problems by means of early prediction [Kleyner and Sanborn, 2008, Corbu et al., 2008].Despite the significant progress in this area, most of the work on predicting warranty claims involves age-based approaches (both in terms of time and mileage) without taking the vehicles' usage into account, despite the fact that only such multi-dimensional data contains complete information. However, there are several recent investigations in the automotive domain on using Machine Learning

approaches for predictive maintenance where the usage of vehicles is considered [Nowaczyk et al., 2013, Prytz et al., 2015]. We believe that warranty claim prediction could be formulated in a similar fashion so that it can benefit from logged sensor data collected on-board vehicles.
Reducing the impact of quality issues requires monitoring and exploiting the data logs of individual vehivles. Significant cost savings will be achieved through combining on-board data, which is now being accessible via telematics solutions, with existing in-office knowledge including logged vehicle data, warranty claims, technical reports, and expert knowledge.
The ARISE project developed machine learning methods for automatic quality analysis that supports experts in their daily work. In particular, the main focus of this project is to address two issues that are significant costs drivers: early detection of emerging quality issues, as well as clustering and categorisation of new, incoming warranty claims. To this end, the proposed approaches exploit and integrate the available data sources such as vehicle logging data, claim data, technical vehicle specifications and other in-house structured and unstructured data. This combination of data sources is clearly Big Data in the context of automotive manufactures. Historical warranty and quality databases are large, and the latest generation of vehicle monitoring solutions, based on telematics, has the potential of being Big Data on its own. This combination of data sources is clearly Big Data in the context of automotive manufacturing. Historical warranty and quality databases are large, and the latest generation of vehicle monitoring solutions, based on telematics, has the potential to being Big Data on its own.

# 4 Purpose, research questions and method

The project consisted of several technical parts where multiple solutions were developed to understand and handle the quality issues. The ARISE project exploited the existing data in a novel way for the early detection of arising quality problems in vehicles already on the market. In essence, the project developed algorithms and models to detect arising quality problems and visualise them in an intuitive way. The following work packages were included in the project:

- Work package 1 and 2 were about quality issue detection and analysis, respectively. We constructed algorithms for the detection of quality issues based on anomaly detection; historical information about past

cases; following trends in the data and detecting changes in them. In the second work package we analysed and evaluated existing, as well as designed new algorithms for discovering the correlations between quality issues and various parameters such as vehicle usage, its configuration, the market, etc. The algorithms we have developed were mainly based on supervised (classification or regression) machine learning approaches.

- Work package 3 was about the warranty case recommendations, in which we investigated how quality issues should be described, compared and clustered. We have also performed analysis of historical quality journals, to improve the understanding of how discovered quality issues can be addressed, and how current processes in this area can be improved in the future.
- Work package 4 was about the demonstrator. This demonstrator was constructed based on the algorithms which were developed in work packages 1-3. These results are mainly implemented within the existing IT system used by the Q&CS department. Additionally, an external data analysis platform enables the results to be fed back to the warranty analysis process.
- Work package 5 was about the management and reporting, where the former part talks about the daily management of the project, and the latter one corresponds to project specific reporting, such as Vinnova reports and meetings.

# 5 Objective

The ARISE project started with the specific ambition to study the issues mentioned above. The expected results, as stated in the application, were:

a. A new framework to detect quality issues in the vehicles and root-cause identification.

b. A comprehensive analysis of Quality Journals (QJs) and historical quality issues.

c. A prototype for anomaly detection to compare the behaviour of healthy and non-healthy vehicles.

d. A tool to detect sub-optimal vehicle configuration.

e. A number of scientific publications

The results in the following sections address each of these stated goals.

# 6 Results and deliverables

The ARISE project consisted of three primary directions for exploration of data concerning warranty claims: logged vehicle data (LVD), diagnostic trouble codes (DTC) and quality journals (QJs). From this perspective the results obtained can be divided into two main groups. The first are predictive models used for forecasting the number of warranty claims, which can be used to improve early detection of quality issues. These models are built from the LVD data and from exploiting Diagnostic Trouble Codes (DTCs), including classification and regression methods. The second are deliverables related to the data mining of the quality journals and historical warranty claims. These separate focus areas have lead to presentations, software and several conference and journal publications. In this section we will present an overview of the insights we have obtained.

## 6.1 Contribution to FFI & BADA goals

We have contributed to the following FFI goals:

- Effective and efficient quality control
  Faster and more accurate quality control increases the overall product development efficiency. This strengthens competitiveness of Volvo as well as the Swedish automotive cluster.

- New approaches developed within the automotive industry
  Solutions developed in ARISE push the technology to the forefront of Big Data within the automotive industry. This increases the need for, and importance of, highly skilled personnel and providing new career opportunities in Sweden.

- Safer and sustainable society
  Safer and more environmentally friendly vehicles with higher quality reduce waste and contributing to a more sustainable society through a prolonged product life time.

- Improving the knowledge on data analytics ARISE increases the knowledge and competence on data analytics within Volvo. With more and more data being collected by the automotive industry, it is important to continuously explore new ways to use this information for increasing competitiveness. This strengthens innovative capacity, not only within Volvo but also in Sweden.

- Increased competitiveness in Swedish research community and industry ARISE contributes to the Swedish research community through the cooperation between Halmstad University and Volvo. The research focuses on machine learning algorithms for doing pattern recognition and statistical modelling on streaming data. New methods developed in the project, originally tailored for the automotive industry, can later be extended to cover other business areas – Halmstad University has a history of successful technology transfers of this kind.

We have contributed to the following BADAs goals:

- Business
  The project showcases the business benefits that are associated with Big Data. Unknown or wrongly categorised quality problems can be very costly and any method that allows for earlier and more precise detection is very valuable for vehicle manufactures. ARISE demonstrates that by combining multiple already available data sources it is possible to reduce the detection time and increase the root cause precision, which in return lowers the impact of quality problems. Arise combines a wide spectrum of competences, combining research (HH), advanced engineering (Volvo Advanced Technology & Research, AT&R) and product development (Volvo Quality & Customer Satisfaction, Q&CS), providing plenty of opportunities for knowledge exchange.

- Technology
  The second result of the project is evaluation of the quality of the available data, in terms of quantifying the business benefits that they can provide. For a number of years now Volvo and Halmstad University have been collaborating on developing new anomaly detection and machine learning algorithms that are particularly suitable for automotive industry. That work has primarily been focusing on the diagnostics, predictive maintenance and uptime areas. In the ARISE project it has

been extended to create novel algorithms for early detection and analysis of quality problems in vehicle populations. In particular, the Big Data aspect is an important issue to handle, as populations of hundreds of thousands or even millions of trucks generate huge amounts of data that needs to be processed in an efficient manner. ARISE also covered the human-machine interaction aspects of advanced machine learning algorithms. Due to the involvement of product development experts in the project, we have put an emphasis on precise methods for evaluation of both existing and new algorithms. ARISE showcased the proposed methods in a demonstrator based on historic and new quality cases.

## 6.2 Early Prediction of Claims Using Claim data and LVD

### Incremental Failure Rate Prediction

In this experiment we only focus on the claim data to build our prediction model. Thus two questions are investigated; first, how much past failures can be used to predict future failures? Second, as the in-service time of vehicles increase, how much does this incremental information help forecasting.
The setting of the problem to answering the two questions is as follows. Let's define production dates as $pd_1, pd_2, ..., pd_t$. At each production date, a batch of vehicles are produced which for $pd_i$ are denoted as $vb_i$. Also, let's define the time in-service parameter as $st$. The $st$ parameter is the parameter that is going to be incremented as time in operation of vehicles increase, and hence more data related to failure can be acquired. Also, note that for each vehicle, failure must be calculated from the operation start data. Having defined these parameters, now the number of failures for each production date can be defined as follows.

$$F_i^{st} = \sum_{v \in vb_i} f_v^{st} \tag{1}$$

In Eq.1, $f_v^{st}$ is the number of failures claimed by vehicle $v$ during $st$. In order to find the failure rate $F_i$ is normalised by the number of vehicles in the $ith$ batch which is denoted as $n_i$.

$$FR_i^{st} = \frac{F_i^{st}}{n_i} \tag{2}$$

As the in-service time increases the number of failures will be increased. So, for instance, if we set the level of granularity for $st$ to be one month, then $F_i^{st=2}$ counts number of failures in both first month and second month. Therefore, it is basically the cumulative sum of failures during the chosen duration.
To answer the first question —how much past failures can be used to predict future failures— the correlation between failure rate during chosen months in-service and final chosen number of months in-service will be calculated. Then, a linear regression model will be used to predict future failure from past failures.
To answer the second question —as the in-service time of vehicles increase, how much this incremental information can help forecasting— the in-service time will be increased and the corresponding correlation will be calculated. In other words, we will look at how much correlation will be increased as more information about failure will be gathered.

**Experiment 1: analysing the correlation between failures in an incremental fashion:** To analyse the correlation of failure rate in an incremental manner, failure rate after 12 months in-service is considered as the target ($FR_i^{st=12}$).
The result reported in Table.1 is for all component groups. It can be seen even after one month in-service the correlation to failure rate in 12 month in-service($FR_i^{st=12}$) is 0.783, which is quite considerable. And, as expected as the vehicles months in-service increase the correlation will increase.
Figure.1 shows the result of applying regression on past failures as dependent variable and 12 months in-service failure rate as independent variable. For visualisation purposes, the line-plots show three different duration in service (one, six, and 12 month(s)). However, for the bar-plot, Mean Absolute Error (MAE) is shown for all different months in-service. As expected, the error decreases as the months in-service increases. The over-estimation of the regression model is due to the fact that there are some production dates where the ratio of final failure rate to earlier failure rate are much higher than the other production dates. These production dates cause a larger slope for linear regression model which results in over-estimation.
According the result that we obtained and reported above, we can conclude that *past failures can be a valuable source of information to forecast future failures. Moreover, as more data is collected about the past failures the ef-*

Table 1: Incremental failure rate correlation

| Month in service | $\text{corr}(FR_i^{st=i}, FR_i^{st=12})$ |
|---|---|
| $st = 1$ | 0.783 |
| $st = 2$ | 0.823 |
| $st = 3$ | 0.854 |
| $st = 4$ | 0.886 |
| $st = 5$ | 0.910 |
| $st = 6$ | 0.930 |
| $st = 7$ | 0.948 |
| $st = 8$ | 0.963 |
| $st = 9$ | 0.974 |
| $st = 10$ | 0.986 |
| $st = 11$ | 0.995 |

*fectiveness of the past failures is becoming bolder.*

**Classification**

In this section we illustrate how machine learning algorithms can be leveraged for failure detection taking into account not only age and mileage, but multiple parameters which express vehicle past behaviors during their operations. Before conducting the prediction process, the integration is needed to merge the LVD and claim datasets, creating an integrated dossier with both the usage and failure information for all the vehicles. We merge the two datasets based on vehicles Chassis id, Date of readout and Date of claim report. To this end, we select a time-window of one month preceding each warranty claim, and consider this to be the interval in which the symptoms of imminent failure are most likely to be visible, and when the vehicle usage has the highest effect on a failure. The conceptual view of labelling positives (non-healthy vehicle) and negative (healthy vehicle) target values in LVD data as a merge process.
We keep this integration setting for this classification problem and implemented multiple experiments as follows:

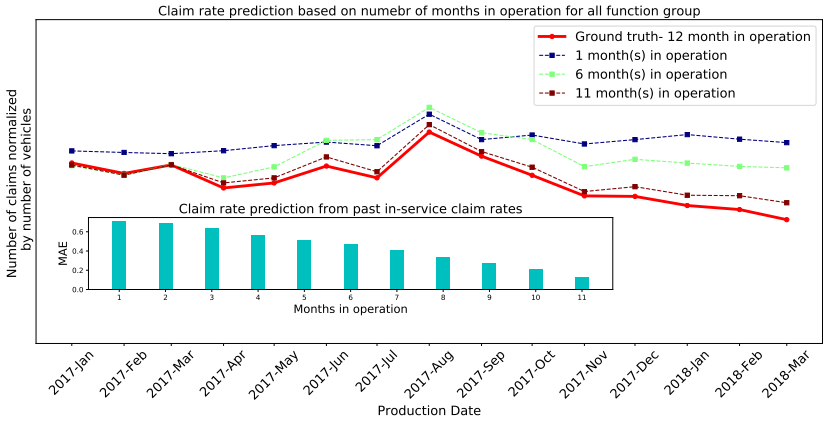



Figure 1: Failure rate of production dates for different month(s) in-service.

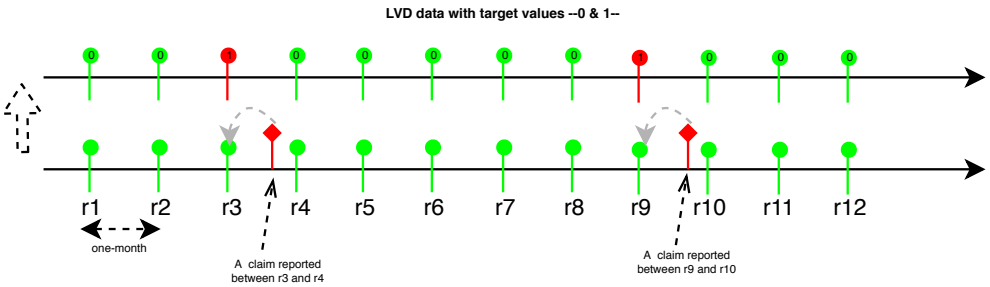



Figure 2: Overview of the labeling process.

**Experiment 1: component failures detection based on Logged Vehicle Data stream in a short time interval before the failure** In this experiment we look at two alternatives of Monthly and Seasonally settings. We focus on the issue of predicting warranty claims taking into account the usage of vehicles in the past. The detailed information of how we construct monthly and seasonally experiments are shown in Figure 3. and described in the following sections:

**Monthly:** To construct the training set in this monthly experiment, data from 2016 and 2017 were taken into consideration, and data from 2018 is used for validation part.

To train the model, we employed GradientBoostingClassifier (GB)[1], which

[1]We have used *sklearn* library in Python to build the model

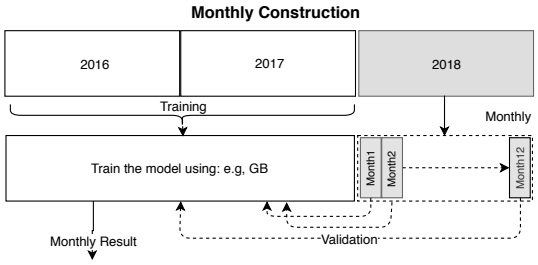
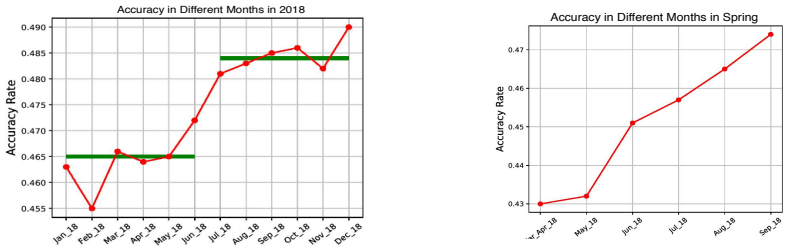



Figure 3: The Construction of Training and Test Sets in Classification.




(a) Monthly (b) Seasonally

Figure 4: Monthly and seasonally accuracy in 2018.

works based on a type of decision tree (CART) [Steinberg and Colla, 2009]. Once the model is trained using data from 2016-2017, the test set is used to validate the classification method in the form of 12 partitions, which is shown in Figure 3. Starting from 1 to 12, each partition includes the data logged in that specific month over the year of 2018. Hence, the samples from all vehicles e.g., in the first month, is considered as one partition, and used to validate the model for that particular month. The result from each partition is depicted in the plot in Figure 4a.

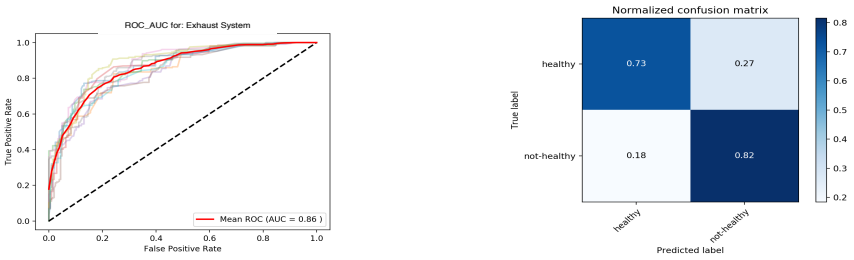
Figure 4a shows the monthly accuracy performance over the year. Within the twelve months validation, the maximum performance that we could obtain is 50% correct classifications in the last month. The unbalanced data in the training and test sets might be the reason of this low accuracy performance throughout the year. Although the accuracy value indicates low performance, this figure with regards to the very low baseline shows an admissible result. In addition, the GB classifier provides the area under the

https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

curve (AUC) [Bradley, 1997] with 0.66 (average over all partitions in the test set) showing that the model works better than random classification. To have an in-depth look at the performance of the prediction over time, a distinguishable pattern can be seen between the first and the second six-month of the year. It basically reveals that over the months the data is getting more meaningful which has a contribution to the model. This explains the consistent growth of the accuracy through months.
**Seasonally:** To evaluate this experiment we have trained the model based on data logged during the Spring and Summer (March–September) over two years, and validated it over the same period the following year. For this experiment, in total 20.000 data points are considered for training set and 3720 for validation set. The conceptual view of training and test sets in monthly and seasonally experiments are illustrated in Figure 1. Similar to the monthly evaluation, we observed that accuracy over the first six-months is increasing starting from 43% to 48%, however we did not obtain a high accuracy value. Taking into consideration both monthly and seasonally plots, we can observe quite similar results in accuracy metrics. In contrast with monthly, though, a much worse AUC value of 56% was achieved in this experiment. To have an in-depth look at both monthly and seasonally evaluation results, we can observe a very similar pattern for the common months with respect to accuracy rate. Thus, to conclude, we can state that *LVD data has a potential value for early claim prediction over time, and the performance of this claim prediction – for a component in the injection system – is not dependent on the seasonality.*

**Experiment 2: failure risk detection based on Logged Vehicle Data stream without specifying the time of the failure** In this experiment we intend to assess whether or not the proposed system is able to predict which vehicles are at risk of failure much more accurately then when this failure will occur, in particular predicting healthy and non-healthy vehicles taking into account a component relates to the exhaust system. Hence, to conduct this assessment, we distinguish two types of vehicles: healthy and non-healthy ones. Healthy vehicles are those that do not have any failure claim during their lifetime, while non-healthy ones have at least one failure. At this stage we do not, however, distinguish when the failure happened. To differentiate from the previous experiment, here we consider a component which is part of exhaust system. We have also selected a balanced data set

consisting of 7.000 data samples. Then, 100 iterations have been executed to randomly select 20% of the data as the test set and the rest of the data is considered as the training set to build the model. Receiver Operating Characteristic curve is depicted in Figure 5. It shows that the BG classifier performs very well in predicting the healthy and non-healthy vehicles, with AUC = 0.86. True negative and true positive ratios depicted in the confusion matrix (see Figure 3, left side), demonstrate a high performance of the classifier (0.84 and 0.73, respectively).




(a) AUC-ROC (b) Confusion matrix

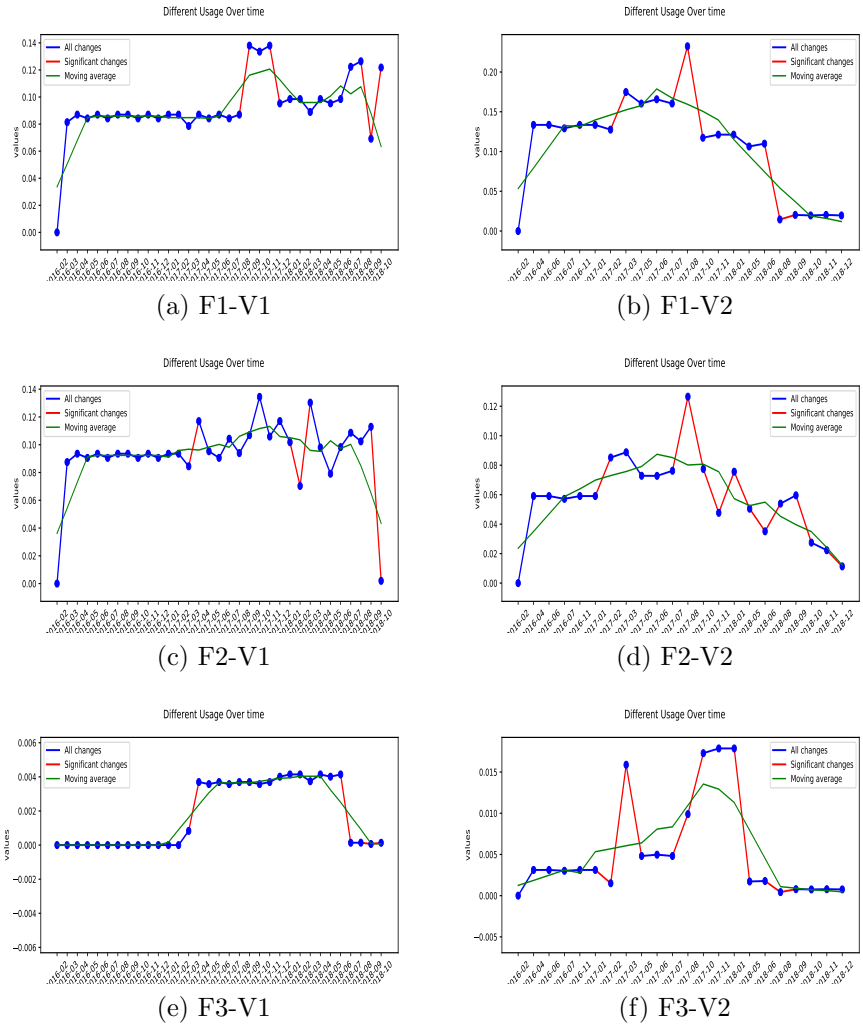
Figure 5: Performance Evaluation for Experiment 2.

Based on these results we can conclude that *although the model cannot express when a vehicle is going to fail, it performs very well by identifying vehicles which will have a failure during their lifetime*. In addition, taking together, *Experiment 1 and 2* are the part of our investigation to achieve the 'Objective 1' introduced in Section 5.

**Experiment 3: hidden information extraction and its relations to the performance of the system and claim ratio** This experiment seeks the hidden pattern that can be extracted from the vehicles usage and its effect to increase the performance of the recognition process, as well as expressing whether or not the changes (in particular significant) in vehicles usage can be a source of failures in vehicles during their operations. In order to reach to the above objective, we developed the feature extraction module to generate new features, not with combination of others to reduce the dimensionality of the data, but to find the hidden information that can not be recognised and used by feature selection, and merge to the selected features before the classification algorithm takes place. The integration of these two feature selection and extraction processes support our proposed approach to preserve

the original data characteristics for interpretability, and reinforce to higher discriminating the data samples using the new extracted pattern. In this part of the experiment, we have excluded categorical data, and focus in our approach only on the parameters which are logged using various sensors. Since these parameters express the cumulative data, their values are continuously increasing over time. Thus, to measure the conditions and usage within a given time period, we calculated the difference between each data point, and classified the changes into four levels: significant, moderate, low and no deviation.
Figure 6 shows an example of significant and moderate changes, highlighted by red and blue. These subplots show the changes in three different features (F1, F2 and F3) from two distinct vehicles (V1 and V2). As can be seen in the plots, these movements (up and down) are distinct in two vehicles. As an example, in Figure 6a, there are two significant changes that happened between months 7 and 8 in 2017, and similar changes monitored in the same duration in 2018. Another obvious pattern can be seen in Figure 6b, where two significant changes are observed between months 7 and 8 of the years 2017 and 2018, respectively. This hidden information indicates a form of pattern in the usage of vehicle that needs to be used for building the model in Learning and Prediction module. The green line in all the subplots also shows the average movement in the changes during the vehicles operation. Thus, to construct the data set to be trained by the classifier, we have merged these extracted changes as extra parameters to the list $F_s$, to get $F_{se} = \{f_{0s},\ f_{1s},\ f_{2s}, \ldots, f_{ms},\ f_{0ex},\ f_{1ex}, \ldots, f_{mex}\}$, which can be exploited further to build the model.
Exploiting the above extraction pipeline we conducted two sub-experiments. In the former, we tried to assess whether or not the hidden pattern can contribute to improving the prediction performance, and in latter we attempt to find the significant changes in vehicles usage pattern from the extracted information and their relation to the claim ratio.

**Experiment 3.1:** In this experiment, similar to experiment 2 we focus on healthy/unhealthy vehicle discrimination, without taking time of failure into account. The extraction pipeline is used to derive an additional 18 features including the deviation of vehicles' usage in different time periods. Hence, 39 features, consisting of the original and the extracted features, were taken to train the model using the same classifier. Then, we repeated the same




(a) F1-V1 (b) F1-V2

(c) F2-V1 (d) F2-V2

(e) F3-V1 (f) F3-V2

Figure 6: Usage changes between different features

experiment which was done for experiment 2.
In Table 2 the comparison between experiment 2 and experiment 3.1 is illustrated by the two sets of features. Concerning AUC and true positive rates, which are significantly important for our classification problem, we can clearly observe that patterns collected from the *Extraction* module have a value to increase the performance of the prediction. Although 7% decline in TNR is obtained for this observation, more than 10% improvement from

| | True Positive | True Negative | AUC | # of Features | Different Changes: Significant, Medium and Less | | Function Group |
|---|---|---|---|---|---|---|---|
| | | | | | Negative | Positive | |
| Feature selection | 0.73 | 0.84 | 0.86 | 19 | x | x | Exhaust System |
| Feature Extraction | 0.84 | 0.77 | 0.89 | 38 | yes | yes | Exhaust System |

Table 2: The results comparison between experiment 2 and experiment 3.1.

73% to 84% in TPR (predicting the failures) brings out the potential value of the hidden information at building the model. AUC is also slightly improved, taking into account these high numbers of samples, 3% indicates an admissible improvement for this warranty claim prediction w.r.t the previous setting experiment.

**Experiment 3.2** In this experiment, we used the extracted features and try to calculate the number of different changes in the style of vehicle usage during the vehicle operation. For example, the number of significant drops and raises (see 2b between months 12 and 14), when the vehicles is operating. In particular, in this evaluation, we intend to find the correlation between the number of usage changes and vehicles failures ratio. To this end, we defined a component including a set of rules in order to assign different sorts of changes in each reading point. Since vehicles performance are logged in a cumulative fashion, the changes (CHs) are calculated w.r.t the previous usage that are illustrated in the box below:

a. If a CH > +30%, then CH=pos_sig_chg

b. If +30% > CH > +20%, then CH=pos_med_chg

c. If +20% > CH > +10%, then CH=pos_low_chg

d. If a CH == 0, then CH=no_change

e. If -20% > CH > -10%, then CH=neg_low_chg

f. If -30% > CH > -20%, then CH=neg_med_chg

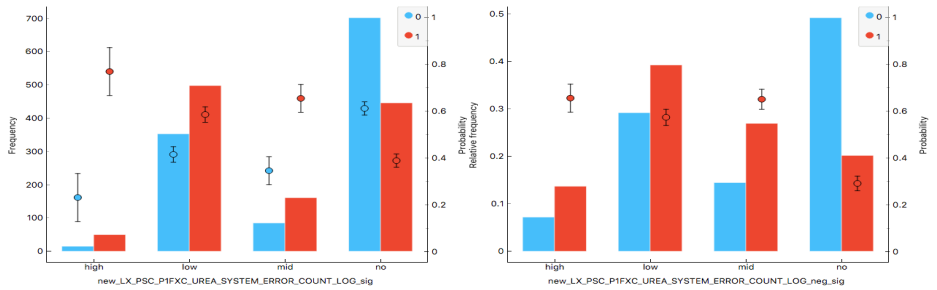
g. If a CH < -30%, then CH=neg_sig_chg

For example, in rule 1, if a change CH is bigger than the 30% of the previous usage that was logged and monitored, the change is assigned to a positive (rise) significant change (CH=pos_sig_chg), while in rule 7 CH=neg_sig_chg

is labelled for the negative change, if the change is less than 30% of the previous usage. It needs to remark that expert knowledge is used to define the values in the rule component in order to assign the various types of the changes. We have implemented this component on each extracted feature, so then a discritization process is carried out on positive significant changes (pos_sig_change) and negative significant changes (neg_sig_change) to categorise the calculated numbers–number of changes–into four groups of *high, medium, low* and *no-changes*. This decision has been taken from our hypothesis that unusual changes in vehicles usages might have the source of failures. Thus, to find the relationship between the number of changes and claim ratio, we grouped them in to healthy (0) and un-healthy vehicles (1), which are shown in Figure 7.

In the plots depicted in Figure 7, the y-axis shows the relative frequency of changes in four categories which are placed on the x-axis. These sub-figures clearly reveal that the proportion of significant positive and negative changes in unhealthy (labelled by 1 in the plots) vehicles are higher that the healthy (labelled by 0 in the plots) vehicles during their life. In contrast, the proportion of healthy vehicles are more than unhealthy, when we took into consideration no-changes to assess the correlation between them. The similar results have been observed, when medium and less significant changes were taken to consideration. Basically, the findings express a message that healthy vehicles have less usage deviation than unhealthy vehicles. We can also look at the significance of the difference between the changes in these population (for this test, we only considered the extracted significant changes). Taking into account the distribution of the changes and in-dependency of the population in both cases, we exploited the non–parametric Wilcoxon test for comparison of the changes distributions [Gehan, 1965]; we used it to investigate if the significant changes from unhealthy vehicles are statistically larger than those from healthy vehicles, and we applied it to all the changes which have been extracted from the selected features.

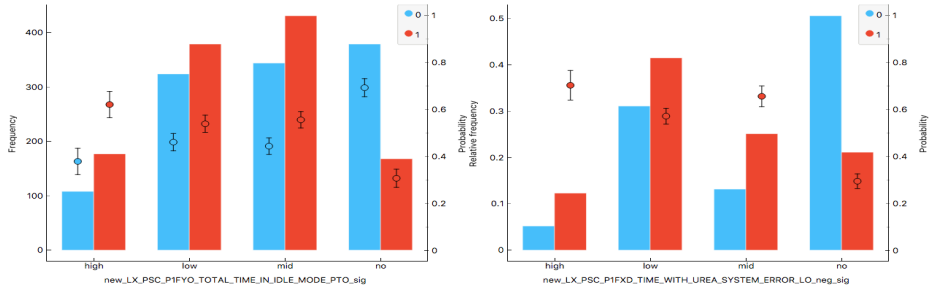
Table 3 shows (only 10 features are listed) the result of a Wilcoxon test on the two populations. As reported in the table, the p-value in all cases is less than the critical value (0.05) which indicates the two populations are statistically different by rejecting the null-hypothesis. Indeed, it is not surprising to obtain such significant difference in all cases, as the two populations contain remarkably high data samples. Thus to properly quantify these difference between the two populations, we applied Cohen's [Fritz et al., 2012] method to calculate the effect size. The calculated effect size values also confirm that




(a) Feature 1



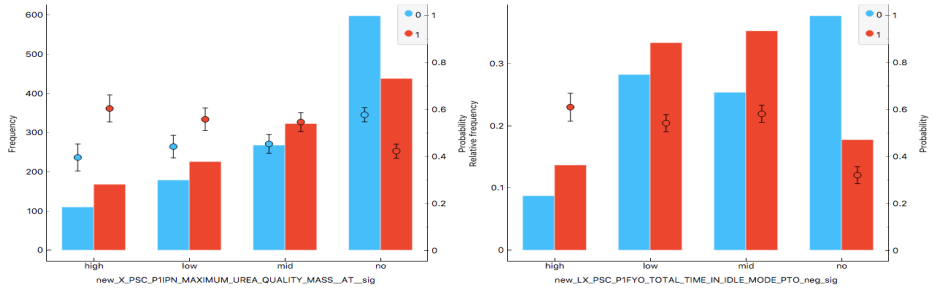

(b) Feature 2




(c) Feature 3




(d) Feature 4




(e) Feature 5




(f) Feature 6

Figure 7: Negative and Positive Changes Statistics

there is a difference between the two populations, but in some features we observe a smaller difference with 0.20, 23 or 0.33 effect size. While in some other features we obtained 0.50 or 0.47 which show a larger difference w.r.t the other features.

On the basis of the data above, we can conclude by stating that *exploiting the extracted hidden pattern may be conducive to higher prediction performance than relying only traditional feature selection process. This pattern also in-*

| Component | Wicxon Score | p-value | Status | Effect Size |
|---|---|---|---|---|
| PSC_P1FX | w=816510 | <2.2e-16 reject | significantly difference | 0.40 |
| TIME-WIT | w=816810 | <2.2e-16 reject | significantly difference | 0.41 |
| P1FYO-TO | w=798610 | <2.2e-16 reject | significantly difference | 0.33 |
| P1L80_TO | w=733800 | = 6.155e-08 reject | significantly difference | 0.20 |
| PCT_DIST | w=754770 | = 3.599e-08 reject | significantly difference | 0.20 |
| BRAKE_M | w=771140 | = 7.697e-11 reject | significantly difference | 0.31 |
| ERROR_C | w=865980 | <2.2e-16 reject | significantly difference | 0.50 |
| KALMAN_ | w=815470 | <2.2e-16 | significantly difference | 0.37 |
| MASS_AT | w=757030 | = 4.224e-09 | significantly difference | 0.23 |
| CUNS_F | w=798810 | <2.2e-16 | significantly difference | 0.38 |

Table 3: Significant Test and the Effect Size of the Difference. Column 'Component', refers to the name of the components in the vehicles so that we used the short form of the them.

*dicated that there is a correlation between unusual changes in vehicles usage and claim ratio, which answers the 'Objective 3' introduced in Section 5.*

**Experiment 4: analysing the effect of LVD and claim fusion to improve the prediction** In Sec.6.2, LVD measurements are used as the predictor variables to predict failures. However, past history of failures for a vehicle can be a meaningful predictor of failures. Taking into account only LVD measurements, a model tries to establish a relationship between LVD measurement and failure, while neglecting the relation between past failures and current failures that the model is trying to predict. It sounds natural to provide model with both LVD measurements and previous failures so that

it can use these two source of information at the same to predict failures. In other words, to take the history of failures of each vehicle into account , information about past failures are added to the LVD measurements as new features.
In order to be able to use these two sources of information, first, each one of them is explored separately in terms of feature engineering and modeling. Then the configuration for their fused version is explained.

**Past claims for LVD data**

To use the other source of information, which resides in Claim database, in this section the configuration to assign historical failures to each LVD reading is described.
In the LVD database, for each record, in addition to all measurements, log_date and vehicle_id are registered. For each record, by using the Claim database, the number of failures for the vehicle_id before the log_date is calculated, denoted as *n_past_claims*. As an illustration in Fig.8 number of claims reported before reading 11 and also reading 12 is four.
In this study, it has been founded the model's performance will be increased by including also three more engineered features— derivative of *n_past_claims*, mean of *n_past_claims*, and standard deviation of *n_past_claims*. An important technicality is that these statistics must be calculated after train-test-split, otherwise, there would be information leak and consecutively over-fitting. In a nutshell, four features are engineered: 1- *n_past_claims*,2- derivative of *n_past_claims*, 3-mean of *n_past_claims*, and 4-standard deviation of *n_past_claims*.

**LVD and past claims fusion**

To get the most out of data, the four features related to past claims are added to the corresponding LVD records. Notice that the features obtained from past claims don't contain information regarding measurements, hence many vehicles could potentially end up having very similar values for these four features. More concretely because we are using tree boosting methods, many similar combination of these four features could ended up in tree's leaves. The idea is that adding the LVD reading can give the model the possibility of further meaningful (meaningful in the sense of not over-fitting) splitting of tree's nodes.
In this experiments, the forecast interval is considered to be 30 days. Hence, if a claim accrues after the reading date until the end of the 30 days, then it is considered as a faulty readout otherwise as a healthy readout. Since there are large number of categorical features, using the state-of-the-art boosting


Figure 8: Assigning historical failure to LVD data

method CatBoost (Categorical Boost) seems natural. Since CatBoost uses various statistics of categorical data, it alleviate to some extend the need for extensive categorical feature engineering. In Figure.9 AUC curves for five subsets of features are shown. First, *n_past_claims* for each corresponding readout in LVD and the derivative of *n_past_claims* are considered as the features. As can be seen in Figure.9, the area is 0.605 which is an indication of a poor performance. However, an interesting observation is that by adding mean of *n_past_claims*, there is a huge improvement according to AUC which the area is 0.853. As an intuition why adding the mean feature cause a considerable boost, we can compare it to modeling failure with Poisson distribution. In failure detection using Poisson distribution one assumption is that the mean of failure is constant over time. Also, mean is very important in the Poisson distribution in the sense that the number of failure only depends on the mean value. In other words, Poisson distribution basically calculate the probability of occurring failures count based on mean value to compensate for the expected mean. Comparing our engineered mean of past claims to the Poisson mean parameter, we can gain insight why it is important. Intuitively, it seems logical to say having mean value as another feature, the model can compensate to keep the mean value constant. As the last engineered feature of past claims the standard deviation of *n_past_claims* is added. Adding std gives another boost to the model (AUC=0.938). Again, the same intuition for mean can be applied for the std feature. It gives the model the ability to predict in a way to compensate for the fluctuation in

failures. To see the forecasting power based only on LVD data the result of CatBoost model only on LVD measurements are reported. The AUC area is 0.924. Now, the idea is to combine LVD and past claims features to see the effect of this fusion.
The measurements from LVD are added to these four claim-related features to use the full potential of information. Adding LVD as features leads to another improvement in AUC (AUC = 0.964). As discussed in section 6.2 the idea is can give the tree boosting model the ability of further meaningful splitting.

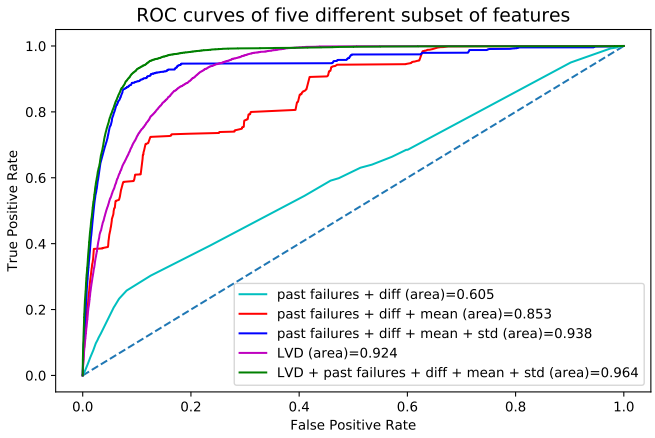



Figure 9: ROC curves of five different subset of features

The confusion matrix result for a specific threshold is shown in Fig.10
The results shown in Figure.9 demonstrates that the 'Experiment 4' can concluded by stating that *the best result is achieved when LVD and claim information are combined. Hence, it confirms the idea of using both source of information to gain higher predictive power.*

### Regression

In this part of the experiment, we intent to tackle the problem from a regression point of view. Similar to the classification pipeline, an integration process is needed to merge the LVD and claim data. Figure. 11 shows the general positive and negative target assignment values to LVD readings. Circle-top events mark positions of LVD readings in time, $rv\_xx$ in which $v$ is the vehicle

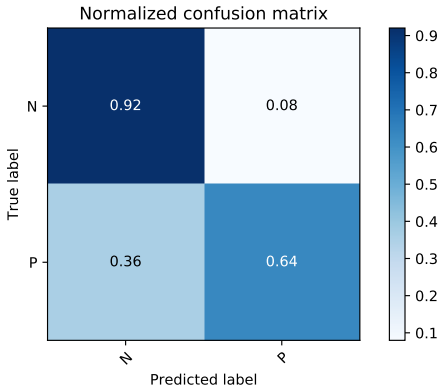



Figure 10: Normalised confusion matrix for fusion of LVD and past claims

identifier and $xx$ is the index to LVD readings. Diamond-top events indicate occurrences of failures (i.e. claims), with the same notation as LVD readings ($fv\_xx$). Here the targets are as for a binary classifications, in which all LVD readings within a certain interval leading to a failure (red rectangles) are marked as faulty (in red) and everywhere else is marked healthy (in green). In order to transfer this target assignment process to be applicable as a regression problem the following procedure is applied. First, a time-window is selected (in this experiment it is selected to be three months). Then, for each reading, number of failures occurred in the time-window interval is considered as the target. In other words, instead of calculating if a failure is happened during the selected time-window, number of failures is accounted as target. This way, we can approach the problem from a regression point of view.

**Experiment 1: Failure prediction time-based split** To conduct this evaluation, the data from 2016 to 2018-06-01 are considered as the training set in this setting of the experiment. Then to construct the test set we took the data which are logged after 2018-06-01, until 2019/02. To train the model, we employed Random Forest Regression (RFR)[2]. Once the model is trained, the test set is used to validate the regression method such that the vehicles which are logged in that specific dates are fed for validation. In

[2]We have used *sklearn* library in Python to build the model
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html

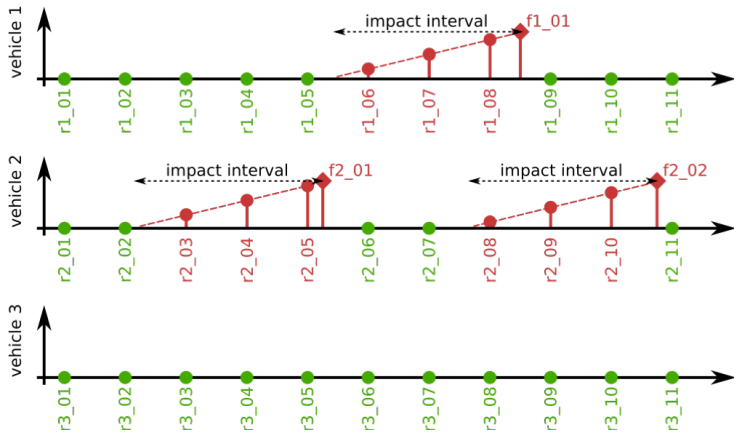



Figure 11: Overview of the labeling process in Regression.

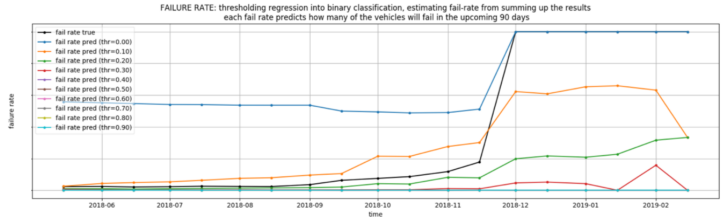



Figure 12: Time-based split, thresholding applied

order to get the result in the form of failure ratio, the trained model makes prediction for each LVD reading, which as a regression model is supposed to predict the number of failures are going to be happened during time-window interval. Then, each predicted value is converted to a binary value using a threshold. Finally, for each reading date, number of LVD readings predicted as failures are summed and normalised. Figure 12. shows the results obtained for different thresholds values.

The plot basically shows the percentage of the vehicles predicted to be failed in the upcoming 90 days. As illustrated in the plot, the green line with a threshold value of 0.2 relatively follows the trends which is shown via the black line. It is clear that the green line could follow the pattern till the end of the year 2018, and then started to have more distance to the ground truth. The gap between ground-truth and predicted values at the end of 2018 and beginning of 2019 is mostly due to the sampling effect — number of sample in that period in the database in much less than the other period.

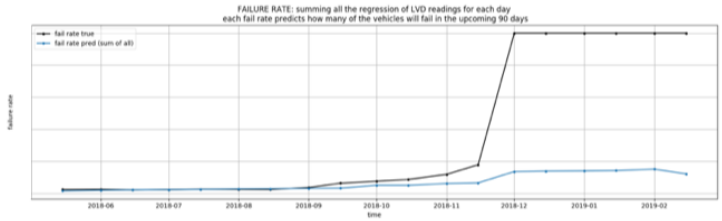



Figure 13: Sum: Time-based split

Figure 13 demonstrates the same result with the difference that this time the regression result is not converted to binary values using thresholds, instead at each point in time regression result for all vehicles logged in that point are summed up together and normalised to obtain the failure-ratio.
To conclude, the experiments which are constructed as classification and regression problems described above were a part our investigation–in ARISE– to achieve the 'Objective 1' defined in Section 5. These also resulted in the publications, which are listed in Section 7.2.

## 6.3 Early Prediction of Claims Using DTC

Diagnostic Trouble Code (DTC) are a useful tool for identifying the underlying cause of a failure on a specific component. As previously demonstrated in the In4Uptime FFI project, DTCs can also in specific cases be used to predict upcoming component failures.
In ARISE, we have investigated the predictive power of DTCs in a broader sense, looking at the prediction of emerging quality issues within a large and diverse vehicle population based on the frequency of DTC occurrences. The investigation has also been broader in that failure of all truck components have been considered, and triggering of every DTC has been considered as a predictor. In essence, the purpose of the calculation has been to find out whether a given DTC can accurately predict faults appearing on components in a given Function Group (FGRP), which is a collection of parts in a truck with a common purpose. Warranty claims data was used as the source of fault information for the function groups.
The calculation considered every combination of DTC and function group (900 000 in total). The vast majority of such combinations hold no predictive power (for instance because the DTC concerns the gearbox whereas the function group describes brake pads), however no manual selection of

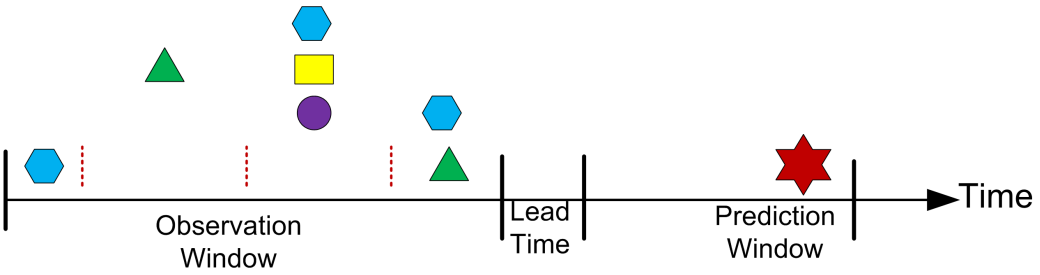



Figure 14: Period based framework consists of three parts, observation window, lead time and prediction window.

promising combinations was done to get as complete a picture as possible. The algorithm for assessing the predictive power consisted of calculating which DTCs were triggered within a time window before a claim on a vehicle (if a DTC is not followed by a claim in a certain function group, it has no predictive power for faults in that group), and then aggregating that information across the entire population. The correlation for every combination of DTC and function group was calculated for a number of time windows; the time windows are useful for determining the prediction time frame of a pair (i.e. how quickly the DTC can be said to lead to a fault).
In general, very few DTC-function group combinations proved to have a sufficient correlation for predicting faults. Out of the 900k combinations, only less than 20 had both high coverage (many failures were preceded by a DTC trigger) and high accuracy (many DTC triggers were followed by a failure), which are both necessary characteristics of a reliable predictor. The reasons for this are related to the purpose of the DTCs - they are primarily intended for fault diagnostics; not prediction. Many DTCs trigger too often, giving false signals that diminish their predictive power, and the time scale between DTC triggering and component breakdown can vary greatly. In the following section we described the proposed system framework to exploit DTC predict the upcoming failure.

**Interactive feature extraction for diagnostic trouble codes in predictive maintenance**

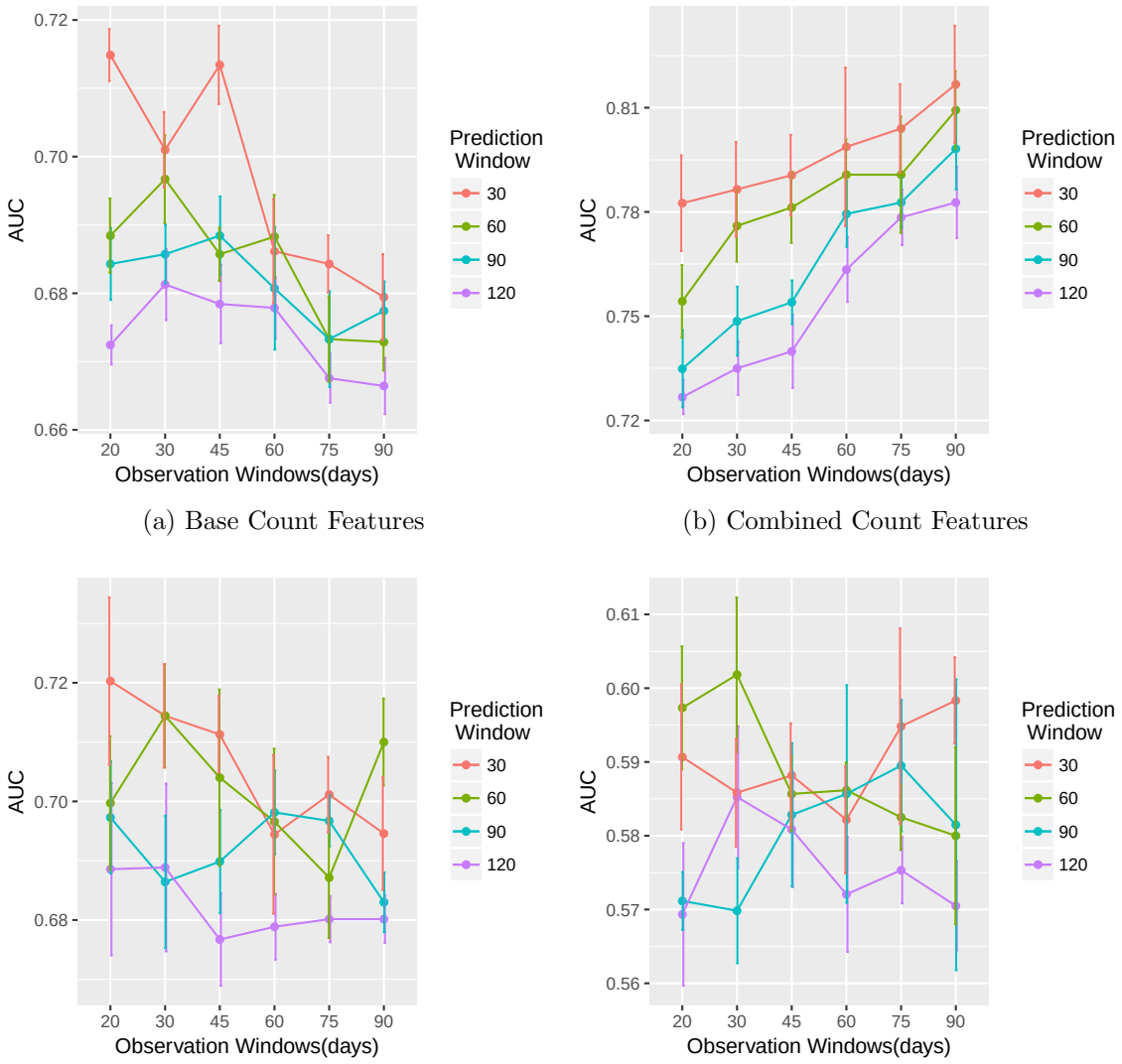
We develop a classification-based predictive maintenance framework. The goal is to identify components that are going to fail within a given prediction window, based on DTC data.

The first step towards building the prediction model is merging two datasets: the DTC data and vehicle repair data. The vehicle repair data contains historical information related to repairs that were performed on a component. We use period-based approach as a framework for prediction (see Figure 14). The period-based approach consists of three parts: observation window, lead time and prediction window. The vector of features is extracted from every observation window and use as an input data. The observation window is composed of a set of consecutive time intervals called sub-windows. Each sub-window spans fourteen days, corresponding to the time interval of DTC data collection. In this experiment we use random forest models for prediction task, since random forest algorithm works very fast and tends to perform well with complex input data. We set a maximum tree depth to 10 and the number of tree sets included in each stage is equal to 100.
Working with DTCs, there are some common parts of the knowledge which can be extracted, but we can gain a lot of new knowledge and distinctive features from experts to develop a framework for predictive maintenance.
Lead time is the minimum time interval that we would like to have preceding the failure time. A long enough lead time would provide sufficient time to deal with the failure that might arise. Since the goal of this paper is limited to outline the relation between DTCs and component failure rather than to create a generic model, for simplicity we consider lead time equal to zero. The period based framework predicts the future failure over a certain period, called prediction window. The class label of an instance is defined positive if there is any component failure inside the prediction window, otherwise is set to negative.
We start from a truck's delivery date to create an instances and then move the prediction point along the lifetime of the truck by a constant parameter as moving step. By repeating this process for all vehicles, we can obtain all the training and testing instances to build the model. The total number of learning instances would be dependent on the size of the windows generated from historical data and moving step. Our goal is to perform a thorough study on factors such as the size of observation and prediction windows and choices of features that could impact the performance of the classifier.

**Experiment 1:** Below we present results from experiments aimed at evaluating the proposed framework. These experiments showcase the effectiveness of extracted features and the idea of applying DTCs in two levels in the learn-

ing process. For all experiments the process of data preparation, instance generation and feature extraction have been done based on configuration of experiment. The first group of experiments was devoted to examining the different parameters that could influence in configuration of proposed framework and as a result the performance of prediction models. For instance, by assigning different values to length of observation window and prediction window, different sets of instances will be generated and as a consequence different results are obtained. Moreover, this experiment leads us to compare the usefulness of different extracted features in predicting the failure of component. The second category of experiment has focused on evaluating the effectiveness of combinations of feature classes in prediction.
An air suspension component and a powertrain component have been selected as example components due to their different functionality. Another reasons for picking these two components are high failure rate and their attendant costs. Applying the same set of DTCs as indicator of failures in these two components enables us to explore the possible relation between DTCs and failure in each component and to compare the effectiveness of the models in different parameter settings (e.g. different size of observation window and prediction window) based on accuracy of classification.
The data used in these experiments includes the records of 1.000 individual DTCs across 30.000 Volvo trucks with delivery date from January 2015 to January 2018. In average DTCs are triggered 50 times during lifelong of a truck. We used data from 22500 trucks to train the prediction model, and tested the model with data from the remaining trucks. It is also worth noting that since the set of trucks for training and testing are different, all the instances of same truck would belong to either training or testing, in order to avoid possible over-fitting problem.

**Performance comparison on different settings** The first series of experiments concerns the performance of prediction model taking into account different configuration of model parameters. This experiment allows us to investigate how the model reacts to using different levels of DTCs, which feature provides the better discrimination and most importantly, what is the best configuration for the size of observation window and prediction window. For simplicity of parameter tuning, the size of sub-window is fixed to two weeks. The performance was estimated using five-fold cross-validation. The cross-validation process is then repeated on training data five times with each



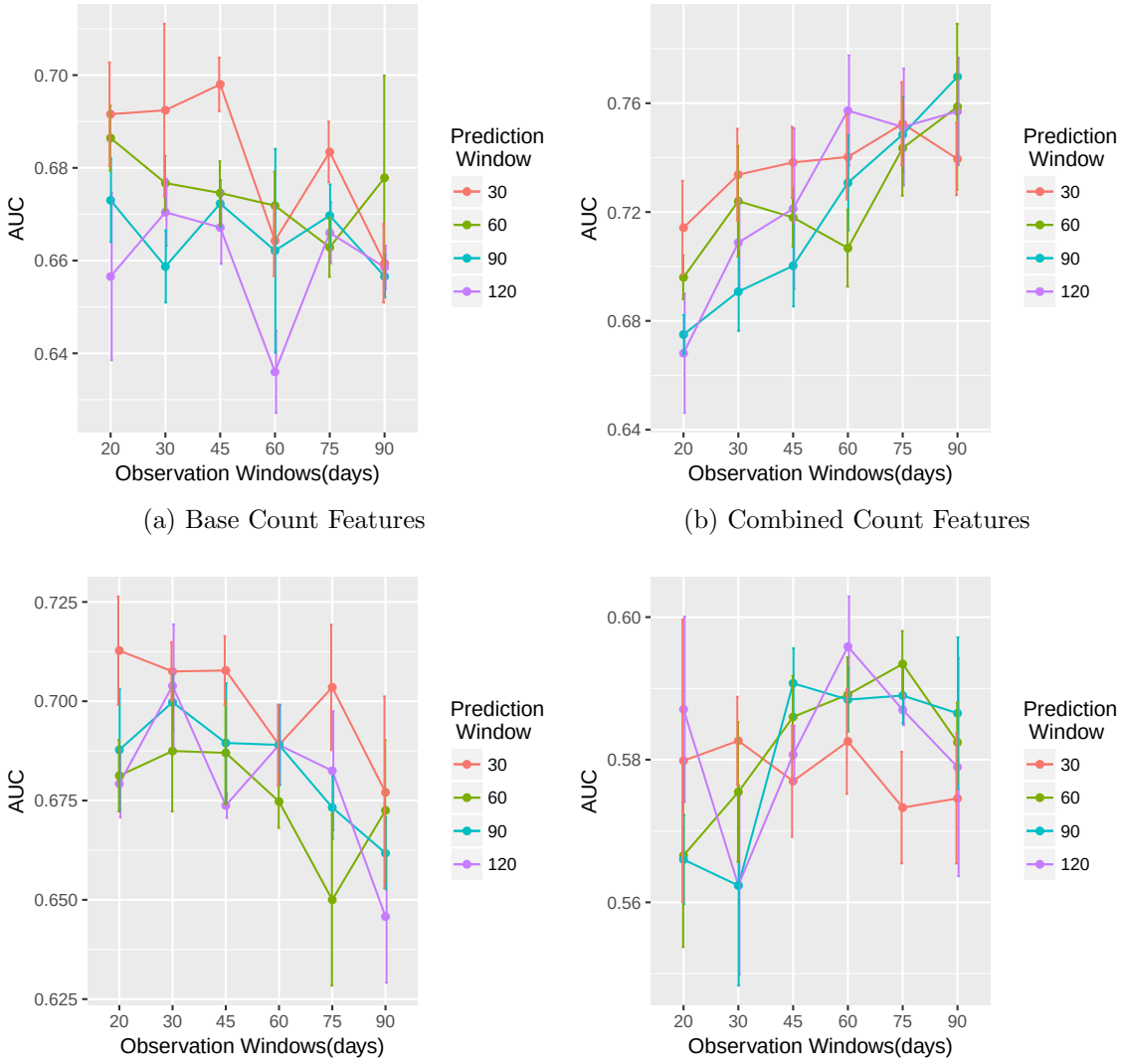

(a) Base Count Features

(b) Combined Count Features

(c) Historical Count Feature

(d) Trend-based Features

Figure 15: AUC values with respect to size of observation window and prediction window for powertrain component in aggregated level



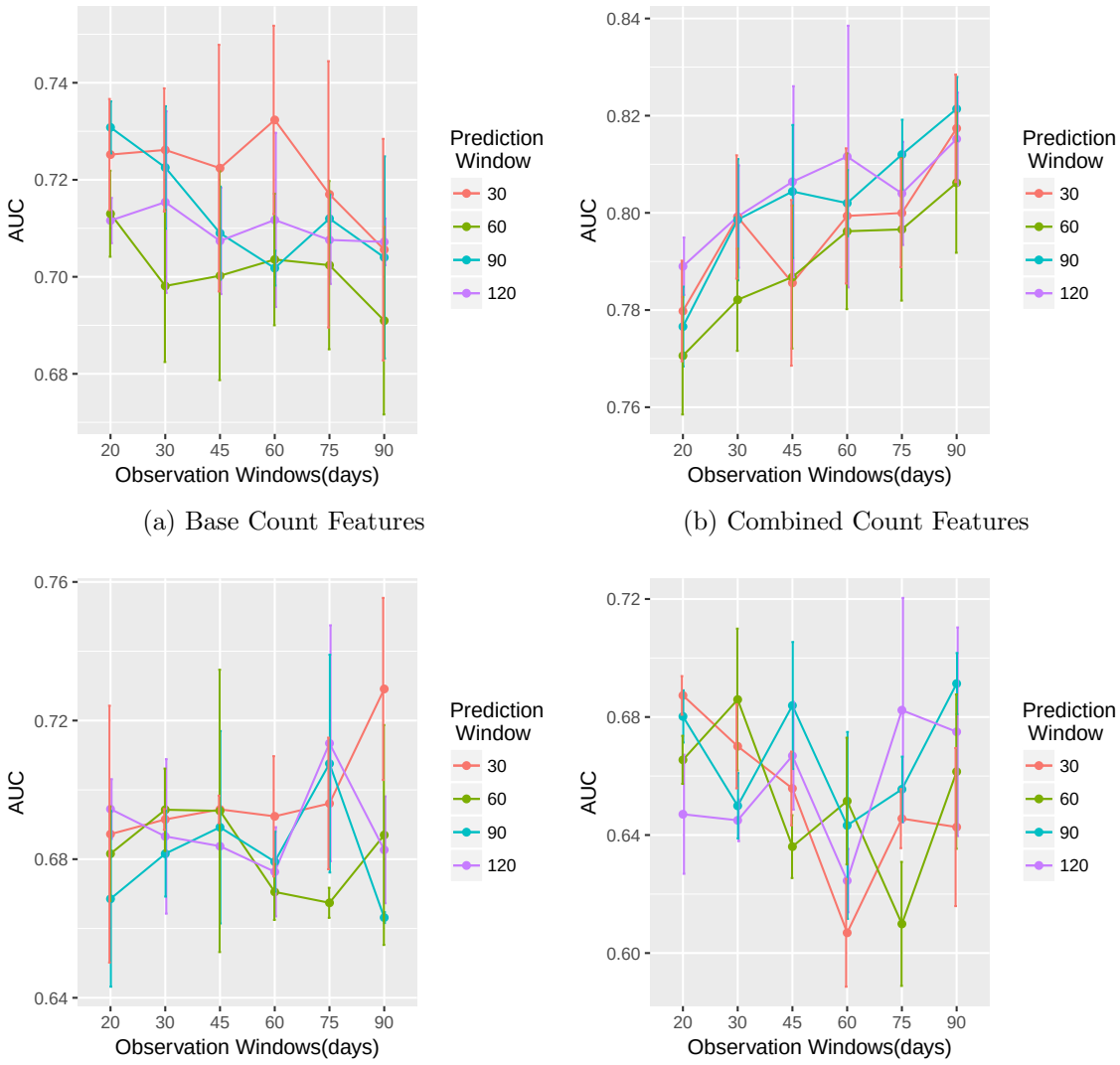

(a) Base Count Features

(b) Combined Count Features

(c) Historical Count Feature

(d) Trend-based Features

Figure 16: AUC values with respect to size of observation window and prediction window for powertrain component in individual Level



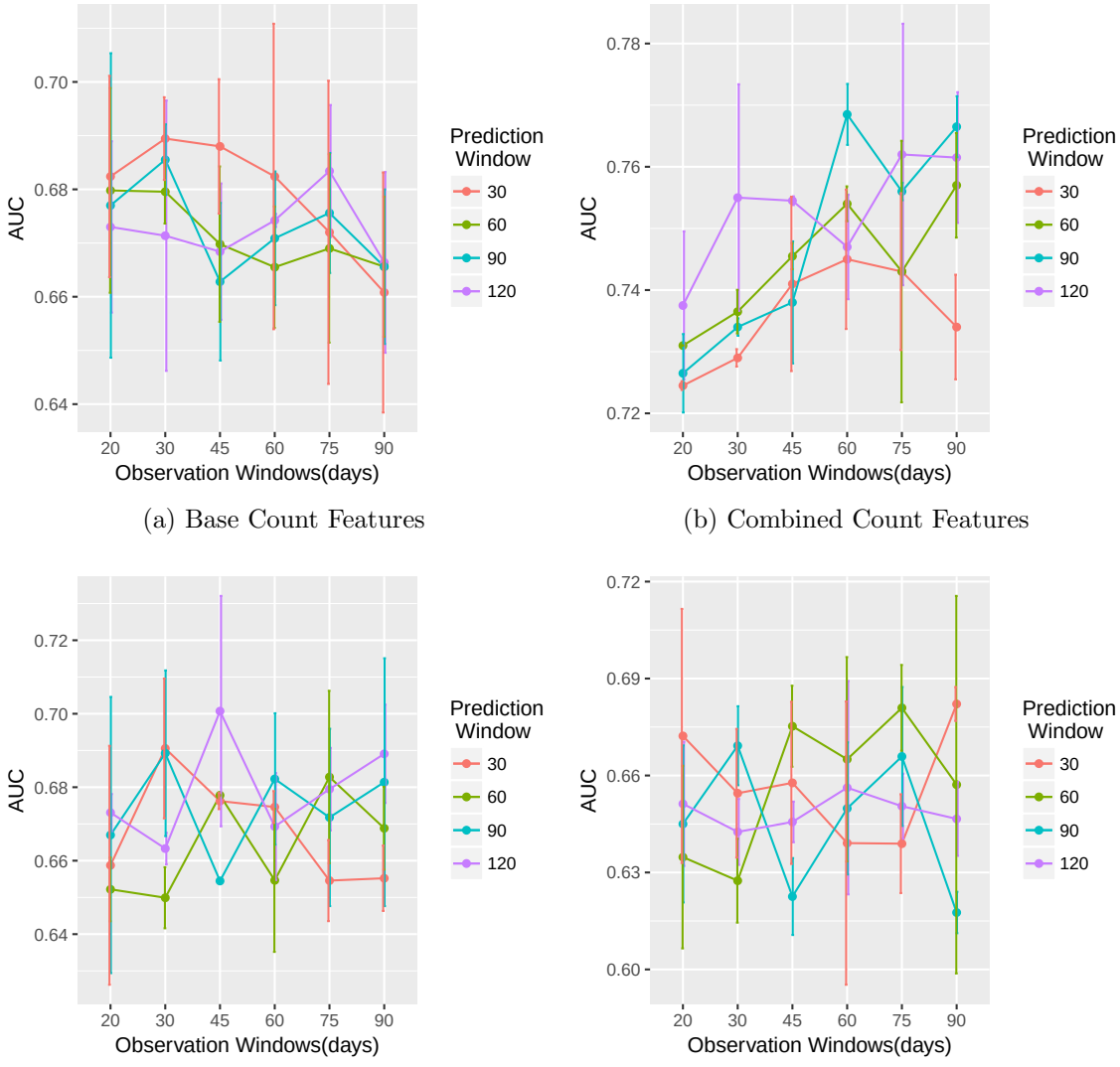

(a) Base Count Features

(b) Combined Count Features

(c) History Count Features

(d) Trend Count Features

Figure 17: AUC values with respect to size of observation window and prediction window for air suspension component in aggregated level




(a) Base Count Features

(b) Combined Count Features

(c) History Features

(d) Trend Features

Figure 18: AUC values with respect to size of observation window and prediction window for air suspension component in individual levels

of the five folds used exactly once as validation data. After that, five models have been built based on the divisions of training data and the models were applied to test data. The reported results are based on performance of classifier on test set.
Two target components have been used in these experiments (an air suspension and a powertrain component), since different components have distinctively different degradation times and mechanisms. Other parameters, such as the size of the prediction window (which corresponds to the lead time needed to schedule a workshop visit) also depend on business requirements that vary across different components. Therefore, the same setting of parameters cannot be applied for different components in deployment and it is vital to examine carefully which setting is more appropriate for each component.
Figures 15 and 16, as well as 17 and 18 show the results for different experimental settings: aggregated and individual DTC levels, respectively, for powertrain and air suspension components. The subplots (a-d) demonstrate different categories of extracted features, in particular how area under ROC curve (AUC) changes depending on different settings for prediction and observation windows.
Generally, when using the base count features on DTC data suggests that smaller size of observation window leads to better AUC; this is more pronounced in aggregated level, but also shows in individual level. One interpretation of this pattern is that most relevant DTCs occur close to failure date, therefore enlarging size of observation window will reduce the informativeness of the input data. On the other hand, combined count features give better result with larger observation window. This led to the hypothesis that by applying combined count features, we add more information to the model and summation of values from multiple DTC readings work better than a single reading from DTC. The large jump in AUC for combined count features, is because the number of instances is considerably smaller and the task of prediction is easier for classifier.
According to figure 15(c), defining more features by taking the reading from former sub-windows doesn't improve AUC. The lowest range of AUC values belongs to trend-based features. Figure15(d) suggests that trend-based features cannot play a significant rule in prediction of failure and there is no pattern between size of observation window and prediction window and accuracy of model applying this feature.
Comparison of results indicates that base count features in aggregated level

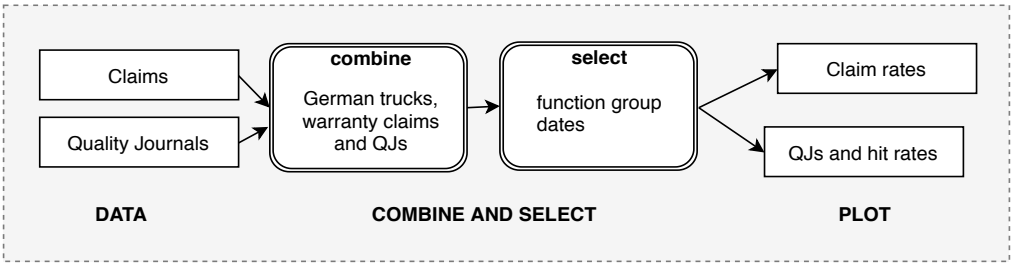



Figure 19: This figure shows the data processing pipeline. Quality journal and warranty claim data sets were combined to provide a historic overview of quality journal usage and claim trends.

work slightly better than individual level. Another observation is that the classification model trained with base count features or historical count features have better performance than trend based features. But the improvement in AUC for historical count features compared to based count features, which is provided by adding values from former sub-windows is non-significant.
This experiment and evaluation of early prediction of claims using DTC has resulted in a publication reported in Section 7.2.

## 6.4 Quality Journal Exploration

Volvo keeps information about warranty claims, reparations under warranty and so called quality journals. Quality journals contain a log of (potential) problems and resolutions related to a specific component (function group) for a certain vehicle class (long haul, short haul). These can be created for several reasons; an increase in warranty claims, security concerns, or a change in the production process.
We were provided with data sets containing historic information about warranty claims and quality journals for German trucks. The data sets covered about six years of data, from 2012 to 2018. This data was explored along a number of different paths as follows.

- claims on claim date (showing seasonality) or on production date (show-

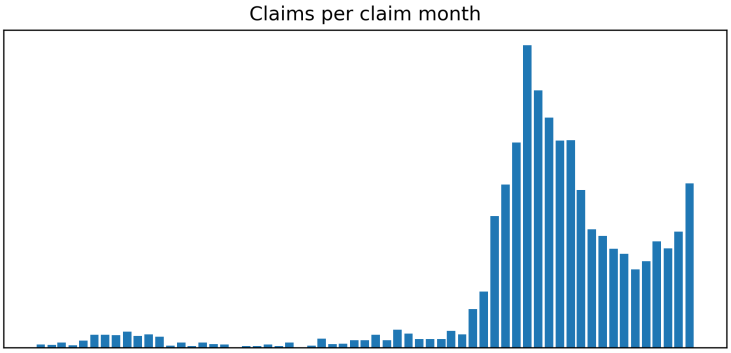



Figure 20: Number of claims per month for a single function group, narrowed down to German trucks, over a five year period.

ing production problems). Normalised on warranty volume or production volume.

- trends in claims, similar behaviour, ...
- claims for trucks in a certain production month claims over time for a certain function group
- claim rate and hit rate calculations
- success of a quality journal
- general statistics on quality journals

We illustrate the observations that can be done from the data by going through the claims for a certain function group in a five year period.
The following figure (Figure 20) shows the number of claims received per month for this function group. It shows that the number of claims started to increase suddenly.
Parallel to this we can show the number of claims for the function group related to the production month of the vehicles. Figure 21 shows that the increase in claims observed above are related to trucks produced after the first

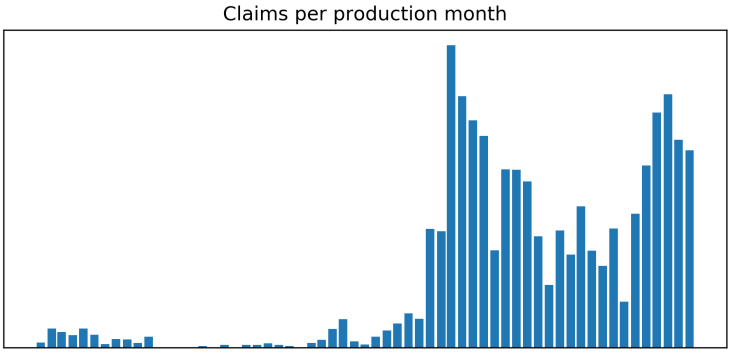



Figure 21: Number of claims per production month for a single function group, narrowed down to German trucks, over a five year period.

three years and onward. It typically takes a few months before production problems can be seen in the monthly claim rates. It takes some time before a vehicle has left the factory, has been taken into production and the fault has manifested itself.
For each production month, we can follow the produced trucks over their 24 (or 12) month warranty period, and plot the resulting claims in each (warranty) month. Figure 22 shows this for trucks produced in a one year period. As in the previous plot, the data is narrowed down to German trucks and for the same function group. In this plot, we see that until the end of the year, there are a few claims following each production month, and the bulk of the claims falls in the first 15 months after production. In the end, we see a larger number of claims starting from five months after production.

### Quality Journals

We also had access to the quality journals related to this, and other, function groups. Using the information contained in the quality journal, we can overlay several key data points on the plots showing the number of claims per production month. The following figure shows an example. In Figure 23 the start and end dates of one of the quality journals related to the function group are shown. The long horizontal light blue line in the middle of the plot

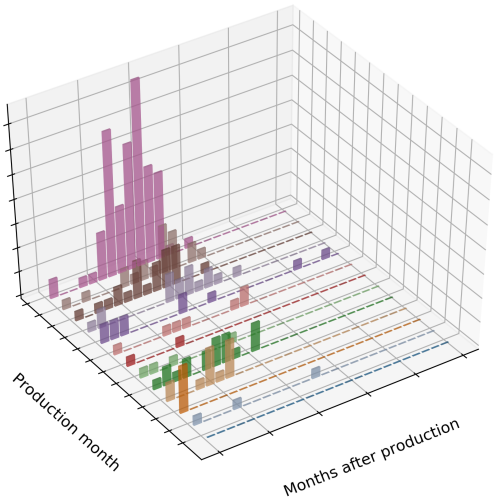



Figure 22: 3D plot showing the claims in the 24 months following a number of production months. The last production month in the plot, in purple, shows a significantly different pattern compared to the 12 months preceding it.

shows the in which months the quality journal was active. The two smaller blue lines on the left and on the right show the three months period before and after the quality journal was opened. If we look at the three months before the quality journal was opened, we see a rising number of claims related to the function group. We call the average number of claims in this period the *claim rate*. About halfway through the quality journal period, the number of claims is still high, but it starts to fall (note that the dip in August is probably caused by the low number of trucks produced in to the summer holiday).
The plots show an interesting picture, but note that they are normalised on the total number of trucks under warranty. While we had access to the number of claims tied to a certain function group for the trucks produced

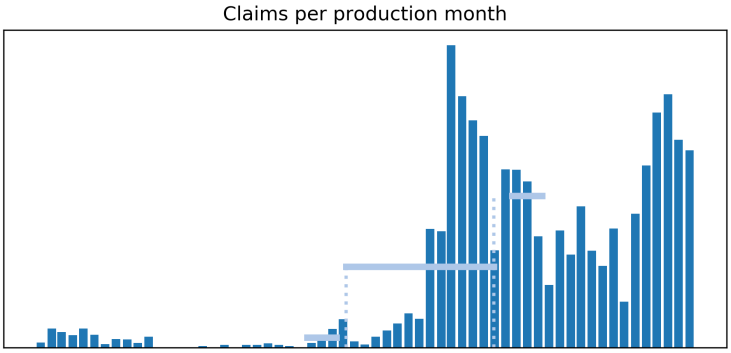



Figure 23: Claim rates during the time a quality journal was active.

on a certain production date, we did not have access to the trucks or truck models which were affected by the quality journals. Quality journals often target a specific model or sub-population of the fleet.
From the claim rates before and after opening a quality journal, we can calculate the *hit rate*. The hit rate is defined as the ratio between the claim rate after closing the quality journal and before opening it. A hit rate of less than one means the number of claims decreased, giving a measure of success to the quality journal. Calculating hit rates for certain time periods or for certain groups of components can give an insight into the quality journal operation.

### Quality Journal Statistics

Looking at the hit rates, we can get a sense of the quality journal usage; were they successful, how long were they open, is there a relation between the length (time) of the quality journal and its hit rate, or which function groups had successful quality journals. In the left hand side plot of Figure 24 we show the length (in days) a quality journal has been active on the day is was closed, aggregated by monthly average. On the right hand side, Figure 24 shows a histogram of the number of quality journals of a certain length (in days), up to one hundred days in length.

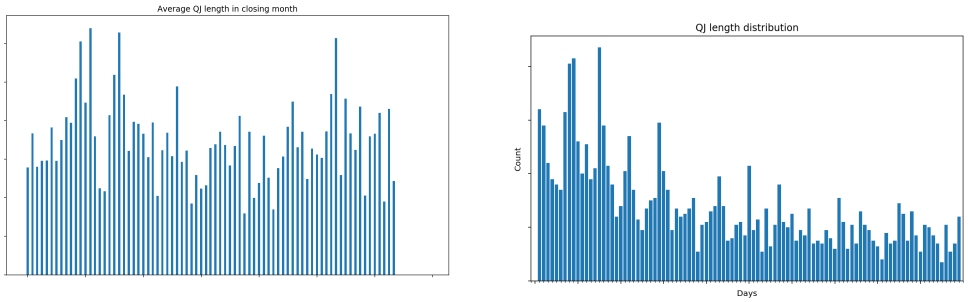



Figure 24: Left side: Length in days a quality journal has been active on the day it was closed, aggregated by monthly average. Right side: Quality journal length distribution

### Seasonality

Some components suffer to varying degrees under varying weather conditions. Batteries are known to perform worse under lower temperatures, whereas an air conditioning unit would typically only be used, and break down, when temperatures are higher. These trends can be observed from the dates the warranty claims were submitted. Figure 25 shows the warranty claims for a heating unit on the left. The number of claims follow a seasonal pattern with more claims in the winter months than in the summer months. On the right hand side, we see an auto correlation plot for the same data, showing correlation at the twelve months point. If we look at the claims related to production month in Figure 26, and the associated quality journals for this function group, we don't see the same clear seasonal pattern we observed in the left plot. We do however see an increase in claims related to trucks produced in a certain point of time. The plot also shows the associated quality journals, suggesting there was a problem which was detected, logged and fixed.

### Claim Patterns

Next, we look at the claims after production for one or more function groups. We can take the number of claims on each of the 24 warranty months after a certain production month, and collect them in a single plot. The left side

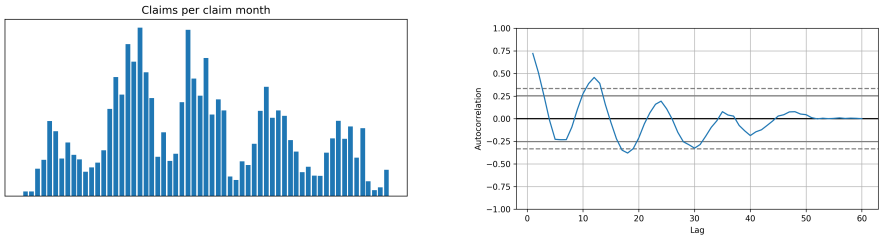



Figure 25: Claims on claim date. The left plot shows seasonal variations in claim rates. On the right, an auto-correlation plot for the data on the left is shown.

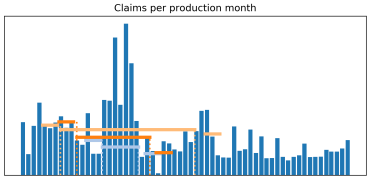



Figure 26: Claims related to production month are shown, together with several quality journals.

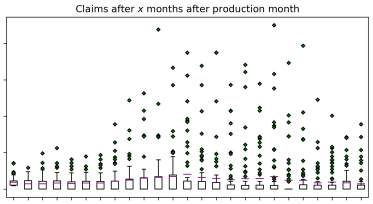



Figure 27: Distribution of warranty claims after $x$ months of production for a five year period, or 60 production months.

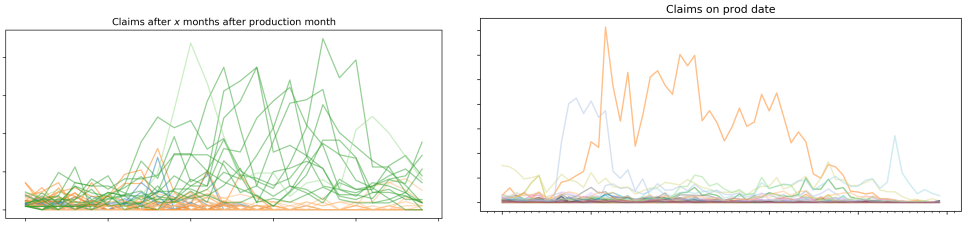



Figure 28: Left side: warranty claims after $x$ months of production for five production years. Right side: patterns of claims related to production date for all function groups related to electrical problems.

plot of Figure 28 shows these patterns for a certain function group. The plot shows all the production months since 2010-01 where we have data. In this plot, claims increase after the first six month of operation. Figure 27 shows a box-plot of this data.

We can create similar plots for multiple function groups. The left side plot of Figure 30 shows the claims in a five year period for all the function groups related to electrical problems. Most of the trends are low and flat (just a small number of claims each month), but three stand out. The first one has a high number of claims over most of the period. The next one has a problem at the end of the period, and the third component appears to have had a problem in the beginning.

The code to explore the claims, quality journals, claim and hit rates was used to create an interactive web based exploration system, one of the tools

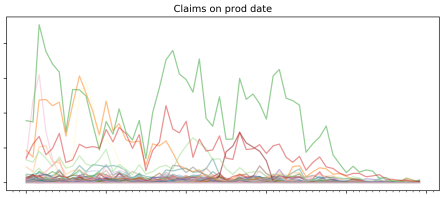



Figure 29: Claim patterns for all components belonging to the same main class.

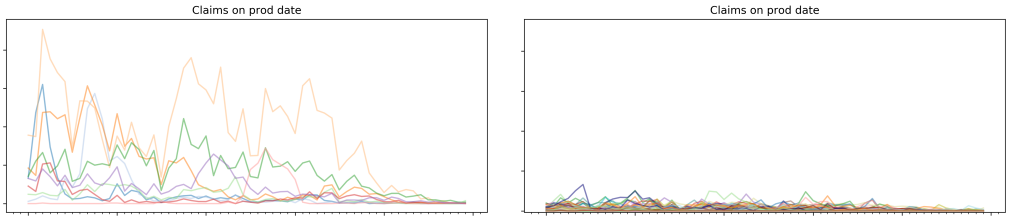



Figure 30: Left side: Outliers. Right side: claims without outliers.

delivered in the ARISE project.

### Outliers

We can apply clustering algorithms or other heuristics to detect outliers in the above plots. The following plot, Figure 29, shows the claim patterns for all the function groups related to a certain main class. Using different clustering algorithms, we get nine outliers out of a total of more than one hundred function groups.

## 6.5 Detecting Sub-optimal Vehicle Configurations

Trucks are highly flexible products, aimed to serve a broad variation of customer needs. There are hundreds of options available when selling and configuring a truck and the number of possible combinations is almost infinitive. A correctly configured truck is crucial to give the customer an attractive transport solution when it comes to fuel consumption, load capacity, driver

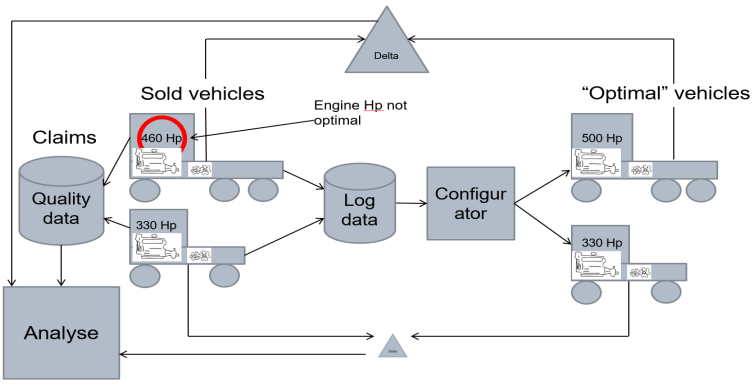



Figure 31: The conceptual view of the tool.

appeal, etc. Add body and trailers, that also have to correspond to the customer needs and to each other, and the complexity goes up even further. A mismatch between truck configuration and actual usage can lead to low productivity and customer dissatisfaction but also to quality problems. These kind of issues can be difficult to identify, as they are not originating from any individual component's quality deficiency, and it requires good insight into the customer's actual usage to be able to identify them.
In ARISE we have developed a tool – see Figure 31 – that can be used as a platform to analyse these kind of problems. A configurator takes vehicle log data and generates an 'optimal' configuration. At this stage, this is only possible for a limited number of vehicle part, i.e., the drive line with engine, gearbox and rear axle ratio.

# 7 Dissemination and publications

## 7.1 Dissemination

Multiple Volvo internal workshops and project presentations have been held during the project to discuss the progress of the work. The final results are also going to be presented at the Volvo Tech exhibition event in Göteborg. The date for this exhibition has not yet been confirmed due to the availability and capacity of the venue. At this event, we will present the results obtained in ARISE including tools, frameworks, data analysis, etc. in the form of posters and interactive discussions.

As a part of the ARISE result, we will also give a presentation at the EPIA Conference on Artificial Intelligence in September 2019. EPIA is a well-established European conference in the field of AI.

In ARISE, we contributed to the development of multiple tools and frameworks for various purposes as follows:

- Forecasting claim rates using LVD data (Section 6.2) was a prove of concept that information about usage in addition to the past claim information can improve the prediction of claim rates. Currently, Volvo is investigating the possibility of integrating some of the LVD feature in the QRAFT platform. Moreover, similar feature extraction methods have been used by the experts in Q&CS.

- We have shown in Section 6.3 that certain occurrences of Diagnostic Trouble Codes (DTCs) are indicators of faults appearing on components in given Function Groups (FGRP). This knowledge enables the experts to distinguish between accidental appearance of DTCs and root cause identification through DTCs. This algorithm has been implemented and integrated in a Volvo quality system called QRAFT in a way that the expert can search for correlations between DTCs and function groups.

- Quality Journal analysis, Section 6.4, provided a measure to higher level management working at Q&CS and component responsible for calculating the effectiveness of quality journals. This measure takes into account resources, Quality journal duration as well as claim rates before and after deployment of the solution.

- Finally, we have developed a tool/platform for proposing vehicle configuration based on customer needs within the hundreds of options available. This is going to matter–for customers–when it comes to fuel consumption, load capacity, driver appeal, etc. This tool is a prototype which has a potential to be used within the Volvo sales departments.

## 7.2 Publications

The ARISE project has led to the following publications and master thesis that corroborate the objectives defined in Section 5.

- Published:
  - Parivash Pirasteh, Sławomir Nowaczyk, Sepideh Pashami, Magnus Löwenadler, Klas Thunberg, Henrik Ydreskog, and Peter Berck "Interactive feature extraction for diagnostic trouble codes in predictive maintenance", Proceedings of the Workshop on Interactive Data Mining (WIDM 19).
  - Sławomir Nowaczyk, Anita Sant'Anna, Ece Calikus, Yuantao Fan "Monitoring equipment operation through model and event discovery", Intelligent Data Engineering and Automated Learning IDEAL 2018, 19th International Conference, Madrid, Spain, November 2123, 2018.
  - Mohamed-Rafik Bouguelia, Sławomir Nowaczyk, Amir H. Payberah "An adaptive algorithm for anomaly and novelty detection in evolving data streams", 2018. Data mining and knowledge discovery. 32(6), pp. 1597-1633
  - Thorsteinn Rögnvaldsson, Sławomir Nowaczyk, Stefan Byttner, Rune Prytz and Magnus Svensson "Self-monitoring for maintenance of vehicle fleets", 2018. Data mining and knowledge discovery. 32(2), pp. 344-384
  - Sepideh Pashami, Anders Holst, Juhee Bae, Sławomir Nowaczyk, "Causal discovery using clusters from observational data", FAIM'18 Workshop on CausalML, Stockholm, Sweden, July 15, 2018.
  - Evaldas Vaiciukynas, Matej Ulicny, Sepideh Pashami, Sławomir Nowaczyk "Learning Low-Dimensional Representation of Bivariate Histogram Data", 2018. IEEE transactions on intelligent transportation systems. 19(11), pp. 3723-3735
- Accepted:
  - Reza Khoshkangini, Sepideh Pashami and Sławomir Nowaczyk, "Warranty Claim Rate Prediction using Logged Vehicle Data", 19th Portuguese Conference on Artificial Intelligence, EPIA 2019, Proceedings, Springer.
- Submitted:

  - Reza Khoshkangini, Sepideh Pashami, Peter Berck and Sławomir Nowaczyk, "Prediction of Field Reliability Deviation from Logged Vehicle Data", 19th IEEE International Conference on Data Mining (ICDM19).

- In progress:

  - Reza Khoshkangini, Peyman Mashadi, Peter Berck, Saeed Gholami Shahbandi, Sepideh Pashami, Sławomir Nowaczyk, Tobias Nicklasson, "Multiple Machine Learning Approaches for Claim Rate Forecasting", to be submitted
  - Fredrik Johansson, Oskar Dahl, Reza Khoshkangini, Sepideh Pashami and Sławomir Nowaczyk, "Understanding Association Between LVD and Vehicle Configuration Parameters – Using Clustering and Rule-Based Machine Learning", to be submitted.

- Master Thesis

  - Fredrik Johansson and Oskar Dahl, Understanding usage of Volvo trucks", MSc Thesis, defended June 2018.

# 8 Conclusion and future research

A practical way to avoid quality issues in vehicles is to predict the possibility of such issues ahead of time. ARISE provided various machine learning approaches suited for the early detection of quality issues by utilising and integrating multiple data sources.
In this project, we have shown how multi-structural approaches can be used for the early detection of component failures, by exploiting data resources such as claim data, LVD and the integration of them. These result in effective and reliable pipelines to support the predictive maintenance strategy to plan before quality issues happen. This is beneficial for both manufacturer and customer in terms of cost and safety, respectively.
Software was developed to facilitate the exploration of the quality journals and warranty claims. In this system the data could be updated periodically to provide the current status of customer and warranty operations. Knowledge about quality journal length and effectiveness can be used to monitor ongoing journals and flag possible issues with open journals. These issues could for example be related to the time the journal has been open.
With respect to the claim rate predictions, we envisaged (a) integration with the aforementioned software, and (b) the construction of an early warning system for unexpected claim rates. Integration with the software could provide an exploration and dashboard system, where historic and current claim rate information is available, augmented with a warning system flagging claim rates which rise faster than expected.
Prediction on individual chassis could be used to schedule the affected vehicles for extra inspection. By identifying potential problems in time, better uptime for the customer can be provided.
Volvo is collecting large amounts of data for the purpose of a better understanding of how their products are used, to improve their products and e.g. design more optimised maintenance programs. We are planning to continue the ongoing collaboration with the Knowledge Foundation (KKS) profile+ project called CAISR+ by focusing on predictive maintenance with data and machine learning techniques. Since in ARISE, we have reached the high technology readiness levels (TRLs), we would like to focus on the industrial impact of the research in the next project.

# 9 Participating parties and contact person



For more information, please contact:
Claes Pihl, Claes.Pihl@volvo.com
Sławomir Nowaczyk, Slawomir.Nowaczyk@hh.se

# References

[Bradley, 1997] Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159.

[Corbu et al., 2008] Corbu, D., Chukova, S., and O'Sullivan, J. (2008). Product warranty: modelling with 2d-renewal process. *International Journal of Reliability and Safety*, 2(3):209–220.

[Fritz et al., 2012] Fritz, C. O., Morris, P. E., and Richler, J. J. (2012). Effect size estimates: current use, calculations, and interpretation. *Journal of experimental psychology: General*, 141(1):2.

[Gehan, 1965] Gehan, E. A. (1965). A generalized wilcoxon test for comparing arbitrarily singly-censored samples. *Biometrika*, 52(1-2):203–224.

[Kalbfleisch et al., 1991] Kalbfleisch, J., Lawless, J., and Robinson, J. (1991). Methods for the analysis and prediction of warranty claims. *Technometrics*, 33(3):273–285.

[Karim and Suzuki, 2005] Karim, R. and Suzuki, K. (2005). Analysis of warranty claim data: a literature review. *International Journal of Quality & Reliability Management*, 22(7):667–686.

[Kleyner and Sanborn, 2008] Kleyner, A. and Sanborn, K. (2008). Modelling automotive warranty claims with build-to-sale data uncertainty. *International Journal of Reliability and Safety*, 2(3):179–189.

[Nowaczyk et al., 2013] Nowaczyk, S., Prytz, R., Rögnvaldsson, T., and Byttner, S. (2013). Towards a machine learning algorithm for predicting truck compressor failures using logged vehicle data. In *12th Scandinavian Conference on Artificial Intelligence, Aalborg, Denmark, November 20–22, 2013*, pages 205–214. IOS Press.

[Prytz et al., 2015] Prytz, R., Nowaczyk, S., Rögnvaldsson, T., and Byttner, S. (2015). Predicting the need for vehicle compressor repairs using maintenance records and logged vehicle data. *Engineering applications of artificial intelligence*, 41:139–150.

[Steinberg and Colla, 2009] Steinberg, D. and Colla, P. (2009). Cart: classification and regression trees. *The top ten algorithms in data mining*, 9:179.