UPPSALA
UNIVERSITET

UPTEC STS 15023
Examensarbete 30 hp
Juni 2015

Finding Patterns in Vehicle Diagnostic Trouble Codes

A data mining study applying associative classification

Moa Fransson
Lisa Fåhraeus

UPPSALA
UNIVERSITET

Teknisk- naturvetenskaplig fakultet
UTH-enheten

Besöksadress:
Ångströmlaboratoriet
Lägerhyddsvägen 1
Hus 4, Plan 0

Postadress:
Box 536
751 21 Uppsala

Telefon:
018 – 471 30 03

Telefax:
018 – 471 30 00

Hemsida:
http://www.teknat.uu.se/student

Abstract

Finding Patterns in Vehicle Diagnostic Trouble Codes

Moa Fransson and Lisa Fåhraeus

In Scania vehicles, Diagnostic Trouble Codes (DTCs) are collected while driving, later on loaded into a central database when visiting a workshop. These DTCs are statistically used to analyse vehicles' health statuses, which is why correctness in data is desirable. In workshops DTCs can however occur due to work and tests. Nevertheless are they loaded into the database without any notification. In order to perform an accurate analysis of the vehicle health status it would be desirable if such DTCs could be found and removed. The thesis has examined if this is possible by searching for patterns in DTCs, indicating whether the DTCs are generated in a workshop or not. Due to its easy interpretable outcome an Associative Classification method was used with the aim of categorising data. The classifier was built applying well-known algorithms and then two classification algorithms were developed to fit the data structure when labelling new data. The final classifier performed with an accuracy above 80 percent where no distinctive differences between the two algorithms could be found. Hardly 50 percent of all workshop DTCs were however found. The conclusion is that either do patterns in workshop DTCs only occur in 50 percent of the cases, or the classifier can only detect 50 percent of them. The patterns found could confirm previous knowledge regarding workshop generated DTCs as well as provide Scania with new information.

Handledare: Ann Lindqvist
Ämnesgranskare: Matteo Magnani
Examinator: Elísabet Andrésdóttir
ISSN: 1650-8319, UPTEC STS 15023

Populärvetenskaplig Sammanfattning

Felmeddelanden i mjukvara uppstår för att varna användaren om problem. I fordon kallas de felkoder och ges av fordonets elektriska styrenheter. Idag blir fordon allt mer komplexa, näst intill självständiga, och är därför starkt beroende av att alla styrenheter kommunicerar med varandra. Kunskap om felkoder är därför avgörande för få fordonet att fungera. Diagnostiska felkoder infördes på Scania för att underlätta underhåll och reparationer i verkstäder. Koderna sparas i lastbilen under körning och laddas sedan upp i en central databas när bilen tas in på verkstad. I och med att databasen växer blir data miningmetoder applicerbara. I denna uppsats har sådana metoder tillämpats för att söka efter mönster i felkoder insamlade från Scanias lastbilar.

Med tiden har felkoder blivit en betydande indikator på ett fordons hälsostatus och statistiker på Scania arbetar kontinuerligt med att analysera koderna för att upptäcka avvikande beteende hos lastbilarna. Förhoppningsvis kan då fel upptäckas innan en större population av lastbilsflottan drabbas. Det har däremot noterats att ett flertal av felkoderna uppstår på verkstad till följd av arbete och tester. Dessa beror då inte på direkta fel i fordonet och skulle med fördel kunna sorteras bort från data för att ge en mer korrekt bild av verkligheten. Tyvärr är informationen bristande om vilka dessa är och varför de uppstår. I uppsatsen har det därför undersökts om mönster i felkoder finns för att kunna flagga dem som verkstadsgenererade. Teorin som tillämpats bygger på en klassificerare där regler genererats av typen $\{x_1, x_2, x_3\}$ ger *klass A* vilket i studiens fall skrivs på formen $\{Felkod_1, Felkod_2, Felkod_3\}$ ger klassen *Verkstad* eller *InteVerkstad*.

Verkstadsmönster gick att urskilja genom att bygga och testa en sådan klassificerare. Däremot presterade klassificeraren inte så pass bra att den i nuvarande form kan implementeras i Scanias verksamhet. Mönster i verkstadsfelkoder hittades i ca 50 procent av fallen. Huruvida detta beror på om mönster endast uppträder i hälften av verkstadsfelkoderna eller om modellen är otillräcklig kvarstår för framtida studier att avgöra.

Även en kvalitativ studie gjordes av de regler som hittades. Intressant visade sig då vara att klassificeraren lyckades hitta de av Scania redan kända verkstadsgenererade felkoderna. Dessa uppträdde dessutom i mönster tillsammans med nya felkoder som tidigare inte varit kända som verkstadsfelkoder. Fortsättningsvis kunde den kvalitativa studien urskilja att endast ett fåtal av de styrenheter som användes vid analysen fanns med i de funna verkstadsmönstren.

Slutligen ska nämnas att samtliga koder som upptäcktes i verkstadsmönster till stor del var timeout-koder eller förlorade kommunikationsmeddelanden. De misstänks ofta uppkomma till följd av uppdatering eller urkoppling av styrenheter. Den kvalitativa studien indikerade att en större beaktning av dem i framtiden är önskvärd.

Preface

This master thesis was carried out at Scania, ordered by a data analysis group. There has been a partitioning of the work concerning the development of the two algorithms. Moa Fransson has been responsible for the algorithm in *ClassifierOne* and Lisa Fåhraeus for the algorithm in *ClassifierTwo*. Furthermore has Moa Fransson modelled and tested on the S8 sample and Lisa Fåhraeus on the S6 sample. However, both authors have been involved in all parts of the project, from designing theory and method to modelling and testing.

The authors wish to express their gratitude to Scania and in particular the supervisor Ann Lindqvist who has provided guidance and constructive criticisms during the project.

Contents

Abbreviations

Controller Area Network (CAN) - Bus system for in-vehicle communication

Class Association Rule (CAR) - Rule generated by Associative Classifier, including items in premises and class as conclusion

Diagnostic Database (DD) - Database where Diagnostic Trouble Codes are stored

Diagnostic Trouble Code (**DTC)** - Error message from vehicle

Electrical Control Unit (**ECU)** - Embedded system consisting of hardware and software controlling different parts of the vehicle

Fleet Management System (**FMS)** - Database to which GPS coordinates are loaded. Can be used to collect information about a vehicles' workshop visits

1 Introduction

The purpose of error messages produced by software is to inform about problems in execution. In vehicles, error messages are called trouble codes. They are generated by the Electrical Control Units (ECUs) which are embedded systems consisting of hardware and software controlling different parts of the vehicle.

Today vehicles are getting more and more complex, becoming close to autonomous, making them heavily dependent on all ECUs to communicate within the vehicle. Knowledge about the trouble codes is therefore crucial in order to make the vehicles function well.

Primarily Diagnostic Trouble Codes (DTCs) were implemented in Scania vehicles to facilitate maintenance and truck reparations by providing help to mechanics while in workshops. The trouble codes are logged in the trucks' ECUs while driving and later loaded into a central database when visiting workshops, enabelling information to be further analysed. This has resulted in a large database containing trouble codes from the main part of all vehicles produced since 2004. As the database grows, data mining methods become applicable. In this thesis, such methods will be used to search for patterns in DTCs collected from Scania trucks.

1.1 Problem Description

Over time, DTCs have become important indicators of vehicles' health statuses. Statisticians at Scania try to find early changes in DTCs' occurrences as indicators of abnormal behavior. This will hopefully help find defects in

trucks, making it possible to act before problems strike a larger part of the vehicle population.

Some of the more frequent DTCs are worrying if they occur while driving, but statisticians at Scania have learned that they are sometimes generated at a workshop, probably due to work done by mechanics. It would be preferable to remove those DTCs since the data would then better describe the health state in the vehicles. There is however no way to know for sure if a truck is in a workshop or not when a DTC occurs. Still, some of these DTCs are removed from the data. Since the trouble codes are removed first after statisticians have considered them as workshop generated, this way of working requires good qualitative knowledge about DTCs. As a consequence comes the risk of the evaluation becoming dependent on the business knowledge possessed by the people performing the analysis. Instead, a data driven process would be preferable where the workshop provoked DTCs could be removed without using qualitative information. This would decrease the dependency on the knowledge of single individuals. One possible technique is to find patterns in the workshop generated DTCs and then automatically flag such patterns as "generated in workshop". Not only would the business knowledge then no longer be required, but new patterns of DTCs generated in workshops could possibly also be found. This thesis will examine if such patterns can be found so that DTCs in the future can be flagged before the data is statistically analysed.

1.2 Objectives

The main purpose of the thesis is to evaluate if patterns can be found in workshop generated DTCs. Established data mining methods will be applied together with two modified algorithms, developed to fit the data structure. The analysis will be both quantitative and qualitative, looking for patterns holding DTCs already known to be frequently occurring at workshops. Great consideration will be given to the interpretability of the results in order to provide Scania with new knowledge about workshop generated DTCs. The following objectives will be examined:

- Can already known workshop DTCs be found in patterns?
- Can new workshop DTCs be found in patterns? If so, what can be given by a qualitative analysis of those?

1.3 Thesis Outline

The thesis is arranged as follows:

Chapter 2 provides a description of the data to give a complete understanding of the context in which the process is taking place. Two data sources will be explained together with data characteristics, later underpinning the choice and design of method.

Chapter 3 introduces the reader to data mining methods, focusing on Associative Classification, which is later on used in the modeling. The first sections discuss and describe the choice of classifier. In chapter *3.2.3 Modified Classification Algorithms*, the two developed classification algorithms will be explained in detail.

Chapter 4 describes the decisions made concerning methodology, data sampling and planning of tests.

Chapter 5 is devoted to the results of the Associative Classification, separated in sections based on data samples and classification algorithms.

Chapter 6 analyses the results and relate them to the objectives and the problem description.

Chapter 7 summarises the results and answers the objectives set out in this thesis.

Chapter 8 discusses the requirements for a possible implementation of the results as well as future work.

2 Data Description

The database where DTCs are stored when a vehicle is connected to the diagnostic software tool is called the Diagnostic Database, further on denoted as DD. Every Scania vehicle that has visited a Scania workshop and been connected to the diagnostic software tool is in DD.

In a truck, every ECU cast DTCs unique to that version of ECU, making it important to only use vehicles with the same ECU-configuration when looking for patterns. There are different types of ECUs in Scania vehicles and each ECU exists in multiple versions, today in a total of 71 unique ECU versions. At the moment, only new ECUs have the technical requirements to read and store GPS coordinates. These vehicles can therefore hypothetically label each DTC as set in a workshop or not by matching trucks' GPS coordinates with map data. This workshop label is not yet implemented in DD, but is planned to be in the future. A remaining problem is that more than 75 percent of the vehicles do not have the GPS functionality, and probably never will. Therefore it is important to evaluate other methods of categorising DTCs as being set in a workshop or not.

A DTC only has a few attributes where the readout ID is the most important. A readout is a "virtual photo" of the vehicle's status, saved in DD every time a vehicle is connected to the diagnostic software in a workshop. It holds information about all DTCs that have occurred since the latest readout. *In this thesis, the relevant attributes of a DTC are:*

Readout ID: A unique number for every readout. Each readout holds many or zero (unusual) DTCs.

DTC_ECU: An attribute unique for each DTC, consists of a code and the ECU version name.

Chassis ID: A number unique for each vehicle, indicating which vehicle the DTC came from.

Chassis assembly period: Period when the vehicle was produced.

Timestamp: The most recent date and time when a DTC occurred. No other historical data about the DTC is available.

Export date: The date and time of the readout.

Input source: Specifies the method used to load data into the database. *A readout can have five different types of input sources. These are:*

> **Automatically:** This type of readout is made automatically by the system as soon as a vehicle is connected in a workshop. After this readout all DTCs within the vehicle should be manually deleted, leaving the vehicle's DTC history "clean".
>
> **Automatically Override:** This is a readout done when an ECU is updated. Should be considered as a manual readout.
>
> **SDP3:** After an automatic readout is done in a workshop, the mechanics can perform another readout by request without disconnecting the vehicle from the system. This readout is then labelled SDP3.
>
> **GUI:** A readout done over USB.
>
> **RPD:** A readout made over the GSM-network when the vehicle is not in workshop.

There is yet another source of data. This is the Fleet Management System (FMS) which is a service provided by Scania where vehicles are connected to the mobile telephone network, sending information to FMS every ten

minutes. This database can therefore give more precise information than DD. The service is free of charge during the first year for all vehicles with an assembly period after 2012. Some of the information, such as position of the vehicle, is accessible through a web browser. In this way hauliers can extract information about their vehicles in order to supervise and plan trips. However, the database covers only 12 percent of the vehicles in DD. This data source will later be used in combination with DD data in order to extract new valuable information. The process is further explained in chapter *4 Method*.

3 Context of study

Progress in data technology in the past years has led to increased possibilities of storing data. Due to this development, the new concept Data Mining has been established. The term refers to the exploration of large data sets with the purpose of discovering hidden relationships and is therefore sometimes called Knowledge Discovery and Data Mining (KDD) [8]. One technology covered by the concept is classification where data is categorised according to its attributes [18]. This will be applied in order to examine the objectives of the thesis.

3.1 Discussion of Classifier

A classifier is a supervised learning method where already classed data is used to train a model [11]. The choice to use a classifier was made simply because if succeeding in classifying, it would mean that patterns exist in the data. In this project labelled training data was available and the classes were predefined as Workshop and NotWorkshop, enabling the use of such a method.

There are many approaches to supervised learning but some, such as neural networks and support vector machines, use what is often called a "black boxing" approach [4, 6]. The term refers to the models as "black boxes" where the user inserts data and gets an output, not being fully aware of how the model operates due to its high complexity [16]. Rule-based classification, on the other hand, is especially preferable when the classifier is to be later analysed and interpreted [9], which is the case of this project. It is frequently

applied in real world contexts due to its easy understandable outcome. Rule-based classifiers can be used to gain new knowledge from rules, expressed in an "if... then" manner. The if-condition refers to a data set and the then-consequence refers to a class [9]. This is why the rule-based approach was selected in this project, to facilitate the qualitative study of the classifier and the possible patterns found.

Two methods for rule extraction are Decision Tree rule extraction and Associative Classification. Even though both generate rules, they deviate from each other in some aspects. While Associative Classification is a way of applying Association Rules to classification, Decision Trees perform a greedy search to find patterns [10]. The method is therefore fast but research shows that Associative Classifiers have in some cases performed with better accuracy [10, 15, 20, 22]. Extracting rules with these methods can however lead to overfitting [19]. The concept refers to a model training too well, adjusting to even the smallest deviations in data. This can include outliers, making the model not generalisable for new data [18].

In this project, modeling with negative rules was delimited. Such rules are expressed on the form *"If x_1 and not x_2 then Class C"* [23]. To create negative rules, each item must be expressed in its presence or absence. This was problematic in the case of DTCs since neither is there a record of all designed DTCs, nor have all designed DTCs yet been activated in Scania vehicles. Items cannot be expressed solely in their presence when using a Decision Tree, in contrast to an Associative Classifier. Therefore Decision Trees were left outside the framework of this thesis and the modeling was performed using only an Associative Classification algorithm.

3.2 Associative Classification

Associative Classification builds upon Association Rules. The rules are used to find associations in a group of items [17]. A common application of Association Rules is Market Basket Analysis within business retail [1]. By using

this method, questions such as "*What other items are likely to be bought if the customer buys peanut butter and jelly?*" can be answered. The results can be applied when deploying products in stores or when planning customized advertising. A rule is written on the form:

$$\{Peanut\,butter, Jelly\} \rightarrow \{Toast\} \tag{1}$$

It is read as *"if a person buys peanut butter and jelly, that person is also likely to buy toast"* [10]. To create Association Rules a data set, D, with multiple transactions is required. Each transaction, d_i, contains one or more items, $x_1, x_2, x_3, \ldots, x_n$. In the example of shopping the items are products and the transaction is the shopping basket. To define how frequent an itemset occurs in the dataset support is introduced. The support of itemset x_1, x_2, x_3, is the percentage of all transactions in D containing x_1, x_2 and x_3 [21].

$$Support(\{x_1, x_2, x_3\}) = P(x_1, x_2, x_3) \tag{2}$$

When constructing the rules itemsets are divided to include a premise, the left hand side in Equation 1, and a conclusion, the right hand side in Equation 1. A quality measure for rules is confidence which in the case of $x_1, x_2, \rightarrow x_3$ is referring to the proportion of rules containing x_1, x_2 in the premise and x_3 in the conclusion [17].

$$Confidence(\{x_1, x_2\} \rightarrow \{x_3\}) = P(x_3|x_1, x_2) \tag{3}$$

When generating rules, thresholds for support and confidence are initially set by the user in order to choose only those rules representative for the data. Such constraints are called minimum support and minimum confidence, further on denoted min-supp and min-conf [14]. All itemsets fulfilling the min-supp constraint are frequent itemsets.

Confidence and support can sometimes be insufficient when studying Association Rules as they do not encounter any qualitative knowledge regarding the context of the case. In order to help analyse the results the subjective

measure *interestingness* is introduced [5]. Interestingness can be expressed in *unexpectedness* and *actionability*. Unexpectedness means that unexpected rules which contradict or surprise the user are interesting. Actionability implies if rules are useful or applicable to real cases. All rules can be explained in these two terms, being both, one or neither of them.

3.2.1 Class Association Rules

When building a classifier based on Association Rules, the same structure of the rule is used with the only difference that the conclusion will contain a class instead of an item, then called a Class Association Rule (CAR) [9].

$$\{x_1, x_2, x_3\} \rightarrow \{Class\ A\} \tag{4}$$

As mentioned earlier Associative Classification is a supervised learning method where already classed data is used to train a model. Before creating the rules each transaction is therefore labelled with a class. The procedure of creating the classifier starts by using the same approach as when creating Association Rules. According to the method Classification Based on Association rules (CBA) [15], the CARs should be checked towards each other to prevent copies of the same premises leading to different classes in the conclusion:

$$\{x_1, x_2, x_3\} \rightarrow \{Class\ A\} \tag{5}$$

$$\{x_1, x_2, x_3\} \rightarrow \{Class\ B\} \tag{6}$$

In this thesis a min-conf above 50 percent will be used, preventing such rules from being generated.

The Apriori algortihm was applied when generating CARs. Apriori uses an iterative approach that in each iteration, k, computes the possible number of itemsets containing k items [7]. In each iteration, itemsets with a support lower than the min-supp are removed and will not be part of any

further calculations. This can be done since if itemset d_i is not considered a frequent itemset, then any set containing d_i cannot be either. Consequently, this decreases the number of possible combinations of itemsets and therefore reduces the complexity when creating the rules.

3.2.2 Ranking and Pruning Rules

After having generated the rules two steps follow, ranking and pruning. They include refining the rules to ensure that the most correct and important are used first to class new data. Applied in this thesis are the Live and Let Live algorithms, further on denoted as L^3. First, the ranking algorithm is applied, ranking the CARs according to highest confidence, support and the numbers of items in the premises [3]. This means that the rule with the highest confidence is ranked first. If the confidences of two rules are equal, the rule with the highest support precedes the one with lower. If confidence and support would also be equal, the most specific rule[1] will be the highest ranked, see Algorithm 1 below. Using the L^3 algorithms will in this way lead to the most specific subset of a rule being ranked first [2].

Algoritm 1 L^3 Rule Ranking Method [2]

Given two rules, r_i and r_j, r_i precedes r_j if:

1. The confidence of r_i is larger than that of r_j
2. The confidence of r_i and r_j are equal, but the support of r_i is larger than that of r_j
3. The confidence and support of r_i and r_j are equal, but the premise of r_i contains more attributes than the premise of r_j

[1] A rule containing more items in its premise is more specific than a rule with fewer items in its premise.

When having ranked all rules, the rules are pruned with the L^3 pruning algorithm, see Algorithm 2. This is also called a Lazy approach to pruning since it discards those rules that only classify incorrectly. It means that if a rule correctly classifies at least one itemset it is saved as "used rules". If a rule is never matched in the pruning data, the rule cannot be considered as having classified neither correctly nor incorrectly. Therefore those rules are also saved, but as "spare rules" [2]. The final set of rules will first contain all used rules and then the set of spare rules.

Algoritm 2 L^3 Pruning Algorithm [2]

```
GenerateClassifier(rules, data)

while data not empty{

    for each d in data{
        covered = FALSE
        NR = number of rules
        r = first rule in rules
        while (covered == false & NR>0){
            if r covers d{
                r.dataClassified = r.dataClassified + d
                if d.class == r.class: r.right = ++
                else: r.wrong ++
                covered = TRUE
            }
            NR - -
            r = next rule from rules
        } delete d from data

    }

    for each r in rules{
        if r.wrong>0 and r.right == 0{
        delete r from rules
        data = data + r.dataClassified
        }

    }

}

for each r in rules {
    if r.right >0: usedRules = usedRules + r
    else: spareRules = spareRules + r

}
```

One drawback of the L^3 method is that very large classifiers are generated, which both tend to slow down the classifying process and increase the risk of overfitting [19].

3.2.3 Modified Classification Algorithms

In traditional Associative Classification, classifiers are used to class items based on their attributes, where the attributes are collected in a transaction d_i. This is done by searching through the list of ranked and pruned rules, labelling a new item with the same class as of the first matching rule [10]. In this thesis the main goal is not to class the entire transaction d_i, but rather the single items or subsets within d_i. This means that if a subset of a new unclassed transaction, d_i, matches the premise of a rule, only those items and not entire d_i will be classed. The remaining items in the transaction are further analysed to find any other matches between subsets and rules. This will continue until all items in the transaction are matched or until no other rule is matching. If there are items left in the transaction after having searched through all rules, these items are labelled as the default class.

To be able to class single items in a transaction, two different classification algorithms have been developed, explained in Algorithm 3 and Algorithm 4 as *ClassifierOne* and *ClassifierTwo*. In the first algorithm any subset of a transaction that is found to match a rule will instantly be removed and not be part of further matching with remaining rules. The rest of the items are either labelled with the same class as that of the first subset or labelled as any other class, depending on the class of the next matching rule.

Algoritm 3 ClassifierOne

```
ClassifierOne(Rules, Testdata)

for d in Testdata{
    i = 1

    while (i <= Rules.length & d.length > 0){
        set r to Rules[i]
        if (r covers d and those items not already classed){
            set d.items.class as r.class
        }
    i = i +1
    }

    if (unclassed d.items exists){
        set those d.items as default class

    }
}
```

One problematic aspect with this approach is that an already classed item in a transaction cannot be included in any further classification since it is already removed. When finding a rule that could have matched an itemset, including items that have already been classed, no consideration will be given to this rule. Instead other matching rules are prioritized. This is explained in Example 1.

Example 1

List of ranked and pruned rules

Rule 1: $\{x_1, x_2, x_3\} \rightarrow$ Class A

Rule 2: $\{x_1, x_2, x_3, x_4\} \rightarrow$ Class A

Rule 3: $\{x_4, x_5\} \rightarrow$ Class B

Items to classify:

Transaction d: $\{x_1, x_2, x_3, x_4, x_5\}$

According to ClassifierOne the following classification of the items in d will be made:

Rule 1 will class as following:

$\{x_1, x_2, x_3\} \rightarrow$ Class A

Leaving the remaining items $\{x_4, x_5\}$ in d to be further classed

Rule 2 will not class any items in d

Rule 3 will class the remaining items in d as follows:

$\{x_4, x_5\} \rightarrow$ Class B

Leaving Transaction d empty: {}

In the example, no consideration is given to Rule 2 since x_1, x_2, x_3 are removed from transaction d after being classed by Rule 1. This is done even though x_4 could have been included in Class A according to Rule 2. As argued in chapter *3.2.2 Ranking and Pruning Rules*, longer itemsets are preferred over a subset of that itemset. To handle this, ClassifierTwo was developed where items already classed are included in further classification as long as the class is the same. As seen in Algorithm 4, this is done by labelling subsets with

the class of the first matching rule instead of removing them. Applying this to the example above, Rule 1 and 2 will both be used, classing x_1, x_2, x_3, x_4 as Class A. Rule 3 will not be applied.

Algoritm 4 ClassifierTwo

```
ClassifierTwo(Rules, Testdata)

for d in Testdata{
    i = 1

    while (i <= Rules.length){
        set r to Rules[i]
        if (r covers d and those items not already classed as !r.class){
            set not already classed d.items.class as r.class
        }
        i = i+1
    }

    if (unclassed d.items exists){
        set those d.items as default class
    }

}
```

3.2.4 Validation

To evaluate the performance of a classifier, a Confusion Matrix [10], also known as a Contingency Table [17] is introduced, see Table 1. The matrix contains the results from a classified test data set. True Positive (TP) and False Positive (FP) represents the items classed as the Positive class by the model, also noted if they were classed correctly (True) or incorrectly (False). True Negative (TN) and False Negative (FN) represents the items classed as Negative, either correctly (True) or incorrectly (False). In this thesis, the

NotWorkshop class is the default Positive class, while the Workshop class is assigned the Negative class.

Table 1: Confusion Matrix [10]

		Predicted Class	
		Positive	Negative
Actual Class	Positive	TP	FN
	Negative	FP	TN

From the matrix, three measures can be calculated.
Accuracy - The overall measure of the performance of the model, referring to the rate of correct classifications.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

Sensitivity - The rate of correctly classed Positive values out of all Positive values.

$$Sensitivity = \frac{TP}{TP + FN} \tag{8}$$

Specificity - The rate of correctly classed Negative values out of all Negative values.

$$Specificity = \frac{TN}{TN + FP} \tag{9}$$

Workshop rules with high precision are preferred in this thesis, therefore another measure calculating the correctness in the class is introduced. It will here on after be called *negative correctness*, referring to the rate of correctly classified Workshop items out of all items classed as Workshop. The *negative correctness* is calculated as:

$$Negative\,Correctness = \frac{TN}{TN + FN} \tag{10}$$

One method to evaluate the performance of a classifier is to do *k-fold cross validation* [12]. The method is an iterative process where the data set is first

divided into k equally sized subsets. When training the model $k-1$ of the subsets are used, leaving the remaining sample for testing. This is repeated k times so that every subset will be the test set once. The performance of the model is the average of all k runs [13].

3.2.5 Summary of Associative Classification

The process of applying Associative Classification can be summoned up in a model, see Figure 1. First training data is used to generate the frequent itemsets, then rules are generated. The ranking and pruning will follow ensuring the precision of the rules. Finally test data will be used to evaluate the accuracy of the model.

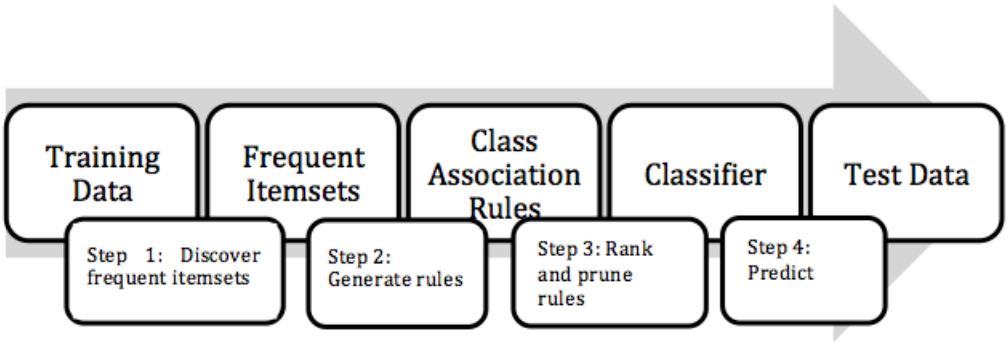

Figure 1: : Associative Classification steps [19]

4 Method

As mentioned in chapter *3 Context of Study*, supervised learning requires already labelled data in order to train and test the model. Therefore one more attribute, the class, had to be implemented. The attribute was either "Workshop" or "NotWorkshop", denoting if a DTC occurred while in a workshop or not. In FMS data, the GPS position of a vehicle can be extracted at the timestamp of a DTC. This can be matched with map data to detect if a vehicle was close to a workshop and consequently be set to either Workshop or NotWorkshop.

In DD data there is limited knowledge whether a vehicle is visiting a workshop or not. Some of the attributes are yet an indication that this would be the case. These attributes were used in a query to label DTCs from DD as Workshop or NotWorkshop. One of the attributes was the input source. According to the input sources explained in chapter *2 Data Description*, only DTCs with a readout labelled as Automatically, SDP3, or GUI can be considered as being set in a workshop. This was used as one of the constraints in the Workshop class query. Then only DTCs with a timestamp the same day as a readout export date were included. DTCs having a timestamp with the value NULL were also Workshop classed since such errors are likely to have been provoked by a mechanic when unplugging ECUs that set time and date. The NotWorkshop class was in the same way contructed using a query, selecting only those DTCs having a timestamp seven days before the readout. The constraints in the query used to get labelled training and testing data are summoned below.

Workshop	**Not Workshop**
Having a timestamp • The same day as the readout day or with a NULL value	Having a timestamp • At least seven days before the readout day
Readout type: • Automatic • SDP3 • GUI	No constraint on readout type

4.1 Sampling Criteria

When building and testing the model all sampled vehicles must have the same combinations of ECUs. Therefore, only vehicles sharing ECU configurations were extracted from DD and FMS data. Since the FMS database contains fewer vehicles than DD, this database was initially studied when selecting vehicle types. All unique ECU configurations were extracted and their occurrences in chassis counted. By doing this, the most frequent ECU configurations could be selected.

The data set was divided into two samples; one having an engine ECU version called S6 and the other called S8. These samples will further be denoted as the S6 and S8 samples. In the S6 sample the vehicles had eleven ECUs in common and in the S8 sample they had twelve. A time frame of two years for the assembly period was set. The decision was made with the truck lifecycle in consideration, where newer trucks are likelier to have a high contingency in workshop visits due to guaranties.

The same ECU configurations corresponding to the ones in FMS data

were selected when extracting chassis from the DD data. Vehicles with a mileage of less than 100 kilometers were excluded since those have not been driven enough kilometers to contribute with valuable information. Also, such vehicles might not have arrived to their first owner yet. All DTCs from the selected chassis were retrieved from the period June 2014 to January 2015.

4.2 The Final Data Set

When extracting data from the FMS and DD databases, the samples presented in Table 2 were received.

Table 2: Summary of the final data sets

Data set	Chassis	Readouts	DTCs	Workshop	NotWorkshop
FMS S6	7.728	9.661	27.009	7.267	19.742
FMS S8	7.207	9.112	25.228	6.293	18.935
ODP S6	8.772	26.905	80.481	16.582	53.899
ODP S8	8.389	15.261	97.382	15.965	81.421

4.3 Outline and Planning of Tests

Since FMS data contains fewer vehicles than DD data, modeling on DD data would be preferable. FMS data is however more precise due to the possibility of using GPS coordinates. As a first step to thoroughly evaluate the two developed algorithms, FMS data was used to build and test a model for those chassis types that are covered in both databases, see Figure 2. If the algorithms would be considered performing well, the same algorithms could be used in a second step to build a model with DD data and test on FMS data. To see if any significant differences from FMS data could be revealed, the interestingness of such generated rules could be analysed and compared to the results in the first step. If the performance of such a classifier is satisfying, a model built and tested on only DD data could as a final step

be developed. This model would potentially be a classifier scalable for all vehicle types covered in DD.

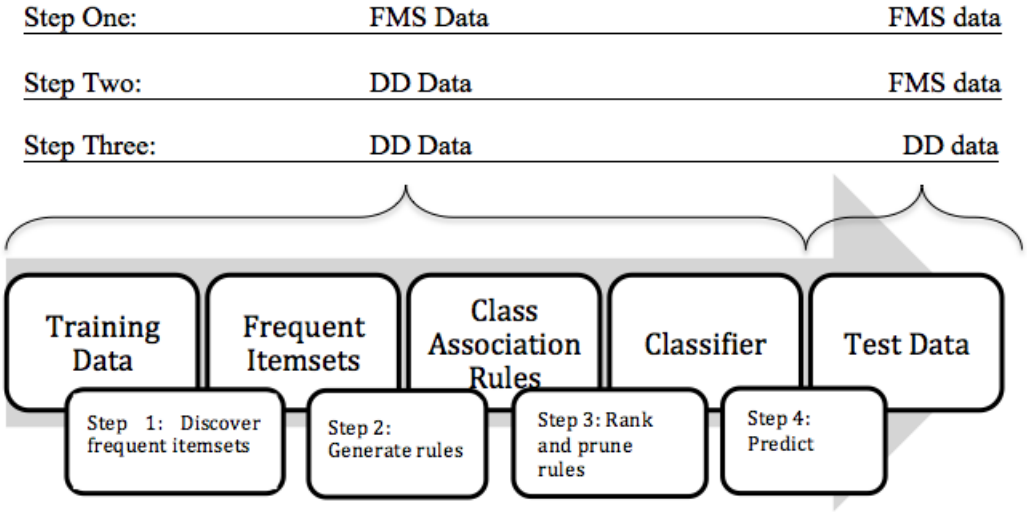

Figure 2: Planning of training and testing with FMS and DD data sets.

4.4 Quantitative Validation

In the first step, when using FMS data in training and testing, a k-fold cross validation was used to compute the Confusion Matrix and accuracy. Generally $k = 10$ is commonly used [12], but since the FMS data only contained 7.728 S6 chassis and 7.207 S8 chassis such separation of the data would have created too small test sets. Instead a value of $k = 5$ was chosen, meaning that the training sample included 80 percent of the data set, where 40 percent were used in the rule extraction and 40 percent in the pruning. The remaining 20 percent were used in the testing. The procedure was repeated five times to make sure that the samples were used in all three parts, twice in generating CARs, twice in pruning and once in testing. In the second step, a cross validation was not possible since the training and validation data came from different data sources. The DD data was instead randomly

divided into two equally sized samples, using one to generate CARs and the other to prune. The entire FMS sample was then used for testing.

4.5 Qualitative Validation

As stated earlier, statisticians at Scania use their business knowledge regarding Workshop DTCs to clean up the data. One important step in validating the model was therefore to see if the classifier could find these DTCs. If so, this would motivate further studies on the model. There are however only few DTCs known to be frequently occurring in workshops, even though there are suspicions that there are many more such DTCs to be found. Through interviews and internal documentation it was found that the DTCs presented in Table 3 often occur in workshops.

Table 3: DTC description

DTC Code	DTC Description	Source
000001_EBS	Timeout coordinator	Internal Documentation
000003_EBS	Timeout tachograph	Internal Documentation
000004_OPC	Timeout coordinator	Internal Documentation
000006_EBS	Error from engine ECU	Internal Documentation
000001_EMS*	CAN Message from coordinator timeout	Interview and Internal Documentation
000002_EMS*	CAN message from tachograph timeout	Interview and Internal Documentation
000003_EMS	Lost VGT activator	Interview and Internal Documentation

*DTCs often occurring together

Included in the qualitative study was a comparison between the patterns found in the classifier and the DTCs in Table 3. The patterns were also presented to employees at Scania working with DTCs. By doing this, the interestingness of different rules could be evaluated by taking expertise knowledge into consideration.

While classing new items in the test phase, a correct/incorrect count was introduced for each rule. This would indicate how many itemsets each rule classified correctly or incorrectly, giving an indication of which rules could be of certain interest.

4.6 Software

Scania's data was stored in SQL Server. SQL queries were used to retrieve and preprocess data from the two databases. The data was then imported to R, where all algorithms were implemented. R was primarily chosen due to being open source and having accessible online documentation.

5 Results

Presented in Tables 4, 5, 6 and 7 are the results from modeling and testing on FMS data. Every sample and classification algorithm is presented in separate tables. The best result in every column is bolded.

Table 4: ClassifierOne. Sample: FMS S6

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.0200	0.51	0.8083	0.3422	**0.9798**	0.8620
0.0150	0.51	0.8127	0.3595	0.9794	**0.8655**
0.0100	0.51	0.8156	0.3901	0.9721	0.8375
0.0100	0.60	0.8182	0.4026	0.9712	0.8371
0.0100	0.70	0.8168	0.3924	0.9730	0.8422
0.0050	0.51	0.8233	0.4435	0.9631	0.8155
0.0050	0.60	0.8220	0.4456	0.9605	0.8060
0.0050	0.70	0.8264	0.4209	0.9674	0.8262
0.0025	0.51	**0.8294**	0.4709	0.9613	0.8174
0.0025	0.60	0.8254	**0.4763**	0.9539	0.7918
0.0025	0.70	0.8218	0.4156	0.9713	0.8423

Table 5: ClassifierTwo. Sample: FMS S6

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.0200	0.51	0.8085	0.3418	**0.9802**	0.8640
0.0150	0.51	0.8126	0.3597	0.9793	**0.8649**
0.0100	0.51	0.8176	0.4029	0.9702	0.8327
0.0100	0.60	0.8165	0.3866	0.9747	0.8489
0.0100	0.70	0.8158	0.3776	0.9771	0.8583
0.0050	0.51	0.8237	0.4472	0.9623	0.8135
0.0050	0.60	0.8217	0.4271	0.9669	0.8261
0.0050	0.70	0.8158	0.3776	0.9771	0.8583
0.0025	0.51	**0.8289**	**0.4736**	0.9596	0.8117
0.0025	0.60	0.8259	0.4566	0.9618	0.8146
0.0025	0.70	0.8258	0.4285	0.9720	0.8494

Table 6: ClassifierOne. Sample: FMS S8

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.0200	0.51	0.8090	0.3204	0.9714	0.7884
0.0150	0.51	0.8132	0.3518	0.9666	0.7777
0.0100	0.51	0.8208	0.4057	0.9587	0.7655
0.0100	0.60	0.8226	0.3761	0.971	0.8117
0.0100	0.70	0.823	0.3641	**0.9755**	0.8316
0.0050	0.51	0.8311	**0.4589**	0.9548	0.7714
0.0050	0.60	**0.8361**	0.4343	0.9697	0.8264
0.0050	0.70	0.8317	0.4062	0.9732	**0.8342**
0.0025	0.51	0.8208	0.4057	0.9592	0.7922
0.0025	0.60	0.8317	0.4411	0.9677	0.8142
0.0025	0.70	0.8351	0.4246	0.9715	0.8321

Table 7: ClassifierTwo. Sample: FMS S8

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.0200	0.51	0.8100	0.3326	0.9686	0.7790
0.0150	0.51	0.8142	0.3585	0.9656	0.7761
0.0100	0.51	0.8188	0.3849	0.9630	0.7755
0.0100	0.60	0.8256	0.3896	0.9735	0.8270
0.0100	0.70	0.8221	0.3596	**0.9785**	0.8314
0.0050	0.51	0.8347	0.4443	0.9646	0.8067
0.0050	0.60	0.8345	0.4341	0.9676	0.8176
0.0050	0.70	0.8351	0.4224	0.9722	**0.8345**
0.0025	0.51	**0.8390**	**0.4586**	0.9655	0.8153
0.0025	0.60	0.8382	0.4330	0.9695	0.8283
0.0025	0.70	0.8355	0.4256	0.9717	0.8335

In the second step, when training with DD data and testing on FMS data, the classifier was found to not perform as well as in the first step. The poor performance, seen in Table 8, 9, 10, and 11 motivated the decision not to continue with further tests, foremost underpinned by the low specificity. As an example only 8 out of 22.226 generated CARs belonged to the Workshop class when using ClassifierTwo, a min-supp of 0.0025 and a min-conf of 0.51. This is the reason why the qualitative analysis was based only on the patterns produced when using FMS data.

Table 8: ClassifierOne. Sample: DD S6

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.02	0.51	**0.7149**	0.0106	**0.9741**	0.1310
0.01	0.51	0.6609	**0.071**	0.8780	0.1768
0.01	0.60	0.7008	0.0345	0.9460	**0.1906**

Table 9: ClassifierTwo. Sample: DD S6

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.02	0.51	0.6702	0.0414	0.9016	0.1342
0.01	0.51	0.6555	**0.0804**	0.8672	0.1822
0.01	0.60	**0.7008**	0.0345	**0.9460**	**0.1906**

Table 10: ClassifierOne. Sample: DD S8

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.02	0.51	0.7400	0.0369	0.9738	**0.3186**
0.01	0.51	0.7226	**0.0620**	0.9554	0.3152
0.01	0.60	**0.7402**	0.0304	**0.9761**	0.2966

Table 11: ClassifierTwo. Sample: DD S8

MinSupp	MinConf	Accuracy	Specificity	Sensitivity	NegCorr
0.02	0.51	**0.7401**	0.0369	**0.9738**	**0.3119**
0.01	0.51	0.7296	**0.0509**	0.9516	0.2587
0.01	0.60	0.7370	0.0388	0.9691	0.2943

When the extracted rules were to be studied, the FMS S8 sample was chosen due to the business knowledge at Scania being more updated concerning this vehicle type. ClassifierTwo was selected over ClassifierOne since it prioritises longer rules. A min-supp and min-conf of 0.0025 and 0.51 were chosen because of the high accuracy and sensitivity. Table 12 is an extraction from the classifier, containing all Workshop rules that classed at least one item. The bolded items are the DTCs from Table 3, previously known as workshop generated. See Appendix A for DTC descriptions.

Table 12: Used Workshop Class Association Rules, extraction from classifier.

Rid	Premises	supp.	conf.	right	wrong
112	**000001_EBS**, **000003_EBS**, 000005_EBS	0.0258	0.8870	43	4
115	**000001_EMS**, **000002_EMS**, 000005_EBS	0.0250	0.8761	42	4
121	**000001_EBS**, **000003_EBS**	0.0273	0.864	12	2
129	000002_RET	0.0152	0.8451	37	4
130	000001_RET,**000001_EMS**, **000002_EMS**	0.0255	0.8417	49	6
132	**000001_EMS**, **000002_EMS**	0.0263	0.8387	2	2
134	000001_EEC	0.0088	0.8333	19	2
136	000001_COO	0.0025	0.8333	3	0
145	000001_RET	0.0298	0.7662	10	2
147	000002_EBS	0.0081	0.7619	22	4
150	000002_OPC	0.0086	0.7556	24	3
157	000007_EBS	0.0028	0.6875	4	5
167	000001_OPC	0.0142	0.5957	23	16
168	000002_COO , 000004_EBS	0.0035	0.5833	8	8
169	000005_EBS	0.0286	0.5567	6	9
170	000004_COO	0.0035	0.5185	9	8
621	**000001_EBS**, **000003_EBS** **000004_OPC**, 000001_RET, 000005_EBS	0.0245	0.8818	40	4
677	**000001_EBS**, **000003_EBS** **000004_OPC**, 000005_OPC 000001_RET, 000005_EBS	0.0222	0.8713	37	4
727	**000003_EBS**	0.0273	0.864	0	1

756	**000001_EBS**, **000003_EBS** **000004_OPC**, 000001_RET	0.0260	0.8583	12	2
805	**000001_EBS**, **000003_EBS** **000004_OPC**, 000005_OPC 000001_RET	0.0238	0.8468	11	2
860	**000004_OPC**, 000001_RET, **000001_EMS**, **000002_EMS**	0.0253	0.8403	1	0
869	000001_EEC, 000003_OPC, 000002_OPC, 000002_RET	0.0076	0.8333	19	2
870	000003_COO, 000002_EBS, 000001_EEC	0.0076	0.8333	19	2
911	**000004_OPC**, 000005_OPC 000001_RET**, 000001_EMS**, **000002_EMS**	0.0230	0.8273	1	0
944	000003_COO, 000002_EBS	0.0081	0.8205	1	0
945	000003_COO	0.0081	0.8205	1	0
947	000003_COO, 000002_EBS, 000003_OPC, 000002_OPC, 000002_RET	0.0078	0.8158	1	0
980	000002_RET, 000003_RET	0.0030	0.8	3	0
996	000002_EBS, 000003_OPC, 000002_OPC, 000002_RET	0.0078	0.7561	2	1
1014	**000004_OPC,** 000001_RET	0.0263	0.7429	0	1
1025	**000004_OPC,** 000005_OPC, 000001_RET	0.0240	0.7252	0	1
1028	000005_OPC	0.0240	0.7252	1	0
1059	**000004_EBS**	0.0038	0.5769	0	2
1064	000002_COO	0.0038	0.5172	1	0

6 Analysis

The analysis begins with a section concerning the effects of labelling the training FMS and DD data. Then a quantitative analysis is provided where the performance of the classifier is examined. This is followed by a qualitative evaluation of the patterns found.

6.1 Error when Labelling Training Data

Before modeling with FMS data, the DTCs were labelled as Workshop or NotWorkshop by matching map data with the GPS coordinates of the vehicle at the time of the DTC timestamp. The probability of a DTC occurring while being at a workshop or not could in this way be calculated. This label can therefore not be considered as an absolute truth since there are possible sources of errors in the procedure. To start with, GPS coordinates are not exact information as they are affected by terrain and weather. Secondly, data is only uploaded from vehicles to the FMS database every ten minutes, giving a possible difference between the DTC timestamp and the GPS coordinates. Finally, the label is calculated based on certain criteria and given a confidence between 0 and 1, implying the precision of the calculation. In this thesis only data with a confidence above 80 percent was used since this limit is applied in other projects at Scania. Today this way of categorizing the data is the most accurate one at Scania and was therefore used in the thesis.

When modeling with DD data, attributes such as time and input source were used to label DTCs as Workshop or NotWorkshop. Due to the poor results when using DD data, this label was probably not correct, rejecting the

hypothesis that a classification can be set by a query as the one explained in chapter *4.1 Sampling Criteria*. The conclusion is that a more accurate labelling must be used when partitioning data into the two classes. This is also the reason why no further analysis was made using only DD data.

6.2 Quantitative Evaluation

When using FMS data, the accuracy was between 80 and 85 percent in almost all tests. Even though this is relatively high, it is only an over-all measure of the classifier. In order to gain a deeper understanding of the performance the other measures have to be further studied.

All tests had sensitivities above 95 percent, referring to the percentage of correctly classified NotWorkshop DTCs. The classifiers also had very low specificities; none of the algorithms classified more than 50 percent of the Workshop DTCs as Workshop. This could partly be explained by the uneven class distribution. As presented in chapter *4.2 The Final Data Set* the S8 FMS data sample had a class distribution of 24.95 percent Workshop and 75.05 percent NotWorkshop DTCs but the model did not take this into consideration when classifying. A similar distribution can be seen in the S6 sample, where 26.90 percent belonged to the class Workshop and 73.10 percent to the class NotWorkshop. A consequence is that CARs belonging to the Workshop class will have a higher min-supp in relation to the class distribution than those of the NotWorkshop class. This results in few rules being generated from the Workshop class, but many more from the dominant NotWorkshop class. When later using such rules to classify, the Workshop CARs will not have as much impact as the NotWorkshop rules, making the classification unbalanced. One solution is to implement a multiple support threshold, further discussed in chapter *8 Future Work*.

Another feature of the classifier that has an effect on the high sensitivity is the implementation of NotWorkshop as the default class. This means that all DTCs in the test sample not classed by a rule were labelled as

NotWorkshop. Especially, all remaining items belonging to the Workshop class were incorrectly categorized as NotWorkshop. However, this can not be seen as a single contributor to the uneven result. In fact, the default class can never increase the specificity as the rate of correct classified Workshop DTCs could only have been higher if the CARs were more accurate. The usage of NotWorkshop as a default class is therefore problematic, but also desirable in relation to the contrast of having the Workshop class as the default class.

The negative correctness refers to the percentage of correct classified DTCs in relation to all DTCs predicted as Workshop. By looking at this number, a sense of the Workshop CARs' precision can be given. As seen in Tables 4, 5, 6, and 7 this number was rather high, indicating that more than 80 percent of the items classed as Workshop were correctly labelled. This implies that the items labelled as Workshop by the classifier were often correct. Concerning the not found Workshop DTCs, they are either not included in any significant patterns or not found due to shortcomings in the classifier.

6.2.1 Consequences of Parameter Settings

As described in chapter *3 Context of Study*, the smallest min-conf used was 0.51. Min-confs of 0.60 and 0.70 were also used to see if any significant difference could be distinguished. The only change in result that could be seen was an improvement in the negative correctness. This is probably due to poor Workshop rules being neglected as the min-conf is increased.

When selecting min-supp, the aim was to set the constraint low enough to include as many Workshop class rules as possible without making the model overfit. This resulted in a range of min-supps spanning from 0.02 to 0.0025. Both lower and higher min-supps were tested but as those supports generated poor results they were not included in the results. When lowering the min-supp, there was a marginal improvement in accuracy and specificity. Yet, more tests have to be made on larger data sets, preferably using a cross validation with a k greater than five in order to decrease the impact the data

samples have on the modeling.

6.2.2 Performance of the Two Algorithms

The two algorithms, ClassifierOne and ClassifierTwo barely differ in terms of results. When using the FMS samples, both algorithms reached similar accuracy, specificity and negative correctness and can therefore be seen as equal in their over-all performance. The main differences are revealed when studying the CARs from the classifier, where ClassifierTwo has longer patterns due not removing items directly when classified.

6.3 Qualitative Evaluation

In the qualitative analysis, the rules from the S8 sample with a min-supp of 0.0025 on ClassifierTwo were selected. This was done primarily since there is more business knowledge regarding this sample but also because ClassifierTwo generates longer rules. Some of the rules in Table 12, containing used Workshop rules from the final classifier, were found to only include one item, making them not so interesting in the sense of pattern mining. They will consequently not be studied further.

When reviewing the found patterns most of the previously known Workshop DTCs mentioned in Table 3 were found. It is noteworthy that most of them occur at the top of the rule list, confirming what is already known; they are frequently occurring in workshops. However, one of the DTCs in Table 3, 000006_EBS, was unexpectedly not found in any Workshop rules in the classifier. Instead it was found as NotWorkshop in 27 rules.

In multiple rules the DTCs occur in groups, often ranked with higher confidence than when alone. This indicates that when in these patterns, they are even more likely to be set in a workshop.

Interesting to notice is also that almost all of the DTCs from Table 3 occur in some rules together with DTCs that have not yet been distinguished

as Workshop DTCs. Such DTCs can be considered to have high unexpectedness and therefore be of certain interest. Examples of such DTCs are 000005_EBS, 000001_RET and 000005_OPC.

Some Workshop rules did not include any of the DTCs from Table 3. Two examples are rule 869 and 870, both having a relatively high confidence of 0.8333 but a rather low support of 0.00758. They are contributing with new knowledge about Workshop DTCs and are therefore interesting with high unexpectedness.

Rule 869: $\{000001\_EEC, 000003\_OPC, 000002\_OPC, 000002\_RET\} \rightarrow \{Workshop\}$

Rule 870: $\{000003\_COO, 000002\_EBS, 000001\_EEC\} \rightarrow \{Workshop\}$

Before implementing the algorithms, a count was added to each rule to see how many times the rules classed correct or incorrect. These are the two right columns in Table 12. The numbers indicate that some of the rules classed truthfully in many cases, sometimes over 40 right, and at the same time incorrectly labelled items in just a few occasions. They are therefore considered as extra interesting. Following patterns are extracted from the rules in Table 12:

Rule 112: $\{000001\_EBS, 000003\_EBS, 000005\_EBS\} \rightarrow \{Workshop\}$

Rule 115: $\{000001\_EMS, 000002\_EMS, 000005\_EBS\} \rightarrow \{Workshop\}$

Rule 130: $\{000001\_RET, 000001\_EMS, 000002\_EMS\} \rightarrow \{Workshop\}$

Rule 132: $\{000001\_EMS, 000002\_EMS\} \rightarrow \{Workshop\}$

Rule 621:	{000001_EBS, 000003_EBS, 000004_OPC, 000001_RET, 000005_EBS}	$\rightarrow \{Workshop\}$
Rule 677:	{000001_EBS, 000003_EBS, 000004_OPC, 000005_OPC, 000001_RET, 000005_EBS}	$\rightarrow \{Workshop\}$

As can be seen by comparing these to the information in Table 3, most of them contain already known Workshop DTCs together with 000001_RET and/or 000005_EBS.

By studying the rules from the classifier using a min-supp of 0.0025 and min-conf of 0.51, a hint can be given that these parameter settings might have been too low, leading to an overfitting being initiated. This is especially apparent when looking at rules with low support and confidence that have rather equal numbers of correct and incorrect classified DTCs. Examples of such rules are 168, 169 and 170 in Table 12, all with very low confidences. Hence, they are not representative for the Workshop class. By setting a higher min-conf such rules would not have been generated. Noticeable is though that such low parameter settings did not decrease the performance of the classifier remarkably. In future work lower min-supp and min-conf than 0.0025 and 0.51 should not be used.

To enhance the hypothesis that DTCs occur in certain patterns when being set in workshops, it was important to compare the Workshop rules to the NotWorkshop rules. By doing this it could be distinguished that 000005_EBS occurred in certain patterns when being Workshop, and in other patterns when being NotWorkshop. As an example, when occurring in Rule 97 together with the 000001_EBS and 000003_EBS, 000005_EBS is likely to have been set in a workshop. On the other hand, when in Rule 82 with 000006_COO, 000005_EBS is likelier to have been set outside a workshop.

Rule 97:	$\{00A0001\_EBS, 000003\_EBS, 000005\_EBS\}$	$\rightarrow \{Workshop\}$
Rule 82:	$\{000005\_COO, 000005\_EBS\}$	$\rightarrow \{NotWorkshop\}$

No deeper understanding can be gained by only studying the DTC descriptions. However, what can be pointed out is that most of them are timeout errors or lack of CAN messages, see Appendix A. Interesting to notice is also that out of all ECUs included in the analysis, only a few of them occur in Table 12. EBS and RET are two ECUs often occurring in the found Workshop patterns, meaning they could be sensitive to disturbance and testing. This indicates that these components should be further studied in order to detect why they are alarming frequently.

6.3.1 Causality

The question remains whether there is not just a correlation between the discovered patterns, but also causality. In order to answer such question, every pattern must be fully understood and analysed beyond what can be done solely by the authors of this thesis. That is why the found patterns were presented to several experts at Scania. Personnel working with statistics, data analysis, vehicle workshop testing and warranty processes were asked to contribute with their business knowledge. In one of the interviews, it was confirmed that timeouts are expected in workshops due to disconnecting or updating control units. It was moreover a shared view that the EBS is an ECU sensitive to disturbance. This is confirmed by the results.

In general, none of the interviewed experts could gain deeper understanding by just reading the DTC descriptions of the codes in the patterns. Everyone yet agreed that the rules are interesting and that a workshop label would be preferable in DD, leading to a potential actionability in the future. This will require investments in further studies and an actual implementation.

7 Conclusion

Patterns in DTC data can be distinguished by building and testing a classifier. This is confirmed by the fact that about half of all Workshop DTCs could be found with an 80 percent accuracy. It can not be said whether the remaining Workshop DTCs were not found due to lack of precision in the classification algorithm or if they simply did not occur in significant Workshop patterns. When studying the rules in the classifier the majority of the previously known workshop generated DTCs were found. This confirms the hypothesis that some DTCs are more likely to occur in workshops. Patterns were also found where already known Workshop DTCs were occurring with previously not known Workshop DTCs. In this sense the classifier could contribute with new knowledge. In particular, some rules were found to be of certain interest either due to their high correct count or high confidence. DTCs included in such rules are 000005_EBS, 000001_RET and 000005_OPC. Some DTCs were also of interest due to their existence in rules where both the class and the premises deviated. In order to implement a flagging of the found patterns further studies are however crucial.

8 Future Work

To continue the search of patterns in Workshop DTCs, larger data samples should be used. One way of extracting such samples is to increase the sample period, another is to decrease the number of shared ECUs. Examples of ECUs that should be included are though the EBS, RET, OPC, COO and the engine type. These are sensitive to disturbance and cast many DTCs. Consideration should be given to the fact that when excluding ECUs the risk of losing patterns increases. If the data set increases, a 10-fold cross validation can be applied to decrease the effect of segmenting data and do a more comprehensive testing.

As stated in the analysis, consideration should further on be taken to the uneven distribution between the two classes. If still using an Associative Classifier, a range of approaches can be applied. Multiple supports is a way of handling uneven distribution among classes where a higher min-supp for the NotWorkshop class and lower for the Workshop class can be used.

It would be interesting to test other pruning techniques on FMS data to examine if the accuracy can be improved. Techniques where only usedRules are included in the classifier could be tested and compared to the L^3 algorithm. If, in the future, more knowledge regarding the designed DTCs would exist, tree-rule extracting methods such as Decision Trees could be used. By testing new approaches, the results of this thesis can be compared and further analysed to examine if patterns only exist in 50 percent of all Workshop DTCs, or if there are simply other more suitable methods. As stated in the objectives of this thesis, Scania required high understanding of the method

as well as the results. If these requirements would change, more complex methods such as Random Forest or Neural Networks could be applied.

The classifier in this thesis could also be used to class other types of DTCs if training data is available. One such categorisation could be to examine if certain type of updating leads to DTC patterns. If, on the other hand, wanting to predict repairs on a chassis, the two algorithms developed in this thesis can be replaced with more regular classification methods.

In order to further analyse DD data, the query to create the two classes needs to be modified. In this thesis, DTCs occurring one day within the time of a workshop visit are labelled as Workshop. This was not sufficient and therefore the estimation should be adjusted, perhaps by further limiting the time constraint. However, the authors of this thesis suggest that modeling should be restricted to FMS data until the classifier reaches a satisfying performance.

References

[1] Aguinis, H., Forcum, L. E. & Joo, H. (2013) "Using Market Basket Analysis in Management Research", Journal of Management, vol: 39, no: 7, pp. 1799-1824

[2] Baralis, E., Chiusano, S., & Paolo, G. (2008), "A Lazy Approach to Associative Classification", IEEE Transactions on Knowledge and Data Engineering, vol: 22, issue: 2, pp. 156-171

[3] Baralis, E., & Garza, P. (2002), "A Lazy Approach to Pruning Classification Rules", Proceedings to the 2002 IEEE International Conference on Data Mining, pp. 35-41

[4] Bernd, T. (1999), "Nonlinear Black Box Modelling - Fuzzy Networks versus Neural Networks", Neural Computing & Applications, vol: 8, issue:2, pp. 151-162

[5] Chen, S. W., Hsu, L. B., & Ma, Y. (2000), "Analyzing the Subjective Interestingness of Association Rules", Intelligent Systems and their Applications, IEEE , vol: 15, issue: 5, pp. 44-55

[6] Diederich, J. (2008), Rule Extraction from Support Vector Machines. Springer: Berlin

[7] Frank, E., Hall, M., & Witten, I. H. (2011), Data Mining - Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers: Burlington

[8] Garcia, S., Herrera, F., & Luengo, J. (2015), Data Preprocessing in Data Mining. Springer International Publishing: Cham

[9] Gorunescu, F. (2011), Data Mining - Concepts, Models and Techniques. Springer: Berlin

[10] Han, J., & Kamber, M. (2001), Data Mining - Concepts and Techniques. Academic Press: San Diego

[11] Hand, D., Manilla, H., & Smyth, P. (2001), Principles of Data Mining. MIT Press: Cambridge

[12] Kantardzic, M. (2011), Data Mining: Concepts, Models, Methods, and Algorithms. Wiley-IEEE Press: Hoboken

[13] Krzysztof, C. J., Pedrycz, W., Roman, S. W., & Lukasz, K. A. (2010), Data Mining - A Knowledge Discovery Approach. Springer US: New York

[14] Larose, C., & Larose, D. (2014), Discovering knowledge in Data: An introduction to Data Mining. John Wiley & Sons: Hoboken

[15] Liu, B., Hsu, W., & Ma, Y. (1998), Integrating Classification and Association Rule Mining. National University of Singapore: Singapore

[16] Ljung, L. (2001), "Black-box Models from Input-out put Measurements", Linköpings Universitet: Linköping

[17] Mayatt, G. J. (2007), Making Sense of Data - A Practical Guide to Exploratory Data Analysis and Data Mining. John Wiley & Sons: Hoboken

[18] Tan, P.-N., Steinbach, M., & Kumar, V. (2014), Introduction to Data Mining. Pearson Education Limited: Edinburgh

[19] Thabtah, F. (2007), "A Review of Associative Classification Mining", The Knowledge Engineering Review, vol: 22, issue: 1, pp. 37-65

[20] Wenmin, L., Jaiwei, H., & Jian, P. (2001), "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules", Proceeding to the 2001 IEEE International Conference on Data Mining, pp. 369-376

[21] Ye, N. (2003), The Handbook of Data Mining. CRC Press: Boca Raton

[22] Yin, X., & Han, J. (2003), "CPAR: Classification Based on Predictive Association Rules", Proceedings of the Third Siam International Conference on Data Mining, pp. 331-335

[23] Zhang, C., & Zhang, S. (2002), Association Rule Mining: Models and Algorithms. Springer: Berlin

Appendix A

Table 13: DTC description

DTC Code	DTC Description
000001_EBS	Timeout coordinator
000002_EBS	The CAN bus
000003_EBS	Timeout tachograph
000004_EBS	The CAN bus
000005_EBS	CAN message
000006_EBS	Error from engine
000007_EBS	CAN message
000001_EMS	CAN Message from coordinator timeout
000002_EMS	CAN Message from tachograph timeout
000003_EMS	Lost VGT activator
000001_RET	Timeout from coordinator
000002_RET	Timeout from engine
000003_RET	Timeout
000001_OPC	Tachograph
000002_OPC	CAN communication from engine control unit
000003_OPC	CAN communication from engine control unit
000004_OPC	Timeout from coordinator
000005_OPC	Gear lever position
000001_COO	Electrical error on Green bus
000002_COO	Timeout in CAN communication
000003_COO	Timeout in CAN communication
000004_COO	Timeout in CAN communication
000005_COO	CAN message
000001_EEC	The communicator with the Ems control unit is lost